



HAL
open science

L'enseignement de concepts informatiques à travers une utilisation didactique de l'ordinateur et des nouvelles technologies

Luc-Olivier Pochon

► **To cite this version:**

Luc-Olivier Pochon. L'enseignement de concepts informatiques à travers une utilisation didactique de l'ordinateur et des nouvelles technologies. Troisième rencontre francophone de didactique de l'informatique, Jul 1992, Sion, Suisse. pp.185-192. edutice-00359238

HAL Id: edutice-00359238

<https://edutice.hal.science/edutice-00359238>

Submitted on 6 Feb 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.

L'ENSEIGNEMENT DE CONCEPTS INFORMATIQUES À TRAVERS UNE UTILISATION DIDACTIQUE DE L'ORDINATEUR ET DES NOUVELLES TECHNOLOGIES

Luc-Olivier POCHON

Résumé : c'est à travers des applications dans les autres disciplines (français, comptabilité, etc.) que l'informatique entre dans l'école obligatoire et dans les écoles professionnelles. En sciences, les outils informatiques à disposition sont multiples : outils de calcul (calculatrices, tableurs, solveurs), outils de simulation et logiciels d'EAO, en particulier. A partir de cette utilisation de l'ordinateur dans les disciplines, trois questions se posent :

- . Quelles sont les connaissances en informatique nécessaires pour utiliser des outils informatiques en sciences (niveau fin de scolarité obligatoire ou début de la scolarité professionnelle) ?*
- . Quels sont les apprentissages en informatique qui sont favorisés par une utilisation d'outils d'apprentissage de base à support d'ordinateur ?*
- . Quel est l'impact des concepts informatiques sur les disciplines ?*

Ce document situe ces quelques questions et apporte des éléments de réponses.

1. INTEGRATION ET USAGE DE L'ORDINATEUR DANS L'ENSEIGNEMENT DES DISCIPLINES

Dès l'apparition des premiers micro-ordinateurs, le monde de l'école a examiné ces drôles de machines sous tous leurs aspects. Tout d'abord ce sont les applications dans le champ de la didactique et de la pédagogie qui ont retenu l'attention. Puis, au fur et à mesure que ces bricolages à base de « puces » étaient pris au sérieux par le monde économique, les rapports de l'école et de l'ordinateur se sont complexifiés. Au désir des enseignants de maîtriser ces nouvelles technologies, répondait leur souci de résister aux pressions mercantiles exercées par les vendeurs de « quincaillerie ». Le devoir de prendre en compte les nouveaux savoirs introduits par les technologies de l'information était modulé par le désir de garder un point de vue critique sur les modalités et les implications de l'introduction de l'ordinateur dans la société. Dans sa « lutte » pour sa survie systémique au sein de la société, l'école est toujours partagée entre l'innovation et la reproduction.

Dans cette situation, divers modes d'introduction de l'ordinateur ont été imaginés : EAO, activités LOGO, etc. Le concept « d'intégration » de l'informatique aux disciplines est devenu extrêmement répandu. Il s'agit, dans ce cas, d'imaginer des usages de l'ordinateur dans différentes disciplines scolaires, entreprise qui a été effectuée systématiquement par le Centre vaudois de recherche pédagogique (Bernet et al., 1988 ; Genevay et Nicolet, 1988 ; Joost, 1988 ; Nicolet, 1988). Dans une analyse effectuée par Francis Meynard (1992), on voit que cette solution volontariste, parfois jugée artificielle, se banalise si le point de vue se porte sur les disciplines et non sur l'informatique.

C'est aussi cette thèse que nous voudrions soutenir en disant que, à travers « l'informatique pédagogique », les étudiants apprennent suffisamment de concepts informatiques. Cette thèse reste vague et générale. Elle demanderait à être particularisée selon le public concerné : élèves de 10 ans et calculatrice de poche, les enfants de l'expérience Edutex (Communication télématique à l'école, 1992), des adultes à faible qualification, soumis à des cours d'EAO, etc. Son intérêt est de nous pousser à rechercher ce qui constitue finalement l'informatique pour un utilisateur « courant » de l'ordinateur. C'est ce que nous tenterons de faire dans un cas particulier, après une brève considération sur l'usage de l'ordinateur.

2. HISTOIRE ET USAGE DE L'ORDINATEUR

Depuis l'époque des premiers projets d'utilisation du micro-ordinateur à fin d'enseignement et d'apprentissage, le panorama informatique s'est fortement modifié. D'une machine que l'on pouvait totalement maîtriser, le micro-ordinateur est devenu un objet à compartiments, chacun ayant ses spécialistes et professionnels ; d'une machine de bricoleur, c'est devenu un outil professionnel ; d'une machine potentiellement universelle, elle est presque entièrement dédiée aux quelques logiciels (performants) que l'utilisateur est capable de maîtriser. Par ailleurs, suite à une introduction massive de l'ordinateur dans tous les secteurs de l'activité, le niveau général des connaissances « informatiques » de la population a fortement augmenté.

Parallèlement à cette évolution, les cours d'introduction à l'informatique laissaient, dans un premier temps, une part importante à l'apprentissage de la programmation qui, nous le soupçonnons, représentait le cœur et l'attrait du phénomène pour la plupart des innovateurs. Puis les propositions ont souvent dérivé vers des aspects généraux sur l'informatisation de la société ou sur l'intégration de l'ordinateur dans les disciplines. Il n'en reste pas moins que l'ordinateur est un dispositif qui nous parle d'informatique¹. L'utilisateur, en « ouvrant » un fichier, accède, sans beaucoup d'intermédiaires, à l'« open » du programmeur qui constituait il y a quelques années une difficulté sérieuse pour de nombreux apprentis dans le domaine.

¹. En ce sens qu'une théorie scientifique se dévoile à travers les modèles et les instruments qu'elle propose. Certains préféreront peut-être restreindre ici la notion d'informatique à la conception anglo-saxonne de « computer science ».

Il peut être intéressant d'étudier plus précisément, en se focalisant sur un usage de l'ordinateur (du traitement de texte, par exemple), les objets qui se révèlent caractériser l'ordinateur, que ce soit sous la forme de remarques de la part des utilisateurs, de difficultés rencontrées ou d'erreurs. Nous appellerons l'ensemble de savoirs identifiés de cette manière, l'informatique « native ». D'une certaine manière, cette informatique est constituée de tous les concepts qui provoquent des réajustements de représentation. Dans le cas des utilisateurs du traitement de texte, on y trouve par exemple l'ouverture d'un fichier précédemment cité ou l'utilisation de la touche ESC, revêtue parfois d'un prestige inattendu.

De l'usage de l'ordinateur naît donc une informatique native. Mais, réciproquement, les concepts et méthodes de l'informatique ont également une influence sur les pratiques et représentations coutumières. Le traitement de texte peut amener certains utilisateurs à modifier leur manière de rédiger, par exemple.

3. ETUDE D'UN CAS

Le cas analysé est présenté en détail dans « Mathématique et informatique : quelques propositions d'intégration » (1992). Il s'agit d'un cours d'informatique qui se situe dans la prolongation d'un cours de mathématique en une utilisation orientée mathématique d'outils informatiques. Il s'adresse à des élèves d'une école professionnelle (école technique du Centre de formation professionnelle du Littoral neuchâtelois), tout en restant à un niveau de culture générale.

Pour situer rapidement le contexte de l'étude, signalons que le cours d'informatique donné depuis une dizaine d'années consistait tout d'abord en un cours de programmation. Les enseignants ont été confrontés à l'hétérogénéité croissante des savoirs des élèves, à la sophistication des outils de programmation, mais aussi au développement des champs d'application. La décision a été prise de modifier, dans la direction citée précédemment, les objectifs du cours. Dans un premier temps, le cours s'est basé sur l'utilisation du « tandem » FrameWork-Eureka, ce choix constituant une sorte de compromis entre l'unité² et la diversification³.

Une partie du cours se déroule sous la forme de rédaction de projets, ce qui provoque l'intrusion de la physique et du calcul professionnel.

Les constatations que l'on a pu faire ont certainement une portée relativement limitée, mais elles s'appuient toutefois sur des observations répétées. Nous n'avons repris que les épisodes qui ont concerné les quatre classes soumises à ce cours (deux classes par année, deux années de suite). Nous ferons une liste des objets de l'informatique « native », puis brièvement nous évoquerons le problème

². Même superficiels, quelques points communs aident à amorcer des transferts : fenêtres, menus déroulants, fonction zoom, « ; » pour introduire des commentaires, etc.

³. Entre ces deux logiciels, on possède trois « paradigmes » de programmation :
 assertionnel classique (FrameWork) : @if(cond,a:=2,a:=3)
 applicatif (FrameWork) : @set(a,@if(cond,2,3))
 déclaratif (Eureka) : a^2=8

des retombées de l'informatique sur l'enseignement des disciplines. L'expérience apporte également quelques hypothèses pédagogiques sur l'utilisation de l'ordinateur. Elles peuvent être toutefois davantage liées à la pratique du projet qu'au contenu du cours lui-même⁴.

3.1 Objets et obstacles de l'informatique native

Qu'est-ce que l'informatique pour un utilisateur de l'ordinateur ? C'est ce que nous tenterons de décrire ci-après par une liste d'objets qui ont provoqué des difficultés ou un étonnement chez les élèves. Il faut toutefois commencer par signaler que selon leur expérience antérieure, leurs réactions peuvent être fort diverses. Nous nous référerons donc parfois à l'un des trois types d'élèves suivants : le « joueur », le « programmeur » ou le « réfractaire ». Le « joueur », cette dernière année, était fortement intéressé par le programme de jeu « Lemmings » dont la structure en tableaux de courte durée se prête bien à l'intrusion dans la classe. Le « programmeur » a acquis une certaine dextérité dans un langage de programmation. Il est souvent en train de procéder à l'achat d'une machine personnelle. Il peut être joueur. L'« indifférent », ou le « réfractaire », ne s'intéresse que très épisodiquement à ce qui se passe sur l'ordinateur. Il a eu des rapports malheureux avec l'informatique scolaire (« j'ai déjà eu trois cours et je n'y comprends toujours rien. »).

3.1.1 L'ordinateur, une fenêtre sur un monde virtuel

L'utilisateur avisé oublie parfois les difficultés rencontrées lors de son premier contact avec l'ordinateur ; les objets côte à côte sur l'écran, véritable tableau de bord, sont de nature diverse malgré leur apparence identique et ne constituent qu'une partie (une vue, une référence) de l'information contenue dans le système. Même pour les « programmeurs », le fait qu'un programme puisse être décomposé en différents modules, contenus dans des « cadres » différents, constitue une source de difficulté.

3.1.2 Les signes et symboles

L'informatique a introduit des symboles inusités dans la typographie usuelle qui semblent porter une partie de la magie de l'ordinateur (la touche *ESC* pour certaines secrétaires, le signe \, etc.). Nous nous permettons de prendre le signe indépendamment de son contexte, parce que ce sont les utilisateurs eux-mêmes qui mettent une partie de leur fascination ou de leur source de difficultés sur le signe lui-même (« cette touche *ESC*, elle est formidable »). La difficulté est de comprendre de nouveaux mots et de nouvelles syntaxes, dans lesquels ces signes

⁴. Quelques hypothèses à propos de « projets » informatiques, que nous laissons le lecteur méditer :

- les élèves se révèlent de précieux vecteurs de l'interdisciplinarité en colportant des exercices d'une branche à l'autre ;
- ils « portent » le projet en tenant à jour la documentation relative à un problème ;
- ils obligent les maîtres à se pencher sur d'autres approches, d'autres manuels, d'autres manières de faire, d'autres notions ;
- l'enseignant peut limiter les élèves (soucis d'efficacité, peur de se lancer).

cabalistiques sont inclus. Souvent ces signes sont à comprendre selon deux syntaxes : syntaxe d'écriture et syntaxe de langage de commande. Leur logique peut différer. Le cas de « @ »⁵ rencontré avec les élèves est très caractéristique de cette ambiguïté : est-ce la représentation d'une fonction qui agit sur la commande suivante (@ empêche l'affichage de *echo*) ou ce symbole appartient-il au nom d'une nouvelle commande *@echo off* qui a la particularité de ne pas s'afficher ? De même, *@sin* peut être présenté comme une nouvelle manière d'écrire sinus jusqu'au moment où FrameWork introduit ce signe pour exécuter un programme contenu dans un cadre. On constate que la plupart des élèves l'oublie systématiquement ou alors l'écrivent déconnecté du nom de la fonction qui suit (*@ sin*).

3.1.3 Les concepts et les actions

Fichiers, répertoires, éléments d'organisation de l'information dans l'ordinateur paraissent des éléments simples. Il semble qu'il suffise de donner comme image les fichiers classiques. Malgré tout, les confusions restent nombreuses entre les opérations que les étudiants appliquent au fichier (principalement visualiser le nom et visualiser le contenu). Claude Mardirossian (1992) montre que l'objet fichier est complexe. C'est tout d'abord un nom (ou une icône), un contenu et des actions que l'on peut effectuer (ou ne pas effectuer) sur lui (par exemple, ouvrir mais parfois plutôt charger⁶). C'est à partir de ces éléments divers qu'une représentation de l'objet fichier peut se faire.

3.1.4 Les langages de manipulations

Les manipulations d'un ordinateur constituent un vrai langage avec sa syntaxe (certaines suites de touches sont interdites) et sa sémantique (F8 c'est une fonction copier pour FrameWork !). Certains élèves en restent à une alphabétisation syllabique où toutes les opérations sont répertoriées en suites d'actions élémentaires sans que le principe organisateur (schéma objet-verbe de FrameWork) ne leur vienne en aide dans leur effort de mémorisation. Il faut acquérir le sens du langage. Peut-être même imaginer d'abord que ce sens existe !

3.1.5 Les concepts de programmation

Ce sont les objets les plus communément identifiés : branchements, itérations, fonctions. Ces notions se répercutent de niveau en niveau, du programmeur du système jusqu'aux utilisateurs.

3.1.6 Faire faire

L'utilisation de l'ordinateur demande une habitude particulière de décontraction. De jeunes enfants s'étonnent lorsqu'on leur donne pour tâche de réaliser un

⁵. *@echo off* dans le cas du système d'exploitation DOS et *@sin* avec FrameWork. Dans un premier temps, les élèves ont de la peine à voir que ce même symbole peut avoir, selon le système utilisé, des « actions » différentes.

⁶. « loading, ça veut dire attendre » nous ont dit des enfants (Grossen, Pochon, 1988). Pour eux, la notion de fichier n'existe probablement pas !

programme leur demandant leur nom et les saluant⁷. On trouve encore des futurs techniciens qui résolvent les questions proposées pour faire afficher les résultats par l'ordinateur. Il y a donc un pas certain entre la capacité à utiliser des mondes simulés et celle de créer ces mondes en « inventant » l'espace et en « agissant » sur le temps !

3.1.7 Méthodologie

Mettre au point un programme demande une certaine systématique qui procède par décomposition du problème général en différentes fonctions et de la réalisation des parties et de leur intégration. Cette manière de faire ne semble pas naturelle. L'idée de procédure que l'on affine rebute les élèves. Sur un traitement de texte, ce travail par touches successives, par cercles concentriques, est également rendu possible. On note, là aussi, que les élèves préfèrent soigner les détails avant même d'avoir fait un plan global de texte à rédiger.

3.1.8 Paradigme de programmation

Les « programmeurs » ont de la peine à réaliser que l'informatique est polyglotte. Le langage connu revêt toutes les qualités (« à mon avis le Basic c'est l'avenir »).

3.2 Les retombées, calcul professionnel, physique et programmation

Les élèves ont l'occasion de rencontrer des théories sous plusieurs aspects : d'une part du point de vue du calcul professionnel (règles permettant par exemple de trouver l'énergie dépensée par une automobile pour vaincre le frottement), du point de vue de la physique (déduction des règles à partir de concepts de base) d'autre part. La simulation par programmation⁸ ouvre une troisième approche. On pourrait donc se féliciter de cet appui mutuel qu'exerce chacune des branches l'une sur l'autre. Toutefois, les difficultés rencontrées par les élèves nous incitent à porter l'attention sur la représentation qu'ils ont de la nature de ces diverses approches : calcul professionnel, physique, programmation, reprennent chaque fois le même problème, la même théorie, mais les tâches à effectuer diffèrent. La connaissance en actes est donc différente.

En *calcul professionnel*, il s'agit d'appliquer une formule. Les paramètres sont fournis par des tables. Les données sont entrées dans les formules (vitesse, etc.) et les résultats découlent de manipulations arithmétiques. La physique expérimentale part de théories (loi de Newton) en faisant varier les données et en observant des résultats. Ce sont les paramètres qui sont les véritables « résultats » de l'opération. On peut ainsi juger de la validité des lois.

En *informatique*, les paramètres et les données sont les mêmes. Ce sont en général également les résultats qui sont attendus (et donc la similitude avec les

7. « Nous, on fait comme si on programmait un programme » (Grossen, Pochon, 1988).

8. On pourra se référer à la présentation de Launaz et al., dans le bulletin no 12 (1992) de la Société des enseignants neuchâtelois de sciences (S.E.N.S.).

résultats expérimentaux permettent de valider la théorie). Mais, à moins que l'on ne possède des environnements de simulation, d'objets mobiles, etc., il faut encore inventer le temps et l'espace, fabriquer la vitesse et l'accélération, tâche d'une tout autre nature et qui introduit dans la partie informatique une problématique supplémentaire, qui rejoint celle des mondes virtuels.

4. POUR CONCLURE

En définitive, l'approche intégrée a certainement plusieurs des qualités qu'on lui prête. Elle permet une économie de moyens, elle permet une contextualisation des activités, elle peut favoriser une approche interdisciplinaire.

Par contre, nous avons vu qu'une utilisation « sur le tas », même avec des élèves avertis, peut poser plusieurs problèmes :

- problèmes de représentation divers que nous avons mentionnés dans une liste hétéroclite ;
- problème de situer les objectifs poursuivis.

Par ailleurs, l'informatique a de nombreux impacts sur les disciplines elles-mêmes, mais n'est-ce pas là une préparation à une manière de penser et d'agir de plus en plus fréquente sur les places de travail ? En connaissant de manière plus fine ces diverses représentations de l'ordinateur, qui peuvent faire obstacle à son utilisation, il pourrait être possible d'augmenter l'efficacité des cours qui se basent sur son usage et d'aménager, le cas échéant, des compléments, que des cours initiaux ou de formation continue seraient à même d'apporter.

Luc-Olivier POCHON

Institut romand de recherches et de documentation pédagogiques.
Fbg de l'Hôpital 43
CH-2000 NEUCHATEL

BIBLIOGRAPHIE

- T. BERNET, P. HABERSAAT, P. JOBIN, J. PASCHOUD (1988). *L'informatique intégrée à l'enseignement des mathématiques : suggestions de quelques situations*. Lausanne : Centre vaudois de recherches pédagogiques. (88.09).
- Communication télématique à l'école : actes de la Journée d'étude du Conseil de direction de l'IRDP, Genève, le 5 décembre 1991*. (1992). Neuchâtel : Institut romand de recherches et de documentation pédagogiques. (Ouvertures 92.403).
- E. GENEVAY & M. NICOLET (1988). *L'informatique intégrée à l'enseignement du français : Réflexions générales*. Lausanne : Centre vaudois de recherches pédagogiques. (88.07).
- M. GROSSEN & L.-O. POCHON (1988). Rapport sur l'utilisation du nano-réseau. *Cahiers de psychologie* (Neuchâtel), 27, 51-66.
- R. JAFFARD (1990). *Exemples d'utilisations pédagogiques de l'ordinateur en classe de seconde*. Lyon : Institut de recherche sur l'enseignement des mathématiques : Mission académique de la formation des personnels de la fonction publique.
- P. JOOST (1988). *L'informatique intégrée à l'enseignement de la musique : une étude de la musique électro-acoustique à l'école*. Lausanne : Centre vaudois de recherches pédagogiques. (88.08).
- J.-P. LAUNAZ and Co (1992). « Physique, calcul professionnel et informatique. » *Société des enseignants neuchâtelois de sciences*, 12, 7-10.
- C. MARDIROSSIAN (1992). Un exemple d'analyse de vocabulaire : la notion de fichiers dans les logiciels de traitement de texte. *Bulletin de l'EPI*, 66, 85-91.
- F. MEYNARD. *Comment intégrer l'ordinateur à la pédagogie*. (tiré à part). (Pour obtenir ce texte, veuillez vous adresser à : AQUOPS. Le Bus. 600, rue Fullum. Montréal H2K 4L1.)
- M. NICOLET (1988). *L'informatique intégrée à l'enseignement du français : traitement de texte : observations d'expériences en 3e et en 8e années*. Lausanne : Centre vaudois de recherches pédagogiques. (88.13).
- L.-O. POCHON (1992). *Mathématique et informatique : quelques propositions d'intégration*. Neuchâtel : Institut romand de recherches et de documentation pédagogiques. (Pratiques 92.202).