

dans \mathbb{R} , qui associe a à 0, b à 1 et c à 2, alors la latitude est le réel associé par cette fonction à 0, la longitude celui associé à 1 et l'altitude celui associé à 2. Ici encore, il y a beaucoup d'autres possibilités, et donc beaucoup d'arbitraire dans ce choix.

Il n'y a donc pas vraiment de raison d'associer à chacune de ces valeurs une place dans un n -uplet plutôt qu'une autre, ou un nombre plutôt qu'un autre. De plus, si dans un programme, on change un peu une structure de données en lui ajoutant un champ, alors il faut décaler tous les autres, ce qui demande de modifier le programme à plusieurs endroits. Ces modifications sont à l'origine de nombreuses erreurs dans les programmes, dans lesquels on finit par confondre la longitude avec la température...

Puisque les programmeurs préfèrent identifier les champs d'une structure par un nom — « latitude », « longitude », ... — et non par une place ou un numéro, les langages de programmation doivent le leur permettre. Cela amène à définir une notion de n -uplet à champs nommés, que l'on appelle *enregistrement*. D'un point de vue mathématique, un enregistrement est une fonction dont le domaine est, non un segment initial de \mathbb{N} , mais un ensemble fini quelconque, dont les éléments sont appelées les *étiquettes* de cet enregistrement.

Cette idée de ne pas indiquer les champs par leur place dans un n -uplet, mais par un nom peut s'appliquer aussi aux arguments d'une fonction. Dans certains langages expérimentaux, on n'écrit pas $f(4,2)$ mais $f(\text{abscisse} = 4, \text{ordonnee} = 2)$ et il est équivalent d'écrire $f(\text{ordonnee} = 2, \text{abscisse} = 4)$.

8.1.2 Une extension de PCF avec des enregistrements

Pour définir une extension de PCF avec des enregistrements, on ajoute trois symboles à PCF : un symbole $\{\}$ qui permet de construire un enregistrement, un symbole $.$ qui permet d'accéder au champ d'un enregistrement et un symbole $<-$ qui permet de construire un nouvel enregistrement identique à un enregistrement déjà construit, sauf en un champ unique.

Avant d'introduire ces symboles, on doit introduire une nouvelle sorte pour les étiquettes ainsi qu'un nombre infini de constantes : une pour chaque étiquette. On peut remarquer qu'il n'y a pas de symbole permettant de lier une variable de sorte *étiquette* et donc qu'il n'y a pas de telles variables dans un terme clos. De plus, le langage ne fournit pas d'autres symboles que les constantes pour construire des termes de sorte *étiquette*. De ce fait, dans un terme clos, les seuls termes de sorte *étiquette* sont des constantes. On peut alors ajouter à PCF

- un symbole $\{\}$ à $2n$ arguments, dont les arguments de rang impair sont des étiquettes et les arguments de rang pair des termes, et qui ne lie aucune variable,
- un symbole $.$ à deux arguments dont le premier est un terme et le second une étiquette, et qui ne lie aucune variable,
- un symbole $<-$ à trois arguments dont le premier est un terme, le deuxième une étiquette et le troisième un terme et qui ne lie aucune variable.