

## Le tableur et l'option informatique: vers la programmation par objets

Raymond Favre-Nicolin

► **To cite this version:**

Raymond Favre-Nicolin. Le tableur et l'option informatique: vers la programmation par objets. Georges-Louis Baron, Jacques Baudé, Philippe Cornu. Colloque francophone sur la didactique de l'informatique, Sep 1988, Paris, France. Association EPI, pp.283-297, 1989, <ISSN: 0758-590 X; <http://www.epi.asso.fr/association/dossiers/d07som.htm>>. <edutice-00373776>

**HAL Id: edutice-00373776**

**<https://edutice.archives-ouvertes.fr/edutice-00373776>**

Submitted on 7 Apr 2009

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

**LE TABLEUR ET L'OPTION INFORMATIQUE**  
**vers la programmation par objets**

**Raymond Favre-Nicolin**

**Lycée Vaucanson, 27 rue Anatole France,  
38030 Grenoble Cédex**

## SYNTHESE

Des difficultés décelées auprès de plusieurs groupes d'élèves dans l'enseignement optionnel de l'informatique en classe de seconde m'ont amené à élaborer puis à affiner par étapes successives une approche particulière du programme. Cette communication présente l'état d'avancement de ma réflexion à ce sujet au cours de l'année scolaire 1987/88.

Dans la première partie de l'année scolaire les élèves apprennent à identifier dans un problème les objets significatifs, à reconnaître a priori le type de données informatiques qui permettra le codage de ces objets (valeurs logiques, chaînes de caractères, nombres, objets composés) et la nature de la définition de ces objets (définition simple, définition conditionnelle, définition itérative). La découverte de ces notions s'effectue progressivement, les objets à définition itérative qui constituent l'introduction à l'algorithmique séquentielle classique ne sont abordés que lorsque la notion d'objet est maîtrisée par un nombre suffisant d'élèves.

Cette démarche englobe à la fois le cours théorique, les exercices et les travaux pratiques sur ordinateurs. Son originalité principale est de s'appuyer pour les travaux pratiques sur le tableur Multiplan, utilisé dans un contexte de résolution de problème, de simulation et d'aide à la décision au cours de la première moitié de l'année scolaire. La force du tableur par opposition à un langage traditionnel résulte :

- de la visualisation permanente de l'état des données (des objets),
- de la possibilité de repousser les difficultés liées à la notion de séquence tout en préparant son introduction et son étude systématique,
- de la facilité de la prise en main.

Les résultats obtenus par des élèves lors d'un exercice d'analyse et lors de deux travaux pratiques sont commentés à titre d'illustration et servent de base à des comparaisons entre le tableur d'une part, et d'autre part les langages impératifs, les langages applicatifs et les langages à objets.

## MOTS CLES

tableur - Multiplan - programmation par objets - didactique - résolution de problème - option informatique

## LE TABLEUR ET L'OPTION INFORMATIQUE vers la programmation par objets

Raymond Favre-Nicolin

Lycée Vaucanson, 27 rue Anatole France,  
38030 Grenoble Cédex

### I. INTRODUCTION

**1- Le contexte** : en seconde, au lycée technique Vaucanson, 50% des élèves s'inscrivent à l'enseignement optionnel d'informatique qui comprend une heure de cours et une heure et demie de travaux pratiques hebdomadaires. Il s'adresse aux élèves qui souhaitent approfondir les méthodes mises en oeuvre dans les logiciels et les systèmes qu'ils rencontrent dans les enseignements professionnels et plus généralement à ceux qui ressentent le besoin d'une culture générale dans le domaine de l'informatique. Expérimentée depuis plusieurs années, plusieurs fois remaniée, la démarche présentée s'est progressivement affinée et débouche actuellement sur l'élaboration d'éléments de didactique.

Afin d'exposer la méthode employée nous avons sélectionné deux sujets traités en travaux pratiques et un exercice résolu par des élèves de seconde au cours du premier semestre.

**2- Le constat de départ** : nous avons décelé chez nos élèves des difficultés diverses et dans le cadre de cet exposé nous nous arrêtons plus particulièrement sur deux d'entre elles. Elles sont rencontrées lorsqu'il s'agit :

- d'exprimer un algorithme séquentiel simple,
- d'identifier ce qui relève de la logique dans un problème.

Tous les enseignants d'informatique sont confrontés à la première difficulté. Elle est signalée de façon universelle, à tous les niveaux, lors de la première année d'enseignement de la matière pour les élèves. Elle a déjà fait l'objet de nombreuses recherches. Nous n'insisterons donc pas sur cet aspect, signalons toutefois dès maintenant que la méthode que nous proposons apporte un élément de solution.

La deuxième difficulté est plus rarement signalée. Cela provient selon nous en grande partie du contenu de l'enseignement dispensé qui ne la met pas en évidence directement mais seulement lors de l'introduction des schémas de contrôle. Par exemple le

programme de l'option informatique en classe de seconde prévoit dans les exemples de données simples à présenter aux élèves "les nombres, les caractères, les chaînes" mais pas les valeurs logiques. Le mot "logique" apparaît dans le chapitre "traitement conditionnel" et uniquement sous les formes "expression logique élémentaire" et "expression logique générale avec connecteurs logiques". Dans ces conditions il est normal que les élèves ne considèrent pas la logique comme un monde d'objets au même titre que celui des nombres par exemple. Les "expressions logiques" sont alors conçues simplement comme éléments permettant le bon déroulement du programme et leurs résultats sont rarement explicités. On peut être étonné de rencontrer cette difficulté dans notre établissement où les élèves reçoivent un enseignement professionnel qui précisément comporte l'utilisation de circuits logiques, mais il s'agit là simplement d'une conséquence du cloisonnement de l'enseignement technique en disciplines étroitement spécialisées. Cette deuxième difficulté peut apparaître en première analyse comme plus anodine et plus facile à surmonter que la première. Elle est cependant significative du statut de la pratique de la programmation dans un langage impératif par rapport à l'analyse algorithmique.

La programmation impérative montre en permanence à l'élève des suites d'actions alors qu'on lui demande lors de l'analyse algorithmique d'établir un raisonnement à partir d'une suite d'états. Cette contradiction entre cours et travaux pratiques participe au rejet de l'algorithmique par certains élèves.

**3- Démarche :** les objets logiques (comme les autres) doivent être observés et manipulés jusqu'à constitution d'une représentation mentale. Nous avons dans un premier temps remédié à la deuxième difficulté signalée par une mise en évidence et une explicitation des objets de nature logique, en nous appuyant pour la mise en oeuvre en travaux pratiques sur le tableur Multiplan en début d'année scolaire. L'apport du tableur par rapport aux langages impératifs ou applicatifs sera discuté plus loin.

Une systématisation de cette démarche m'a conduit à rejoindre un courant important de l'informatique théorique, la programmation par objets. Les exercices pratiqués sur le tableur constituent une bonne approche de ce mode de raisonnement. L'analyse d'un problème donné commence toujours par une reconnaissance explicite d'objets ainsi que par la mise en évidence de la structure, du type et de la nature de la définition de ces objets. Au niveau de la classe de seconde nous introduisons ainsi successivement par ordre de complexité les valeurs logiques, les caractères, les chaînes de caractères, les nombres en ne détaillant que le cas des nombres entiers. Les objets étudiés sont d'abord à définition simple, puis à définition conditionnelle. La programmation séquentielle (accompagnée par des travaux pratiques en langage Pascal) n'est abordée que lorsque la notion d'objet est suffisamment comprise et pratiquée "naturellement" par un nombre suffisant d'élèves. La transition s'effectue en explicitant dans des exercices et des travaux pratiques la notion d'état de la mémoire.

A partir des premières conclusions sur ces expériences nous nous proposons d'inclure une initiation à une méthodologie de classement des informations qui pourrait s'appuyer sur les techniques documentaires, l'objectif essentiel étant de faciliter la spécification informatique d'un problème.

## II. PREMIER EXEMPLE : JEU LOGIQUE

**1- Premier contact** : lors de la première séance de travaux pratiques de l'année les élèves prennent en main le tableur Multiplan en réalisant chacun la traditionnelle fiche d'identification destinée au professeur. Les premiers cours théoriques sont consacrés à la présentation de la notion d'information puis à celle des données simples et à leur codage dans l'ordinateur sur lequel nous insistons. Nous commençons par les données logiques en nous appuyant sur des exemples de propositions logiques auxquelles il est facile pour les élèves d'attribuer l'une des deux valeurs VRAI ou FAUX. Pour illustrer le cours, la deuxième séance de travaux pratiques commence par la lecture d'une petite énigme logique extraite du livre de R. Smullyan "Le livre qui rend fou".

**2-Sujet** : le sujet suivant est distribué aux élèves en séance de travaux pratiques :

### INFORMATIQUE - SECONDE - TP N° 2

L'histoire suivante est extraite du livre "Le livre qui rend fou" de R. Smullyan (éd. Dunod) :

"... le roi l'entraîna vers les deux cellules et lui montra les affiches qu'il avait lui-même collées sur les portes.

1
Il y a une princesse dans cette cellule et un tigre dans l'autre

2
Il y a une princesse dans une cellule et il y a un tigre dans une cellule

« Dois-je faire confiance à ce qui est écrit ? » questionna encore le prisonnier.

« Une des affiches dit la vérité, promet le roi, et l'autre ment. »

A la place du prisonnier, quelle cellule auriez-vous choisie ? "

Construire une feuille de calculs avec Multiplan qui simule le problème ci-dessus et puisse servir d'aide à la décision. La feuille de calculs sera sauveée dans un fichier sur la disquette. La note du TP sera attribuée en fonction de ce fichier (fonctionnement, présentation, facilité d'emploi).

**3- Analyse** : la discussion initiale porte spontanément au départ sur le problème lui-même. L'enseignant n'apporte pas de solution, il n'indique pas non plus si les solutions proposées sont justes ou fausses. Il est rapidement nécessaire d'inciter les élèves à pratiquer une analyse constructive s'appuyant sur l'identification des objets concrets du problème, lequel a été choisi précisément pour les rendre d'emblée évidents. Le professeur insiste ainsi sur :

- la nécessité du codage (il n'y a ni tigre ni princesse dans l'ordinateur !),
- la nature des propositions logiques qui figurent sur les affiches,
- l'aspect "aide à la décision".

**4- Programmation** : les commentaires qui suivent s'appuient sur une feuille de calculs effectivement réalisée par des élèves. Au sujet du codage, je recommande de nommer les cellules utilisées pour définir ce codage et d'étiqueter ces cellules. Ce qui donne par exemple :

	colonne 1	colonne 2	
ligne 8	tigre :	0	cellule L8C2 nommée : tigre
ligne 9	princesse:	1	cellule L9C2 nommée : princesse

La colonne 1 contient les étiquettes qui permettent la "lecture" de la feuille de calculs. La cellule **L8C2**, située à l'intersection de la ligne 8 et de la colonne 2, est nommée tigre. Elle contient le nombre 0 qui constitue le code (interne) correspondant à l'objet tigre. De même en **L9C2** l'objet princesse est codé par le nombre 1. Si l'on désire comparer la démarche suivie à celle pratiquée à l'aide d'un langage classique, le lexique ainsi construit constitue également une déclaration des "variables", ce qui prépare la suite du cours.

Les cellules de la prison sont représentées par deux cellules du tableur :

	1	2	3
11	représentation des cellules :		
12	-----		
13		cellule 1:	0
14		cellule 2:	1

La cellule 1 située en ligne 13, colonne 3, nommée cellule1, contient la formule :

cellule **L13C3** : tigre

Ce qui se traduit par l'affichage du nombre 0, résultat de l'évaluation de cette formule compte tenu du codage défini plus haut. Dans l'exemple présenté, les élèves font l'hypothèse (arbitraire a priori) que la cellule 1 contient un tigre et la cellule 2 une princesse. Les propositions logiques écrites sur les affiches sont également traduites, par exemple :

	1	2	3	4	5
16	pancarte 1:		cellule 1		cellule 2
17	-----		FAUX		FAUX

Voici les formules qui traduisent simplement les informations apportées par l'affiche posée sur la cellule 1 en ce qui concerne le contenu des cellules dans l'extrait ci-dessus :

cellule **L17C3** : cellule1=princesse

cellule **L17C5** : cellule2=tigre

La deuxième affiche, les paroles du roi, constituent autant de propositions logiques susceptibles d'être traduites informatiquement. On remarquera sur l'exemple présenté ci-dessus l'indépendance du programme vis-à-vis du codage numérique interne.

Selon les classes, selon la technique employée plus ou moins directive, selon que l'on affine plus ou moins la feuille de calculs réalisée, deux à quatre séances de travaux pratiques sont consacrées à ce sujet. La simulation du problème est poussée plus ou moins loin selon les groupes. A ce niveau de l'année, les fonctions logiques (OU, ET, NON) n'ont pas été abordées dans l'enseignement d'informatique. Ces fonctions ont cependant déjà été étudiées par certains groupes dans les disciplines technologiques et quelques élèves les ont alors utilisées. Les jeux logiques susceptibles d'être abordés selon l'approche décrite ici ne manquent pas. Je pense qu'il ne faut pas choisir des jeux trop complexes, cependant il est peut-être possible de proposer plusieurs sujets de difficultés graduées. Je critique également les sujets proposés qui conduisent à réaliser des tableaux (du genre tableaux de Karnaugh) car ils laissent trop peu d'initiative à l'élève.

**4- Compte-rendu** : les élèves rendent en guise de copie un fichier sur disquette. Les critères retenus pour l'attribution de la note portent sur l'exactitude et la simplicité de la solution, la documentation (les commentaires doivent être suffisamment explicites pour que l'organisation de la feuille de calculs soit compréhensible), la présentation et le respect de quelques règles. En ce qui concerne ce premier TP, je ne tiens pas compte de la



méthode de travail employée (rationalité de la démarche ou bien tâtonnements, temps total passé).

### III. DEVELOPPEMENT DE LA METHODE

**1- Le cours** : la suite du cours comporte des notions sur la numération, ce qui donne l'occasion de revenir sur le codage nécessaire de toute donnée. Les élèves sont progressivement entraînés à analyser des problèmes qui comportent des objets plus complexes et moins évidents.

**2- Exercices** : Voici, à titre d'exemple, le corrigé d'une question d'un devoir, rédigé par des élèves travaillant en groupe :

OBJETS : dans le texte suivant :

"En fonction de leur moyenne, les élèves sont classés en groupes, groupe A pour ceux dont la moyenne est supérieure ou égale à 13, B si elle est comprise entre 10 et 13, etc"

nous identifions les objets MOYENNE, ELEVE et GROUPE. Ces objets ne sont pas indépendants, car le groupe est défini à partir de la moyenne, elle-même propre à un élève. L'objet ELEVE est ainsi un objet composé par :

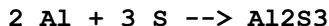
- son NOM : objet simple codé par une chaîne de caractères
- sa MOYENNE : objet simple codé par un nombre décimal
- son GROUPE : objet à définition conditionnelle :
  - 1er cas : si sa moyenne est supérieure ou égale à 13, le groupe est un objet simple codé par le caractère "A"
  - 2ème cas : si sa moyenne est comprise entre 10 et 13, le groupe est un objet simple codé par le caractère "B"
  - 3ème cas : si sa moyenne est inférieure à 10, le groupe est un objet simple codé par le caractère "C"

L'exemple précédent montre comment j'aborde le problème de l'analyse préalable, comment je l'introduis à la fois comme une nécessité que j'impose, qui facilite la mise en oeuvre et qui sera pratiquée même lorsqu'on n'utilisera plus le tableur, et toujours à partir d'exemples concrets. Il s'agit bien sûr de préparer les élèves à raisonner sur des états.

### IV. DEUXIEME EXEMPLE : PROBLEME DE CHIMIE

**1- Sujet** : l'exercice est proposé en janvier ou février. Cette année il sera le dernier traité à l'aide du tableur. Le problème est extrait du livre de chimie de seconde, de R. Didier et R. Vento, (Armand Colin) :

Dans la réaction



quel est, parmi les mélanges suivants, celui qui est stoechiométrique ? Pour les mélanges non stoechiométriques, indiquer le corps en excès.

a/ 10,8 g Al + 19,2 g S

b/ 6 g Al + 9,6 g S

c/ 10 g Al + 11 g S

TRAVAIL DEMANDE : Résoudre le problème ci-dessus à l'aide d'une feuille de calculs Multiplan.

CONSEILS DE REDACTION :

- prévoir l'organisation de la feuille de calculs en s'inspirant des travaux pratiques précédents
- utiliser cette feuille de calculs pour résoudre d'autres problèmes du même type, par exemple pour d'autres réactions chimiques de même nature.

2- **Exemple de feuille de calculs** : voici quelques extraits d'une feuille de calculs significative des résultats obtenus au cours de l'année scolaire 1987/88. Tout d'abord les objets "Premier réactif" et "Deuxième réactif" sont identifiés et traduits ainsi :

1	2	3	4	5	6	7
1 NNNNNNNN	Pppppppp		TP N°5	2T7		
2 NNNNNNNN	Ppppp					
3 =====		Le 5 février 1988				
4 Premier réactif				Deuxième réactif		
5 NOM:	ALUMINIUM			NOM:	SOUFRE	
6 SYMBOLE:	Al			SYMBOLE:	S	
7 MASSE MOLLAIRE:		27		MASSE MOLLAIRE:		32
8 Nb MOLES:		2		Nb MOLES:		3

L'objet "Premier réactif" est compris comme un objet complexe, caractérisé par son nom, son symbole chimique, sa masse molaire, le nombre de moles de ce corps mises en jeu dans la réaction. Le codage de chacun de ces objets élémentaires est naturel (chaîne de caractères pour les deux premiers, nombres entiers pour les deux derniers). Le fait marquant est la reconnaissance de l'entité "Premier réactif" comme un objet qui se code facilement par un ensemble de données élémentaires et que l'on a intérêt à considérer comme un tout, représentant une entité caractéristique du problème posé. Dans la feuille présentée comme exemple, l'accès à chaque champ est obtenu en nommant individuellement les objets élémentaires, la cellule L5C3, par exemple, est nommée NOM1. D'autres groupes d'élèves ont utilisé un accès indexé. Il est possible en effet de

nommer globalement l'objet "Premier réactif" situé en colonne 3, ligne 5 à 8 et d'obtenir individuellement chaque objet élémentaire qui le compose à l'aide de la fonction INDEX de Multiplan.

A partir de ces données il est possible d'obtenir la réaction chimique par une formule qui met en oeuvre des concaténations de chaînes de caractères :

10 REACTION:	2A1 + 3S ----> A12S3
--------------	----------------------

Une zone d'entrée des données numériques particulières est prévue :

12	ENTREE DES MASSES DES REACTIFS		
13 ALUMINIUM	10,8	SOUFRE	19,2

Dans la reproduction ci-dessus le mot "ALUMINIUM" est obtenu par la formule NOM1 qui désigne l'objet correspondant tel qu'il est défini dans la zone "Premier réactif". Le paramétrage de la feuille de calculs est donc assez poussé et permet la résolution de tout un ensemble de problèmes semblables concernant d'autres réactions chimiques.

Je ne reproduis pas la zone qui contient des calculs intermédiaires, voici celle contenant la réponse :

27 le mélange est-il stoechiometrique?:	VRAI
28	
29 CORPS EN EXCES:	
30 IL N'Y A AUCUN CORPS EN EXCES	

La réponse en ligne 30 est un objet de type chaîne de caractères à définition conditionnelle. La "condition" dépend notamment de la stoechiométrie du mélange proposé, codée par une valeur logique dans la ligne 27 ainsi que d'autres objets situés dans la zone des calculs intermédiaires qui servent à indiquer le corps éventuellement en excès.

**3- Remarques :** les reproductions ci-dessus ne montrent ni la dynamique de la construction si importante en pédagogie, ni le côté "hypertexte" de la "copie" obtenue, qui résulte de la bidimensionnalité du tableur et des différentes couches : formules, attributs de visualisation, valeurs affichées sur l'écran.

## V. VERS LA PROGRAMMATION PAR OBJETS

**1- Tableur et langages impératifs :** la programmation impérative classique s'appuie sur la séquence, étroitement liée à la notion d'affectation, et sur les schémas de contrôle. Pour

être bien comprise, une séquence doit se concevoir globalement, par examen des états successifs des données, en privilégiant l'état initial et l'état final. Cela suppose des facultés de mémorisation, d'abstraction, d'organisation, d'oubli des détails qui, actuellement, n'ont pas été développées auparavant chez la plupart de nos élèves.

La programmation impérative n'offre pas la possibilité d'aborder les difficultés séparément.

Son utilisation en travaux pratiques au début de l'enseignement de l'informatique impose à l'élève d'assimiler à la fois la séquence et l'affectation, autrement dit il est confronté d'emblée, d'une part aux structures de données, d'autre part aux algorithmes séquentiels. Dans le même ordre d'idées, la distance est grande entre les TP où il construit sur son écran des suites d'ordres et le cours théorique - même s'il est conçu à partir d'exemples concrets - où, en phase d'analyse, on lui impose de raisonner sur des états qu'il ne voit jamais concrètement. Pour les très bons élèves cela ne pose pas de problèmes mais cela limite la culture informatique à trop peu d'entre eux, ceux qui s'orientent vers des sections à dominante théorique.

D'où l'idée d'utiliser le tableur dans la première phase d'apprentissage afin de présenter les points difficiles séparément. Un autre avantage non négligeable du tableur est dû au fait que les problèmes d'entrées/sorties sont résolus automatiquement sans devoir être explicités au préalable.

Cependant la partie programmation du programme d'enseignement optionnel d'informatique de la classe de seconde est actuellement axée vers la maîtrise de la construction de séquences de traitement correctes. Le respect de ce programme impose donc de fait une pratique de programmation de style impératif quel que soit le langage utilisé.

Voici comment je situe le tableur dans cette approche : le tableur montre en permanence l'état (les valeurs) des objets. On peut oublier provisoirement la séquence. Je fais donc travailler d'abord les élèves sur le tableur. De bonnes habitudes doivent être prises dans cette phase, tout d'abord l'assimilation des données informatiques de base : données logiques, nombres, chaînes de caractères. Par la suite des objets plus complexes sont introduits, ce qui permet d'insister sur la phase de codage. Celle-ci est trop souvent survolée parce qu'elle est jugée naturelle lorsque l'on ne travaille que sur des nombres ou sur des textes. Elle est aussi fréquemment absente lorsque l'on ne dispose pas de l'outil permettant sa mise en évidence de façon aisée. Le tableur est excellent pour aborder cette partie du programme et il permet ainsi la mise en place des briques élémentaires solides à partir desquelles un savoir pourra par la suite être construit. L'organisation des feuilles de calculs mériterait également une mention spéciale, cependant je pense que l'expérimentation pédagogique à ce sujet doit être poursuivie avant que l'on puisse émettre quelques conclusions. Dans l'état actuel de ma réflexion je pense qu'il ne faut pas à ce sujet imposer trop de règles.

Notons enfin que la séquentialité n'est pas absente totalement, mais qu'elle se restreint à celle de la mise en oeuvre, avec des incidences sur les capacités acquises par les élèves dans l'organisation de leur travail personnel qui se concrétisent en phase de développement des projets qui leur sont proposés en fin d'année.

J'aborde le traitement séquentiel seulement après au moins trois mois de pratique sur le tableur. La transition nécessite un soin particulier que je ne développerai pas spécialement ici et qui s'appuie sur des exercices de réflexion portant sur l'explicitation d'états successifs de la mémoire d'un ordinateur. Cette année les travaux pratiques ont été réalisés en Pascal. La méthode employée prolonge ce qui précède : les élèves identifient dans un énoncé les objets nécessaires a priori à la représentation du problème, recherchent le codage qui convient, la nature de la définition de ces objets avant tout essai de traitement algorithmique.

Pour conclure je dirai que la complémentarité de la pratique d'un tableur et d'une programmation de style impératif me paraît évidente à la lumière de l'expérience décrite ici, l'objectif étant de suivre au plus près le programme de l'enseignement optionnel de l'informatique en classe de seconde, tout en s'efforçant de le rendre accessible au plus grand nombre possible d'élèves de seconde technique.

**2- Tableur et langages applicatifs** : les langages applicatifs offrent l'avantage de favoriser la programmation fonctionnelle. La démarche de spécification d'objets est alors praticable, notamment en faisant appel aux "listes de propriétés". Le progrès par rapport à la programmation impérative est considérable sur le plan conceptuel. L'apport principal du tableur dans l'optique de la programmation fonctionnelle est de rapprocher l'image en construction sur l'écran de la perception que les élèves ont du problème. Avec un langage applicatif, tout se passe dans la tête, l'expression et la communication sont purement textuelles. Avec le tableur on voit sur l'écran les objets que l'on construit, c'est bien plus facile lors d'une première approche et tout aussi performant quant aux notions acquises. J'ai eu l'occasion d'associer le tableur à LOGO, également dans l'enseignement optionnel en classe de seconde [11]. Des conditions matérielles défavorables ont empêché la poursuite de cette expérience mais il est certain que sa portée dépasse le cadre de l'enseignement optionnel tel qu'il existe actuellement. Je veux dire par là que l'association tableur-langage applicatif mériterait une étude approfondie susceptible de déboucher sur une remise en cause plus large d'idées admises actuellement sur l'introduction de l'informatique de la sixième à la terminale. Cependant il serait peut-être plus rentable d'orienter de telles recherches en utilisant comme support un langage à objets.

**3- Tableur et langages à objets** : ce style de programmation s'appuie sur des objets, c'est-à-dire des entités décrites par des attributs et des méthodes et qui communiquent par des messages. Les langages à objets ne sont effectivement disponibles en France sur micro-ordinateur que depuis peu, citons seulement Smalltalk [3]. Cette nouvelle étape de la

révolution informatique pourrait bien porter en elle un nouveau paradigme susceptible d'entraîner, dans les dix prochaines années, une remise en cause méthodologique comparable à celle apportée par la programmation structurée vers 1970. Les langages à objets possèdent des qualités propres qui peuvent s'évaluer par comparaison avec les autres styles de programmation, programmation fonctionnelle et programmation en logique notamment [1] [2].

Je considère la cellule élémentaire du tableur comme un objet, dont l'état (attribut) est défini par une formule (méthode). Cela convient pour le codage d'objets simples, réalisable à l'aide des objets de base disponibles. L'exemple du jeu logique présenté dans cet article illustre ce concept. Des ensembles organisés de cellules peuvent constituer des objets plus complexes tels qu'on peut les voir dans l'exemple du problème de chimie. Le tableur apparaît ainsi avec un statut nouveau qui en renouvelle l'intérêt, permettant une ouverture vers des concepts de la programmation moderne tout en préservant une certaine simplicité pour l'approche de notions qui sans lui resteraient trop théoriques pour certains publics d'élèves.

**4- Modélisation des concepts** : l'analyse (la spécification) par objets s'applique aussi bien à la description d'une situation nouvelle pour l'élève qu'à la résolution plus traditionnelle d'un problème scolaire fermé. Il nous semble nécessaire d'introduire à ce niveau des pratiques rationnelles héritées des techniques documentaires [13] [14]. C'est non seulement une question d'efficacité immédiate mais aussi une évolution qui tient compte de la quantité d'informations sans cesse croissante à la disposition des élèves. Ils doivent trier l'essentiel, distinguer ce qui se rapporte à leur problème, hiérarchiser les entités retenues.

Il leur faut une méthode pour cela. L'analyse informatique en terme d'objets, éclairée par les techniques documentaires, constitue à mon avis la meilleure des réponses actuelles dans ce domaine. Il en résulte presque naturellement une modélisation d'un accès (personnalisé) à une connaissance.

Le développement d'actions pédagogiques en vue de favoriser ces qualités d'analyse n'est pas évident dans le contexte d'un système éducatif cloisonné en disciplines qui s'ignorent trop au niveau du second cycle. Cependant l'enjeu me semble suffisamment important, dépassant certainement le cadre strict de l'enseignement optionnel de l'informatique, pour que je souhaite une progression du nombre des équipes qui travaillent dans la direction défendue ici ainsi qu'une diversification des thèmes abordés comme support des travaux pratiques.

## REFERENCES BIBLIOGRAPHIQUES

### **pour découvrir et situer les langages à objets dans la jungle des langages informatiques :**

- [1] "L'avenir des langages" dossier collectif, 01 Informatique n°986, 14 décembre 1987
- [2] "Les logiciels", Alan Kay, Pour La Science" n°85 de novembre 1984
- [3] "Smalltalk/V ou la programmation objet sur micro", Gilberte Houbart, Micro Systèmes n°79 d'octobre 1987

### **applications pédagogiques des langages à objets :**

- [4] "Langages orientés-objets et Education", Isabelle Borne, Bigre + Glogbule n°48 de janvier 1986, actes des journées AFCET Langages Orientés Objet, 8-9-10 janvier 1986
- [5] "Application d'un environnement objet à la représentation imagée des connaissances pour la résolution de problèmes arithmétiques", Isabelle Borne, Gisèle Lemoyne, actes du Congrès francophone sur l'enseignement assisté par ordinateur, Cap d'Agde 23-24-25 mars 1987, Agence de l'Informatique
- [6] "Conception d'une base de connaissances orientée objet pour l'EAO de la géométrie", Eugène Chouraqui, Carlo Inghilterra, actes du Congrès francophone sur l'enseignement assisté par ordinateur, Cap d'Agde 23-24-25 mars 1987, Agence de l'Informatique

### **applications pédagogiques des tableurs :**

- [7] "Tableur et pédagogie de l'informatique", de Jean-Baptiste Maraninchi et Raymond Favre-Nicolin, brochure éditée par le CRDP de Grenoble, juin 1986
- [8] "Enseigner la programmation à l'aide d'un tableur ?", Jean-Baptiste Maraninchi et Raymond Favre-Nicolin, "L'option informatique, réalités et pratiques, contributions au séminaire de Sèvres de juin 1985", Direction des Lycées et CRDP de Poitiers

- [9] "Le tableur en sciences physiques, vers la programmation par objets" de Raymond Favre-Nicolin et Jean-Baptiste Maraninchi, actes des troisièmes journées Informatique et Pédagogie des Sciences Physiques de Grenoble, mars 1988
- [10] "Tablez sur les tableurs" de Jean-Baptiste Maraninchi et Raymond Favre-Nicolin, Ecole et micro n°4 de novembre 1986
- [11] "Tableur et récursivité en classe de seconde" de Raymond Favre-Nicolin et Jean-Baptiste Maraninchi, brochure Pratiques et savoir-faire des élèves de l'option informatique, Direction des Lycées et Collèges et CRDP de Poitiers, 1987
- [12] "Les tableurs : évolution et impact" de Hervé Thiriez, Modèles et bases de données n°5 de décembre 1986
- [15] "Multiplan, algorithmes et calculs scientifiques" de Jean-Baptiste Maraninchi et Raymond Favre-Nicolin, à paraître, Eyrolles

**techniques documentaires :**

- [13] "Analyse et langages documentaires", de Jacques Chaumier, Entreprise moderne d'édition, septembre 1982
- [14] "Informatique documentaire" de A Deweze, Masson 1985