SICORE, un outil et une méthode pour le chiffrement automatique à l'INSEE

Eric Meyer et Pascal Rivière, INSEE, France

SICORE (Système Informatique de COdage des Réponses aux Enquêtes) est un système de chiffrement automatique développé par le Département des Applications et des Projets de l'INSEE, qui a fait l'objet de tests sur de nombreuses variables et a déjà été appliqué en production avec succès. Son usage est appelé à se généraliser à l'INSEE, et au sein d'autres services statistiques, français et éventuellement étrangers. Le but de ce document est de présenter la méthodologie SICORE, de décrire le système SICORE actuel, son fonctionnement, et enfin d'expliciter son utilisation en pratique.

1. La méthodologie SICORE

Pour bien comprendre SICORE, nous allons tout d'abord définir ce qu'est le codage, avant d'aborder le principe du codage automatique QUID qui a servi de point de départ pour SICORE. Ce n'est qu'après ce préalable que nous pourrons mettre en évidence ce qui a conduit à l'élaboration de SICORE.

1.1. Le codage

Le codage, également appelé chiffrement, est une opération qui consiste à interpréter le sens d'un libellé, en attribuant à ce dernier un code. Le libellé doit décrire un certain concept statistique, c'est-à-dire une certaine variable, et le code doit appartenir à une certaine nomenclature, que l'on suppose connue. Il se peut aussi que le texte seul ne suffise pas pour le chiffrement, et que l'on ait alors besoin de *variables annexes* pour coder : ainsi, pour le chiffrement de la profession, le statut ou la qualification font partie des variables supplémentaires généralement utiles.

L'opération de codage est indispensable, dans une enquête, dès lors qu'il existe des questions ouvertes dans le questionnaire et que l'on veut les traiter statistiquement : en effet, il est nécessaire de diviser les réponses en plusieurs catégories, et placer une réponse dans une catégorie n'est rien d'autre que coder.

En pratique, le codage peut être réalisé de trois façons :

- le codage manuel dans lequel l'opérateur utilise ses propres connaissances et éventuellement des documents sur papier,
- le codage assisté dans lequel l'opérateur s'aide des fonctionnalités d'un logiciel mis à sa disposition, avant de prendre sa décision finale,
- le codage automatique où l'informatique réalise seule le codage, avec pour conséquence l'existence de rejets de codage qu'il faudra traiter ultérieurement par l'une des deux méthodes précédentes.

Le chiffrement "à la main" a l'inconvénient d'être très coûteux, car il requiert beaucoup de temps ; c'est pourquoi les services statistiques cherchent de plus en plus à réaliser cela soit de façon assistée, avec des logiciels interactifs d'aide au codage, soit avec des programmes de codage automatique.

1.2. QUID

L'INSEE s'est préoccupé de codage automatique depuis longtemps : l'algorithme QUID, mis au point en 1979 par l'unité de recherche de l'INSEE, a donné de bons résultats. Le logiciel qui met en pratique cette méthode a été utilisé sur de nombreuses applications, en particulier le codage de la profession dans l'enquête Emploi, et le codage de la catégorie socio-professionnelle et de la commune dans les déclarations annuelles de données sociales.

La méthode QUID est générale, au sens où elle est indépendante de la variable à coder. Mais pour la mettre en oeuvre, il est nécessaire de disposer d'un fichier de référence reliant les libellés aux codes, appelé *fichier d'apprentissage* : il s'agit en quelque sorte d'une nomenclature fondée sur les réponses effectives des enquêtés passées.

L'algorithme fonctionne alors en deux temps. Dans une première phase, il "apprend" le fichier, l'assimile, et crée une structure synthétique du fichier ainsi digéré, structure appelée *arbre de questionnement*. Dans un deuxième temps, le programme est capable de coder, en parcourant cette structure : l'arbre généré a en effet l'avantage d'être remarquablement adapté au chiffrement, qui se révèle donc très rapide avec cette méthode.

1.3. Les problèmes rencontrés

Le codage automatique ne permet jamais de traiter tous les libellés : il reste une proportion de non-codés, qui n'est pas toujours négligeable. Une analyse de ces échecs devrait permettre d'améliorer petit à petit le fichier d'apprentissage, en y incorporant les intitulés les plus fréquents parmi ceux que l'on n'a pu chiffrer auparavant. Comment et en fonction de quoi le mettre à jour ? Avec quels outils ? Selon quels critères ?

En pratique, cette amélioration régulière et raisonnée du fichier d'apprentissage n'a pas réellement eu lieu avec QUID ; à titre d'exemple, l'enquête Emploi, qui utilise cette méthode, n'a pas changé ce fichier de référence depuis la création de la chaîne de codage automatique, en 1990.

On peut aisément expliquer cette lacune. En fait, l'enrichissement d'un fichier d'apprentissage nécessite une organisation adaptée : des experts, spécialisés dans le chiffrement d'une variable, des logiciels facilitant le travail de mise à jour effectué par les experts (ateliers d'expertise), une centralisation des connaissances sur chaque domaine, et une prise en compte réelle des chiffrements automatiques ayant déjà eu lieu.

Tout cela n'était pas pris en compte par les structures et outils existants : le chiffrement automatique proprement dit fonctionnait bien, mais le travail qui restait à faire était *autour* du chiffrement.

1.4. Le projet SICORE

SICORE a été lancé, vers la fin 1993, pour deux raisons essentielles. En premier lieu, comme nous venons de le voir, il fallait bâtir une organisation et mettre au point des outils autour du codage. Mais le souhait était également de généraliser l'utilisation du codage automatique à l'INSEE : pour cela, il devenait indispensable de rendre cette opération plus facilement accessible aux non-informaticiens.

La volonté a donc été affichée dès le départ de créer un système simple, unificateur, efficace et portable pour le chiffrement automatique, avec un algorithme de codage s'appuyant mathématiquement sur QUID, mais cherchant à l'assouplir et à en généraliser l'approche. De plus, comme l'efficacité n'est jamais de 100%, il fallait préparer le cycle d'enrichissement des connaissances.

Cela passait d'abord par l'écriture d'outils en plus de la codification automatique : visualisation des libellés non-codés, ou bien analyse de fichiers de connaissances sur le codage (en particulier le fichier de référence). Cela nécessitait que les connaissances statistiques existent, et qu'elles évoluent régulièrement sur la base de ces analyses. Enfin, cet objectif ambitieux exigeait également que soient mis en place des repères méthodologiques et organisationnels pour la mise à jour de connaissances.

Cet ensemble "programmes + connaissances statistiques + méthodologie + organisation" constitue ce que nous appellerons un **système** de codage automatique, prenant en compte l'ensemble des tenants et aboutissants.

Pour arriver à cela, le principe de base du projet a toujours été de "faire d'abord", c’est-à-dire de fabriquer le plus tôt possible des prototypes et de les appliquer. Ceci permet en effet d'anticiper sur les outils, de se donner la possibilité de les ajuster, et de créer la méthodologie et l’organisation de façon naturelle et pragmatique. Il ne fallait pas imposer *a priori* des outils et une méthodologie : ceux-ci devaient se dégager de l'utilisation courante. Il en est de même pour l'organisation : une organisation ne se décrète pas, elle naît de l'action.

2. Le système SICORE

Le projet SICORE est à l’heure actuelle achevé : le système SICORE, au sens proposé plus haut, existe effectivement. En effet, on dispose bien

des 4 composantes préconisées, mais elles ont toutes vocation à évoluer ; les connaissances s'enrichissent pour améliorer l'efficacité et la qualité des futurs chiffrements ; de nouveaux principes méthodologiques émergent d'une utilisation régulière du codage automatique ; enfin, les organisations sont malléables et éphémères, leur existence dépendant de leur utilité réelle et de la volonté des acteurs.

2.1. Les programmes et les connaissances

2.1.1. La séparation programmes — connaissances

C'est là un des apports fondamentaux de SICORE. Les programmes de codage automatique de SICORE sont généraux : ils ne concernent aucune variable particulière ; à eux seuls, ils sont donc incapables de réaliser le moindre chiffrement. Pour fonctionner, il leur faut des *connaissances* sur la variable à coder, complètement indépendantes des programmes généraux.

Le fichier d'apprentissage, que QUID utilisait déjà, est l'une de ces connaissances. Mais SICORE en emploie d'autres, notamment les *règles de normalisation* (synonymes, mots vides, troncatures, caractères à éliminer) et les *règles logiques* (prise en compte des variables annexes dans des règles de décision de type "système expert"), et les *paramètres d'apprentissage* (permettant à l'utilisateur expert en variable d'orienter l'algorithme d'apprentissage comme il le désire).

Toutes ces connaissances dépendent de la variable à coder, mais il s'agit de connaissances générales dans leurs domaines respectifs : ce sont toujours les mêmes, quelle que soit l'enquête. Pour atteindre cette généralité, les règles de décision, qui mettent en jeu les variables annexes, font intervenir la modalité "manquant", ce qui permet de coder les enquêtes pour lesquelles la variable annexe est absente.

Mais il existe également des connaissances spécifiques à l'enquête qui viennent enrichir les connaissances générales pour répondre au mieux aux besoins spécifiques de traitement : dessin de fichier à coder, table de passage entre les variables annexes de l'enquête et celles des règles logiques.

En définitive, **c'est toujours un couple [SICORE — bases de connaissances] qui réalise le codage**. On emploie, par souci de simplification, l'expression "SICORE code ...", mais elle est en toute rigueur impropre. La distinction nette qui est faite entre programmes généraux et connaissances offre au statisticien l'avantage de bien maîtriser ce qu'il fait : en quelque sorte, les connaissances sont des spécifications écrites à l'extérieur des programmes.

2.1.2. Le fonctionnement du codage automatique avec SICORE

Pour coder, SICORE utilise toutes les bases de connaissances dont il dispose, et opère en trois temps.

En premier lieu, il simplifie le libellé, grâce à des règles de normalisation (mots vides, synonymie, ...) : on passe d'un texte à un autre texte, on ne change donc pas d'univers.

Dans un deuxième temps, il faut passer de l'univers des textes à celui des codes : on va transformer notre libellé simplifié en un code, éventuellement imprécis, après apprentissage du fichier de référence. Sur le plan théorique, l'algorithme s'inspire largement de l'algorithme QUID[12], mais en diffère cependant par certains aspects, ce qui a permis non seulement de gagner un temps important (SICORE effectue un apprentissage 40 fois plus vite que le logiciel QUID sur certains essais), mais aussi d'améliorer l'efficacité de codage, et enfin d'obtenir une meilleure stabilité de l'arbre de questionnement.

Si le code obtenu est un code intermédiaire, on lève l'ambiguïté sur le code définitif grâce aux règles logiques de codage intégrant les variables annexes, informations supplémentaires que nous avons déjà évoquées.

Avec SICORE, le codage d'un libellé est très rapide, même si les bases de connaissances sont de taille importante : de l'ordre de quelques millièmes de seconde tant sur site central que sur un PC de type pentium.

2.1.3. SICORE en tant que logiciel

SICORE est un **ensemble de programmes qui sont des briques élémentaires pour le codage** : lecture de connaissances, apprentissage, normalisation, reconnaissance de libellé, traitement de variables annexes, ... Ces briques sont écrites en langage C (norme ANSI), et fonctionnent aussi bien sur site central MVS que sur PC (WINDOWS 3.1, WINDOWS 95, WINDOWS NT 4).

Sur site central, il n'y a rien de plus que les briques élémentaires, qui sont appelables par des programmes.

Sur PC, toutes ces briques sont réunies au sein d'un seul et même logiciel. Outre les programmes de chiffrement automatique, il comporte une interface permettant de créer, d'analyser et de mettre à jour des connaissances. L'objectif majeur de ce logiciel, appelé *l'outil SICORE*, est de **mettre au point des connaissances** ; le codage automatique fait partie des outils qui contribuent à cette mise au point, mais il n'est pas ici une fin en soi. L'outil SICORE a très peu d'utilisateurs (il est utilisé sur 7 postes aujourd'hui) : ce sont les experts de variables.

2.2. Bases de connaissances

SICORE a été appliqué à de nombreuses variables, ce qui a nécessité, dans plusieurs cas, la création de bases de connaissances. A l'heure actuelle, de telles bases existent en langue française pour les variables suivantes : SICAV, communes, départements, nationalités, professions et

[12]Mais pas sur un plan informatique : aucun programme du logiciel QUID n'a été réutilisé.

CS, occupations, lieux de séjour, raisons sociales et adresses d'établissements.

Toutes ont conduit à des tests sur des données réelles : par exemple, le codage de la profession par SICORE a été appliqué à des fichiers de quelques dizaines de milliers de libellés, provenant du Recensement de la Population, des Déclarations Annuelles de Données Sociales, de l'Etat-Civil, de l'enquête Emploi, ou de l'enquête Permanente Conditions de Vie.

Ces bases de connaissances sont de tailles très variables : presque 5 millions de libellés pour le fichier d'apprentissage des établissements au niveau national[13], de l'ordre de 45000 pour celui des communes[14], un peu plus de 4000 pour les SICAV, quelques centaines pour les départements ; de même, on peut avoir aussi bien quelques synonymes épars (nationalités, SICAV) que plusieurs centaines (occupations), voire plusieurs milliers (2000 pour la profession).

Par ailleurs, des négociations sont en cours pour élaborer des bases de connaissances en vue du codage de la profession dans le recensement américain, ce qui démontrera l'indépendance du logiciel par rapport à la langue de traitement.

2.3. Méthodologie

Pour mettre au point les connaissances dans de bonnes conditions, une méthodologie d'utilisation a été constituée petit à petit. La documentation méthodologique, qui a découlé de ces réflexions, aborde des sujets variés : description détaillée du fonctionnement du chiffrement automatique avec SICORE, principes de contrôle de cohérence, détection d'erreurs dans une base de connaissances, choix des connaissances à modifier, mesures de qualité, mise à jour des connaissances au fur et à mesure du traitement de l'enquête, codage de libellés hétérogènes,

La méthodologie ne doit surtout pas être considérée comme une chose définitive et figée : SICORE a dégagé un certain nombre de principes généraux, mais il est probable que l'on en découvrira de nouveaux, ou que l'on en viendra à amender certaines règles existantes.

2.4. Organisation

SICORE nécessite l'existence d'une organisation pour deux raisons bien distinctes : d'une part, pour faire en sorte que les bases de connaissances soient "vivantes", c'est-à-dire évoluent et s'enrichissent régulièrement ; d'autre part, pour être prêt à assurer la mise en place du chiffrement automatique dans une enquête ou une source quelconque.

---

[13]Test qui a nécessité un serveur doté de 2 gigaoctets de mémoire centrale.

[14]Il y a souvent, en pratique, plusieurs libellés possibles pour la même commune, par exemple "ISSY LES MOULINEAUX" et "ISSY LES MLX".

Pour le premier objectif, SICORE a créé la notion d'*expert de variable*, dont le rôle est de mettre au point les bases de connaissances relatives à la variable en question. Pour cela, chaque expert dispose de *l'outil SICORE*, sur PC. L'ensemble des variables devant faire l'objet de ce travail d'expertise n'est pas fermé : il est destiné à évoluer. Le projet SICORE a commencé avec les variables "Profession" et "Commune".

Mais la présence d'experts ne suffit pas. Pour avancer, ceux-ci doivent pouvoir se nourrir d'autres expériences, provenant d'autres enquêtes que celles qu'ils connaissent, ou bien de l'expérience des personnes de terrain. C'est pourquoi des *groupes de travail par variable* se sont créés, autour des variables Commune et Profession. Ces groupes se réunissent deux à trois fois par an ; il est probable qu'ils évolueront, aussi bien dans leur composition que dans leur périodicité de réunion, ou leurs objectifs.

Enfin, pour qu'une "culture d'expert de variable" se crée, il existe également un *club utilisateurs* de l'outil SICORE. Il ne réunit que les experts (seuls utilisateurs de ce logiciel), et permet de faire le point sur les éventuelles lacunes de l'outil, et donc de faire remonter les spécifications.

Tous ces réseaux (groupes de travail par variable, club utilisateurs) sont animés par l'*expert SICORE*, membre de l'unité de méthodologie de l'INSEE, qui est la plaque tournante de l'ensemble de l'organisation.

Le deuxième objectif, c'est-à-dire la mise en place du chiffrement automatique dans une application donnée, nécessite une organisation complètement différente, à nouveau centrée autour de l'expert SICORE. Il s'agit ici de mettre en correspondance trois entités : l'enquête (représentée par un responsable statistique et un responsable informatique), SICORE (représenté par l'expert SICORE et le responsable informatique de SICORE), et la variable à coder (représentée par l'expert de variable).

La difficulté de cette mise en place varie énormément selon la variable et selon l'enquête considérée : il arrive que cela ne prenne que quelques jours, mais il se peut aussi que ce soit long, notamment quand les bases de connaissances n'existent pas déjà.

3. Utilisation de SICORE en production

3.1. Pratique de SICORE

SICORE n'a rien d'un outil clé en main, et ce serait d'ailleurs impossible pour un outil général : ce ne sont que des *modules élémentaires de codage*, qu'il faut intégrer dans une application, et auxquels il faut joindre les bases de connaissances. Ainsi, le module de codage automatique (proprement dit) reçoit-il, en entrée, un libellé additionné d'éventuelles variables annexes ; en sortie, il renvoit un résultat de codage et un code retour.

Mais ce n'est pas tout : outre l'intégration des modules élémentaires, le codage automatique engendre plusieurs travaux, en amont et en aval, dont on ne doit pas sous-estimer l'ampleur. Ainsi, l'on doit prévoir la saisie

des libellés, dont le coût n'est pas négligeable, quelque soit la sophistication du poste de saisie. Il faut également écrire des logiciels conviviaux pour le traitement des "rejets" du codage automatique, c'est-à-dire des outils de codage assisté. Enfin, la mise en place du codage au niveau statistique nécessite un minimum de préparation ; dans le cas de "nouvelles" variables, cela conduit à la création de bases de connaissances ex-nihilo, ce qui peut s'avérer long (cela dépend en fait de l'existence ou non d'une première nomenclature reconnue).

3.2. Applications de SICORE

SICORE a été appliqué à l'enquête sur les transports dans l'agglomération lyonnaise, appelée SYTRAL, dans laquelle il fallait coder des **communes**, en l'occurrence des points de départ et d'arrivée pour chaque utilisation d'un transport en commun. Sur quelques milliers de libellés, seuls trois n'ont pas été codés par SICORE : le chiffrement assisté n'a donc pas été véritablement nécessaire.

Il est difficile d'en tirer des conclusions sur un plan statistique, car le chiffrement de la commune n'est certes pas le plus difficile. En revanche, informatiquement, on a pu voir que dans ce cas (où il n'y avait pas véritablement d'intégration dans une chaîne de traitements), le temps de travail nécessaire était très minime (un à deux jours).

SICORE a également servi à chiffrer **la CS (variante de la profession) et les lieux de séjour** dans l'enquête Permanente Conditions de Vie de l'INSEE. La situation était idéale pour une évaluation de SICORE, puisqu'un codage manuel avait été fait au préalable.

La codification des CS a porté sur 9992 libellés, celle des lieux de séjour sur 12239. **SICORE a codé automatiquement 76% des CS et 92% des lieux de séjour**[15]. Il restait à savoir si les codes SICORE étaient "bons"[16]. Pour évaluer cela, une technique s'imposait : rechercher dans combien de cas le code automatique différait du code manuel, puis analyser une par une ces divergences.

Dans les cas des lieux de séjour, lorsqu'on étudie les écarts entre codage automatique et codage manuel, on s'aperçoit qu'il y a très peu de différences : seulement 3% des libellés (315 exactement) codés par SICORE avaient des résultats distincts du codage manuel. En étudiant un par un les libellés en question, on s'aperçoit que **le code SICORE est juste dans 82% des cas, et le code manuel dans les cas restants**.

Parmi les erreurs de SICORE, citons l'exemple de "CANARIES" (qui n'était pas dans le fichier d'apprentissage), codé comme la ville corse "CANARI".

Le codage d'une CS est en revanche une opération plus délicate, ce qui implique qu'il y ait beaucoup plus de divergences entre codage automatique et codage à la main : parmi les intitulés de profession codés par SICORE, plus de 1 sur 6 (18%) divergent du code manuel. La question se pose à nouveau : qui a raison ? Les différences de chiffrement de CS ont donc été analysées : les experts Profession s'en sont chargés.

---

[15]C'est-à-dire lieu de vacances, décrit sous la forme d'une ville, d'un département, d'un pays étranger, d'un circuit, ...

[16]En effet, il ne faut pas espérer d'une méthode de codage automatique que *tous* ses chiffrements soient exacts : le seul moyen, pour cela, serait de ne coder que des libellés strictement identiques à ceux du fichier d'apprentissage (et encore, il peut subsister des erreurs dans celui-ci ...).

L'étude des divergences montre une nouvelle fois que c'est plutôt le codage automatique qui donne les meilleurs résultats : dans 62,5% des cas, le code SICORE est le bon code. A l'inverse, le codage manuel a raison dans 17,5% de ces cas de divergence. Enfin, dans les 20% restants, les deux codes peuvent convenir : il y a ambiguïté ; celle-ci est souvent due à la non-utilisation du libellé d'activité en clair.

Bien entendu, les principes de base de SICORE ont été immédiatement mis en pratique : toutes les erreurs de codage ont été réanalysées une par une, de même que les libellés non-codés, ce qui a conduit à améliorer les bases de connaissances en conséquence.

Du côté des applications de SICORE à l'extérieur de l'INSEE, citons le codage de la CS au sein du CEREQ, organisme français chargé de statistiques sur les qualifications, mentionnons également les négociations actuellement en cours avec le bureau du CENSUS américain pour l'emploi de SICORE pour le recensement fédéral.

3.3. L'intégration de SICORE

Une fois réglées toutes les questions de mise au point des *connaissances SICORE* par l'*outil SICORE* sur PC, il faut déterminer de quelle façon l'on souhaite utiliser SICORE.

Pour une réussite optimale, l'enquête utilisatrice de SICORE doit au préalable fixer la méthode d'intégration, ce qui revient en fait à arbitrer le couple "degré d'intégration" "personnalisation du codage automatique".

Les deux méthodes proposées dans l'intégration sont en quelque sorte les deux extrêmes de cet arbitrage :

- l'intégration des briques SICORE constitue un moyen très souple d'intégration car cette méthode permet d'ordonner et de maîtriser les appels à SICORE, par exemple en travaillant simultanément sur plusieurs variables. En revanche, elle exige que l'application ait une forte adhérence à SICORE, notamment de par son implémentation. Il ne faut également pas perdre de vue que les compétences requises sont celles d'un informaticien maîtrisant le langage C[17] ,
- l'intégration des traitements SICORE, quant à elle, permet de ne pas lier les traitements de SICORE avec ceux de l'applicatif, puisque seul l'enchaînement des traitements est impacté. Cependant, il ne peut s'agir que de traitements prédéfinis dont l'unité indivisible est le fichier, ce qui ne lui confère pas beaucoup de souplesse. La compétence nécessaire est du niveau utilisateur ou utilisateur confirmé, capable d'enchaîner les étapes d'un travail (par exemple grâce à la maîtrise du JCL dans le monde MVS). Il faut alors que les traitements sur les fichiers (normalisation et codage) soient disponibles sur la plate-forme envisagée pour l'exécution.

L'important pour faire le choix est d'avoir en tête les tenants et aboutissants de chaque solution, sachant qu'il est toujours possible de faire un choix pour une partie de l'application et un autre pour une deuxième : tout n'est alors qu'affaire d'organisation.

3.3.1. L'intégration des briques SICORE

Comme nous l'avons vu, il s'agit du cas où l'adhérence de SICORE à l'application est la plus forte : le programmeur à qui il revient de finaliser cette intégration dispose :

- des fichiers contenant les déclarations de toutes les fonctions et constantes symboliques (par exemple pour identifier les codes-retour) nécessaires à l'utilisation de SICORE,
- des fichiers contenant les modules-objet nécessaires pour l'édition de liens de SICORE avec l'application,
- de la documentation d'intégration sous forme HyperText qui le guidera dans les étapes d'intégration.

Les briques correspondent aux fonctions classiques que nous avons déjà rencontrées, à savoir principalement la normalisation et le codage. Cependant, il faut en général se préoccuper du chargement en mémoire des bases de connaissances, chargement qui à l'exécution occasionnera une consommation en temps et en mémoire centrale[18]. A titre d'exemple,

---

[17] Il est possible d'écrire des applications en un autre langage, et qui font appel à des sous-programmes écrits en langage C, avec plus ou moins de difficultés suivant les plate-formes, mais au prix d'une technicité plus pointue.

[18] En effet, pour des raisons de performances, le choix a été fait de faire résider en mémoire toutes les données nécessaires aux traitements, évitant ainsi les lenteurs des périphériques d'entrée-sorties.

le chargement de la base de connaissances complète concernant la commune occupe 11 méga-octets en mémoire et demande 20 secondes sur un pentium 90, alors que ces chiffres sont respectivement de 8 méga-octets et 10 secondes pour la base complète de la profession. On aura donc intérêt à minimiser le nombre des chargements de bases de connaissances, autant que possible.

Il existe cependant un cas dans lequel on peut s'affranchir du chargement de bases de connaissances : dans le prolongement du projet SICORE, l'INSEE a décidé de développer un serveur de codage dans l'environnement du moniteur transactionnel CICS d'IBM. Le fonctionnement en est simple : une transaction serveur effectue le chargement d'une ou plusieurs bases de connaissances une fois pour toutes, les transactions-clientes programmées par le service-utilisateur[19] interrogent directement ce serveur sans avoir à réaliser les chargements de données. Ce serveur n'est pour l'instant disponible que dans le monde MVS, mais permet cependant de répartir les bases de connaissances sur plusieurs noyaux CICS interconnectés, de sorte qu'une transaction-cliente s'exécutant sur un noyau peut très bien interroger une base qui se trouve en fait sur une autre machine MVS, et ce sans le savoir puisque c'est SICORE qui prend intégralement à sa charge le routage de la requête et de la réponse.

[19] Là aussi avec l'aide d'une documentation en ligne.

3.3.2. L'intégration des traitements

Il s'agit dans ce cas de traiter des fichiers de données. Une étape consiste en la normalisation ou le codage d'un fichier pour une variable donnée. Comme précédemment, un chargement de base de connaissances est nécessaire, mais cette fois pour chaque fichier à traiter. L'avantage de cette solution est cependant la simplicité de mise en oeuvre puisqu'il suffit de peu d'informations pour réaliser les traitements (définitions de dessins pour déclarer la position des informations actives). Ces traitements sont en fait constitués de petits programmes appelant les briques SICORE de base. Du fait que ces programmes réalisent des opérations non-portables (prise en compte des paramètres et restitution de statistiques), ils ne sont pas portables (au contraire des briques) : il faut donc les développer pour chacune des plate-formes d'exécution. Cependant, d'une part leur simplicité les rend facile à développer (surtout avec l'aide de la documentation d'intégration), d'autre part il existe des versions déjà écrites dans le monde MVS[20].

3.4. Mise à disposition de SICORE

Chaque unité d'acquisition de SICORE comprend :

- l'*outil SICORE* (environnement PC),
- la documentation méthodologique détaillant toute la méthodologie SICORE actuelle, disponible à la fois sur papier et sous forme de documentation HyperText,
- le glossaire SICORE reprenant l'ensemble des termes propres à la méthodologie SICORE, disponible à la fois sur papier et sous forme de documentation HyperText,
- la documentation de l'utilisateur de l'outil SICORE pour manipuler correctement l'outil, disponible à la fois sur papier et sous forme de documentation HyperText,
- la documentation d'intégration de SICORE décrivant les modes d'intégration de SICORE dans des applications, disponible uniquement sous forme HyperText.

Sous réserve de négociation complémentaire avec l'INSEE, peuvent être proposés :

- une formation à l'outil SICORE,
- les bases de connaissances déjà réalisées par l'INSEE pour ses propres besoins, mais qui peuvent intéresser d'autres organismes,

[20] Sur PC, l'outil SICORE peut très bien réaliser cette tâche. Il n'est cependant pas souhaitable de confondre l'outil SICORE avec un outil de production. L'INSEE s'interroge actuellement sur l'opportunité de développer un outil bridé ne reprenant du poste de l'expert que les fonctionnalités de production.

- une aide à la constitution de bases de connaissances spécifiques à un organisme faisant l'acquisition de l'outil SICORE,
- la fourniture des modules de SICORE pour son intégration dans une application.