



# ***Journée PEEP*** **« L'Informatique et l'École »**

*La pensée informatique, au cœur de nos vies quotidiennes*

*Par Maurice NIVAT*

*Enseignant-chercheur en Informatique*

*Membre de l'Académie des Sciences*

*6 février 2016*

Je suis heureux de m'adresser à vous, parents d'élèves de l'enseignement public, appartenant à la PEEP et je remercie les organisateurs de cette journée de m'avoir invité à prendre la parole.

Il y a évidemment beaucoup de choses à dire au sujet de l'enseignement du « numérique » à l'école mais n'ayant la parole que très peu de temps je ne puis qu'aller à l'essentiel.

Le but de l'enseignement primaire est de donner aux enfants des connaissances suffisantes de la langue écrite et parlée et des langues des sciences pour pouvoir communiquer avec les autres membres de la société, exprimer des idées, décrire des objets et des phénomènes observés, asseoir un raisonnement en logique « naturelle » et en retour comprendre une partie suffisante du discours incessant auquel nous sommes soumis.

Apprendre à lire et à écrire, apprendre un peu de grammaire et d'orthographe est indispensable mais n'est qu'une faible partie de ce que l'enfant apprend à l'école primaire où la découverte du monde et des phénomènes dont il est le théâtre s'accompagne de l'acquisition du vocabulaire nécessaire pour en parler avec le plus de précision et le moins d'ambiguïté possible. Le travail essentiel porte sur le sens des phrases, le sens que l'on veut transmettre en parlant ou en écrivant et dont on espère que l'autre le comprendra, ou bien les sens que peuvent bien avoir des phrases entendues, des phrases qu'on a dites à l'enfant et qui impliquent souvent des réactions, réponse à un questionnement, obéissance à une injonction, remerciement pour un compliment.

Dans le monde de Rabelais, il n'y avait pas de voitures automobiles et on se déplaçait à cheval ou dans divers types de carriages tirés par des chevaux : il n'est ainsi pas étonnant que dans l'œuvre de Rabelais le mot qui revient le plus souvent après Dieu est le mot « cheval », mot qui était accompagné de beaucoup d'autres mots ou expressions liés aux chevaux, leur morphologie, leur caractère, la façon de les dresser, de les monter, de les atteler, de les conduire qui étaient forcément d'usage courant et le sont restés longtemps (il

n'y a pas si longtemps que les chevaux-vapeur ont disparu avec les 2CV et les 4CV). Progressivement tout le vocabulaire, et tout le savoir inscrit dans la langue, concernant les chevaux a été remplacé par tout un vocabulaire, et tout un savoir inscrit dans la langue, lié à la machine à vapeur, et au chemin de fer, puis par un vocabulaire lié aux véhicules automobiles, vite devenus simplement « autos ». Et toujours autour d'un mot, cheval, vapeur, chemin de fer, auto s'épanouit toute une nébuleuse d'autres mots et expressions qui emmagasine tout un savoir élémentaire que l'on apprend avec la langue, en apprenant la langue.

Je pense que tout ce que je viens de dire est bien connu des professeurs des écoles s'agissant des mots techniques désignant des objets scientifiques ou techniques d'usage très courant et des savoirs qui s'y attachent : savoirs qui peuvent être multiples ou au moins doubles, un savoir naïf, et un savoir scientifique précis. Le mot « fort », extrêmement courant, et pouvant être utilisé dans les contextes les plus variés (fort en bouche, en gueule, en thème...) ne prend un sens précis que quand on définit la « force » (mesurable) qu'il faut appliquer à une masse pour la soulever, en physique élémentaire. Vitesse, accélération, puissance, masse, poids, énergie sont ainsi d'autres mots dont l'usage naïf peut être assez distinct du sens précis que la physique leur donne et c'est bien aux enseignants de CM et CM2 de faire distinguer ces usages en leur inculquant les rudiments de physique nécessaires.

Le monde nous dit-on est devenu numérique, ce qui n'est pas faux, bien que l'emploi du mot numérique pour désigner ce nouvel état du monde constitue à mes yeux un contre-sens : le mot informatique était beaucoup plus approprié dans la mesure où c'est la généralisation de la notion d'information et le développement de puissants moyens techniques et scientifiques de traitement de celle-ci qui a amené ce changement. Ce n'est pas parce que l'information que traitent les ordinateurs est représentée par des suites de bits (ou si l'on veut de 0 et de 1) qu'elle est devenue nombre et que son traitement a quoique ce soit de numérique (numérique veut bien dire relatif au nombre). Si la suite de bits codant en MP4 le dernier discours de notre président, et celle, codant en JPG, la photo du vainqueur de l'Open d'Australie peuvent en effet être interprétées aussi comme des nombres, les dits nombres n'ont pas le plus petit intérêt d'un point de vue numérique, arithmétique, mathématique.

Ainsi les choses commencent très mal : on nous dit, le Président nous dit, son Ministère de l'Education Nationale, nous disent qu'il faut apprendre aux enfants du numérique pour pouvoir vivre et se comporter comme il faut dans le monde devenu numérique, mais il y a un contre sens dans la définition du numérique et l'usage de ce mot est en lui-même trompeur. Parfois dans certains textes ou programmes on voit réapparaître le terme « informatique » certainement plus approprié que celui de « numérique » pour désigner l'ensemble des sciences et techniques qui s'intéressent au traitement de l'information. Mais dans la plupart des textes publicitaires ou de propagande c'est plutôt d'apprendre « du code » ou « le code » ou « à coder » qu'il s'agit pour singer le Président Obama qui n'hésite pas à faire campagne lui-même pour le « coding » ou « the hour of code » (l'heure de codage).

Prenez maintenant un à un les mots anciennement français ou devenus français récemment qui gravitent autour du pôle sémantique que les pouvoirs publics ont décidé de baptiser numérique : machine, automate, programme, calcul, code, langage de programmation, algorithme, boucle, itération, logiciel, bureautique, information, données, data (big), intelligence (artificielle ou non), systèmes (opérateurs ou non), processus, mémoire, virus, antivirus, pare-feu, réseau, réseau social, protocole, partage, monitoring, mécanisation, robotisation et j'en passe (en fait je ne donne qu'un petit échantillon) et vous vous apercevrez qu'on se sait pas vraiment ce qu'ils veulent dire, que l'on peut en trouver plein de définitions différentes, voire carrément contradictoires, qui ont, ou n'ont rien à voir, avec les sens naïfs de ces mots quand ils existent (certains depuis fort longtemps) dans la langue.

Or tous ces mots, les enfants les entendent et ce, de plus en plus, depuis leur plus jeune âge, nombreux sont nos enfants qui pianotent sur un clavier ou un écran bien avant de savoir lire. La première des choses à faire pour aider les enfants à se situer dans le monde numérique qui est désormais le nôtre est simplement de leur apprendre patiemment, avec ce qu'il faut d'exemples aussi concrets que possible, et qu'ils peuvent comprendre, le sens de ces mots, des expressions et phrases qui les contiennent. Dans le même mélange d'apprentissage de la langue et d'initiation à la science par l'observation de la nature qui les amène à distinguer les usages scientifiques rigoureux des mots de leurs autres acceptions courantes en même temps que se forment dans leur esprit les concepts (que j'appellerais plutôt préscolaires) sur les quels vont reposer les sciences.

Dans le domaine qui nous rassemble aujourd'hui, disons numérico-informatique, la clarification du sens des mots utilisés quoiqu'on fasse est encore plus nécessaire qu'ailleurs car, non seulement ce sens n'est pas clair dans l'esprit de la grande majorité de la population, mais il ne l'est pas évidemment non plus dans l'esprit d'une grande partie de la corporation des informaticiens, et de plus il semble que beaucoup de gens (spécialistes comme non spécialistes) s'ingénient à le brouiller et ou le déformer . Plus va, plus on entend parler des algorithmes comme des entités autonomes qui bientôt vont nous gouverner et dont on omet de dire que ce sont tous des créations de l'esprit humain. Et l'enseignement primaire, tel qu'il existe et est pratiqué depuis plusieurs décennies, contient plusieurs algorithmes dont il est demandé aux enfants de savoir les mettre en œuvre, eux-mêmes, à la main ou mentalement. Des algorithmes qui ne sont pas du tout anecdotiques mais qui sont à la base des savoirs considérés comme fondamentaux et sur les quelles la politique actuelle est de concentrer tous les efforts dans les premiers cycles de l'enseignement public et obligatoire : les algorithmes permettant d'assurer la correction grammaticale et orthographique d'une phrase française sont au cœur des « savoir lire » et « savoir écrire » comme les algorithmes arithmétiques sur les nombres entiers sont au cœur du « savoir compter ». Je crois qu'il serait tout à fait désastreux d'enseigner une notion d'algorithme qui ne recouvre pas les algorithmes que l'on demande par ailleurs aux élèves de connaître : or ni les uns ni les autres ne sont faits pour être exécutés par des machines, ne sont faits pour être écrits comme des programmes dans aucun des langages de programmation existants.

Et ceux qui concernent la grammaire et l'orthographe du français ne sont pas présentés du tout comme des algorithmes (c'est dire une suite d'opérations effectuer pour déterminer par exemple l'orthographe correcte d'une phrase donnée) mais comme un ensemble de règles éparpillées, et souffrant pour la plupart de nombreuses exceptions : si, toutefois , une

orthographe correcte existe pour chaque phrase, la vérification par l'auteur que la phrase écrite satisfait bien toutes les règles dont l'ensemble constitue « l'orthographe » doit lui permettre de mettre l'orthographe correcte, et en ce sens, à l'ordre près de leur vérification, l'ensemble des règles détermine bien un algorithme de « correction orthographique ». Le véritable algorithme de l'addition qui est enseigné, dont la mise en œuvre par l'élève nécessite la connaissance par cœur des tables d'addition, n'est vraiment pas destiné à être exécuté par un ordinateur, une calculette ni aucun bidule électronique puis qu'il consiste à écrire, sur une feuille de papier, l'une sous l'autre, les deux suites de chiffres représentant les deux nombres dont on souhaite faire addition (« sous » signifie que les chiffres des unités de la seconde soit bien sous le chiffre des unités de la première, celui des dizaines de la seconde soit bien sous celui des dizaines de la première et ainsi de suite) . Après quoi on tire un trait horizontal, puis on commence à additionner les chiffres des unités (qui sont à droite, l'un sous l'autre) en écrivant comme chiffre des unités du résultat le chiffre des unités de cette petite addition partielle et en « retenant » 1 si ce résultat est supérieur à 10. Ainsi s'enclenche le mécanisme de l'addition en prenant les chiffres des deux opérandes qui sont les uns sous les autres, dans la position immédiatement à gauche (de celle que l'on vient de traiter) en en faisant l'addition et en rajoutant la retenue s'il y en a, en écrivant le chiffre correspondant du résultat, sous les deux que l'on vient d'additionner et en retenant ce qu'il faut. On continue jusqu'à épuisement des chiffres des opérandes.

Ce que je viens de vous dire devrait suffire à vous convaincre que la description en français de ce qu'il faut faire, de ce que l'on demande aux élèves d'apprendre à faire, mécaniquement, automatiquement, sans se tromper n'est pas simple. Mais c'est comme ça, et aucun langage de programmation ne viendra jamais à rendre cela plus simple : les difficultés intrinsèques sont dans le positionnement des chiffres et de la désignation de ceux qui feront l'objet de l'addition élémentaire suivante, puisque, c'est là le nœud de toute l'affaire, grâce à la numération décimale, on ramène l'addition de deux grands nombres écrits en numération décimale à une suite d'additions de chiffres compris entre 0 et 9, et aussi dans la notion d'itération. Il faut que les enfants comprennent que le processus que nous avons décrit peut se poursuivre de droite à gauche, que évidemment il se termine par épuisement et que quand on a traité tous les chiffres de l'opérande la plus courte (en tant que suite de chiffres) on a bien obtenu le résultat. Ce que je viens de vous dire c'est aussi que les enfants sont (officiellement, par les programmes) confrontés à la notion de boucle dès l'âge de six ans et que par conséquent les innombrables discussions pour déterminer si l'âge convenable pour comprendre ce qu'est une boucle (tant que, jusqu'à, ou pour tout) est dix, douze ou quatorze ans est assez futile.

Faute de temps je ne peux pas poursuivre longtemps la démonstration du fait évident pour moi que la construction du savoir de base, du savoir, tant pratique que théorique, le plus fondamental que l'on espère faire acquérir aux enfants dans l'enseignement primaire entre les âges de six et de onze ans passe par quantité d'algorithmes obligeant les instituteurs et institutrices à faire de l'informatique sans le savoir depuis fort longtemps : car reconnaître un algorithme comme tel le décrire précisément ( d'une façon ou d'une autre, dans un langage ou dans un autre), se convaincre de sa validité, éventuellement le comparer à d'autres qui se proposent le même but c'est vraiment faire de l'informatique.

Pour conclure je dirai que jusqu'à présent personne n'a vraiment réfléchi à l'intégration de

concepts, d'idées, de méthodes issues de l'informatique dans les savoirs de base, ceux dont on se plaît à penser qu'ils constituent le socle sur lequel tous les savoirs ultérieurs seront construits. Et je voudrais que l'on y réfléchisse car pour moi, l'informatique est très proche de l'écriture : dès que les hommes, un individu, des individus, des groupes d'individus veulent faire quelque chose la question de « comment faire ? » se pose, et c'est une question algorithmique, il faut un algorithme pour atteindre un but dans des circonstances données avec des moyens donnés. La recherche, l'invention d'algorithmes appartient en général au domaine dans lequel il est question de faire, mais la description de l'algorithme, ce que j'appellerai sa programmation, est essentiellement un problème d'écriture, ce n'est pas un hasard si l'informatique a développé des milliers de langages pour ce faire et continue à en produire de nouveaux. Il existait déjà beaucoup de langages particuliers à des domaines pour lesquels le langage vernaculaire n'est certainement pas le mieux adapté (la musique, la danse, le tissage, la taille de la pierre, l'architecture, l'arpentage, la comptabilité...). L'exécution de l'algorithme est de plus en plus souvent le fait de quantités d'hommes et de machines, électroniques ou non, et l'expérience prouve que les erreurs et les défauts se rencontrent plus fréquemment dans le programme que dans la conception de l'algorithme ou les organes physiques des machines servant à son exécution. La programmation d'un système est toujours double : tout ce qui devra être exécuté par des machines « programmables » doit être écrit dans un langage ad hoc que la machine comprend et le commun des mortels en général ne comprend pas, et doit donc être écrit dans un langage que comprennent les humains concernés, auteurs, managers du système ou simples exécutants : il faut bien que quelqu'un comprenne ce que font toutes les composantes du système et cela ne peut se faire dans un langage de programmation qui n'est pas destiné être lu.

C'est dans le jeu des deux discours celui formaliste, rigide, et aride en langage des machines, et celui métaphorique, paraphrastique, en langue naturelle, discours qui doivent être développés simultanément que se situe la grande difficulté comme la grande beauté de l'informatique, c'est aussi là que s'insèrent les possibilités d'innovation, de création, d'imagination d'autres façons de faire ou d'autres façons d'exprimer des façon de faire connues permettant de mieux les articuler entre elles.