

GESTION MICROINFORMATIQUE SIMPLIFIEE DES FICHIERS D'INVENTAIRES ET LEURS ASSOCIES

Philippe Bourret
(Ichtyologie)
(UR 305)

Antenne ORSTOM
Muséum d'Histoire Naturelle
Paris

1. LES PETITS RIENS

Les taxinomistes, zoogéographes et écologistes fondent l'essentiel de leur travail sur des fichiers d'inventaires documentés dont la taille est très variable suivant qu'ils sont personnels ou collectifs, circonstanciels ou permanents. S'ils veulent simplement archiver des informations, les mettre à jour et les consulter de temps en temps, le bac à fiches reste pour la plupart de ces fichiers l'expédient le moins coûteux et le plus rapide. Mais dès que l'utilisateur envisage des traitements plus élaborés, tris statistiques, tableaux ordonnés, etc., constituer un fichier informatique, même restreint, devient à terme un gain de temps, malgré toute la rigueur et le formalisme qui forcément accompagnent cette opération.

Bien souvent l'utilisateur ORSTOM n'a accès aux Centres de calcul, et même aux minis, que pour des gros fichiers et des traitements spéciaux dont il n'a pas la maîtrise. Et seul devant son micro pas très sophistiqué (le "bilan de l'existant" en recensait déjà 250, il y en a maintenant au moins plus du double) il balance entre trois solutions :

- *acheter un logiciel tout fait* : mais les logiciels "grand public" sont soit trop simplistes pour satisfaire ses besoins particuliers, soit trop généraux pour éviter que son application ne se perde dans les méandres d'une adaptation complexe ;

- *faire programmer un logiciel sur mesure* : mais les informaticiens "maison" sont peu nombreux et le recours à une SSII extérieure trop cher. De toutes façons cette solution est laborieuse et même parfois décevante faute d'un dialogue de niveau ;

- *écrire son propre logiciel* : et là il recule devant l'investissement. Apprendre un langage ne demande guère plus qu'un peu de persévérance, et beaucoup ont suivi avec succès des cours de Basic, mais programmer, transformer des idées en ins-

tructions exécutable est autrement plus difficile et exigeant, mieux vaut en prendre l'exacte conscience avant de s'y engager.

2. POURQUOI POI ?

Pour traiter toutes sortes de fichiers de taille moyenne, depuis les catalogues de types d'environ 200 fiches, jusqu'à la "check-list" des poissons de Nouvelle-Calédonie qui en comporte plus de 1 600, en passant par la gestion des membres de la Société Française d'Ichtyologie, les bibliographies de thèses ou d'articles, etc., j'ai écrit en 1986 le programme *POI*, et par là voulu atteindre deux objectifs :

- éviter aux utilisateurs d'apprendre un langage ou un logiciel de gestion de fichiers. Ce qui signifie permettre à tous les membres d'une équipe travaillant sur les mêmes fichiers de les créer, les consulter, les mettre à jour, les corriger, les éditer, les transférer, etc., sans risquer d'intempestives et parfois fatales modifications de structure ou de contenu ;

- améliorer les performances du progiciel par une adaptation étroite aux besoins spécifiques (par exemple : déroulement de menus et d'explications libellés dans un vocabulaire familier, pas de codages malgré leurs indéniables avantages pour l'informaticien, linéarité des opérations les plus courantes, transparence des opérations complexes), par une programmation optimisée pour la rapidité d'exécution, et par une sécurité accrue (par exemple : validation des entrées, gestion conservatrice complète et transparente des index, sauvegardes automatiques).

Dans ce système, les utilisateurs déterminent eux-mêmes l'architecture de l'ensemble des fichiers mis en relation (par exemple : fichier d'espèces associés à un fichier de synonymes et un fichier bibliographique), et la structure de chacun de ces fichiers : ce qui s'acquiert d'expérience en suivant quelques règles de conduite très simples, et ne doit au demeurant pas faire naître d'excessives angoisses puisque les structures peuvent être modifiées à tout moment.

Le programme permet alors un très grand choix dans :

- le mode d'entrée des données, de consultation, de correction, etc. ;

- la forme des sorties, qui peuvent être directes sur diverses imprimantes ou sur écran, ou bien indirectes sur des traitements de texte et des tableurs, puisque le programme crée des fichiers "ASCII séparateur".

3. LA LANGUE DE POI

Les "moulinettes" Basic sont certes des outils pratiquement irremplaçables pour transcoder des séquences de caractères, changer des formats de fichiers, transférer des données, etc., mais le Basic interprété n'est pas bien structuré, d'écriture malcommode, et surtout, comme d'ailleurs les langages compilés tel Pascal, il rend les programmes et les fichiers étroitement interdépendants.

Les générateurs d'applications, Multigen, Magic, Yes You Can ou Dialogue II, qui réalisent le logiciel dès lors qu'on leur a précisé la structure clavier-écran, permettent d'obtenir assez rapidement des applications qui "tourment". Mais répondre aux questions posées par ces générateurs est aussi difficile qu'une programmation classique, et la longueur du logiciel final rend son exécution extrêmement lente.

Il m'a donc paru préférable de recourir à un logiciel paramétrable qui gère complètement les données sur les disques et comporte un langage procédural assez complet.

POI est un programme d'environ 380Koctets, suivant ses accessoires, écrit en dBASE II version 2.4, et disponible sous CP/M et MS-DOS, version 3.0 (en cours une version dBASE III, compilée par Nantucket, alias Clipper, pour accroître sa vitesse et délivrer des versions "run-time"). Les limites des fichiers sont les suivantes :

- nombre de champs : 32 (en entrée simultanée 23 seulement) ;

- taille maximale d'une fiche : 1 000 caractères ;

- nombre théorique de fiches : 65 536 ;

- champs numériques : entiers de 0 à 999.

Son menu initial ci-dessous donne un aperçu des principales opérations qu'il réalise, avec la meilleure adéquation sur les fichiers d'espèces et leurs associés, mais en fait aussi avec une grande efficacité sur une gamme très large de fichiers.

SORTIE (et sauvegarde)

Nouvelles ENTREES dans le fichier

RECHERCHES et CORRECTIONS

INDEXATIONS du fichier

ADDITION de taxons supérieurs

LISTES, SOUS-FICHIERS, FICHIERS TEXTES

INDEX GENERAL

Consultation/mise à jour FICHER de SYNONYMES

CHANGEMENT de fichier de travail

AJOUT d'un fichier au fichier de travail

Qu'il me soit permis de tirer de cette expérience de programmation, et de la co-direction d'un projet de gestionnaire de données bibliographiques, au moins deux observations :

- la définition initiale des spécifications d'un logiciel aboutit à une sorte de cahier des charges très approximatif, qui au gré des demandes peut se transformer rapidement en une carcasse inutile ou monstrueuse, si l'on ne veille à s'imposer de sévères limitations. La définition des besoins doit nécessairement être un processus itératif, auquel il faut donner une sortie, voire un compromis, car hélas on peut tout programmer ;

- seule une étude très approfondie des performances du progiciel, notamment aux conditions limites, révèle ses "bugs", ses réactions inattendues, parfois aléatoires, et surtout ses lenteurs. C'est le prix élevé à payer pour optimiser la vitesse de procédures complexes sur des fichiers lourds, souvent d'ailleurs au moyen de solutions informatiques franchement atypiques.

Le logiciel *POI* est à la disposition des utilisateurs potentiels : il est fourni avec une documentation très complète (soixante-dix pages) qui permet une prise en main immédiate.