

# Langages de description et langages d'action

Gilles Dowek

# Langues et langages



Nous nous exprimons en utilisant deux outils : les langues et les langages qui ont des histoires **parallèles**

Langue :

Longtemps, je me suis couché de bonne heure.

Langage :

```
let rec fact n = if n = 0 then 1 else n * (fact (n - 1))
```

# Plus de différences que de points communs

Langues :

- ▶ qui a inventé l'italien ?
- ▶ unidimensionnelles (linéaires)
- ▶ doublement articulées
- ▶ orales ou écrites

Langages :

- ▶ créés de toutes pièces
- ▶ uni- ou multidimensionnels
- ▶ doublement articulés ou non :  $\text{p}\dot{\text{i}}$ , mais  $\pi$ , fact, mais !
- ▶ écrits

# Explosion aux XX<sup>e</sup> et XXI<sup>e</sup> siècles

Langages de programmation  
plus généralement informatiques

Mais aussi logique **formelle**

Mais avant cela...



$$x^3 + 3x^2 = 20$$

$$2^3 + 3 \times 2^2 = 20$$



21, XXI, **OOD**

# Les langages des sciences

Toutes les sciences utilisent des langages

Mais certaines en créent



et d'autres en empruntent

$$y = -\frac{1}{2}gt^2$$

# Langage de description

Dans les langues :

- ▶ Sigebert est vivant, Sigebert est mort
- ▶ Sigebert meurt
- ▶ Clodéric tue Sigebert

**Mais** dans le langage de la logique des prédicats

$$3 \leq 4$$

Pourquoi ?

- (1) les propositions d'état sont les plus fondamentales
- (2) le but est de formaliser le langage mathématique : pas de transitions en mathématiques

## Pas de transitions en mathématiques ? !

Et  $\ddot{y} = -g$

Et  $y = -\frac{1}{2}gt^2$

Et les systèmes dynamiques ? Et les processus de Markov ?

Oui, mais la fonction  $t \mapsto -\frac{1}{2}gt^2$  est continue, décroissante, dérivable... aujourd'hui, **et elle le sera encore demain**

Les objets mathématiques décrivent le mouvement, mais eux sont immobiles



# Les langages d'action reviennent par la fenêtre

Les premiers langages de programmation  
(grande candeur des pionniers de l'informatique)

Un programme exprime ce que l'ordinateur doit faire

Des phrases à l'impératif

Un abîme sépare

$$z = x + y + y$$

de

$$z := x + y + y$$

## Une deuxième révolution galiléenne

Quelle équation, quelle proposition, **relie** le brin d'ARN messenger à la protéine ?

```
let rec debut l = match l with
  | A::U::G::_ -> l
  | _::r -> (debut r)
  | [] -> []
let rec traduction l = match l with
  | x::y::z::r ->
    if List.mem (x,y,z) stop
    then []
    else (List.assoc (x,y,z) table)::(traduction r)
  | _ -> []
let arn_en_proteine l = traduction (debut l)
```

**Décrire est inférer** : des lois de Newton à la prévision des éclipses

# Aujourd'hui

Diverses manières de décrire le monde :

- ▶ langage des équations différentielles
- ▶ autres langages mathématiques
- ▶ langages de programmation
- ▶ des “langages” dont les expressions sont des données massives

Mais surtout : on ne picore plus dans un corpus fixe (les équations différentielles, les équations chimiques, Fortran)

Au lieu d'utiliser un langage inventé par d'autres : **créez le vôtre**