Document generated on 08/06/2025 6:30 a.m.

Sens public

Humanités numériques ou computationnelles
Enjeux herméneutiques

Jean-Guy Meunier

2014

Repenser le numérique au 21ème siècle

URI: https://id.erudit.org/iderudit/1043651ar
DOI: https://doi.org/10.7202/1043651ar

See table of contents

Publisher(s)

Département des littératures de langue française

ISSN

2104-3272 (digital)

Explore this journal

Cite this article

Meunier, J.-G. (2014). Humanités numériques ou computationnelles : enjeux herméneutiques. *Sens public*. https://doi.org/10.7202/1043651ar

Article abstract

S'inscrivant dans l'univers des technologies informatiques, les humanités dites numériques se présentent comme un nouveau programme de recherche. Le vrai défi de ce programme de recherche ne sera cependant pas dans le traitement numérique qu'il veut appliquer aux objets et opérations des humanités mais dans la modélisation computationnelle qu'il doit leur appliquer. Malheureusement l'utilisation du concept de numérique masque la grand complexité de cette tâche de modélisation qui, elle, devra marier des fonctions computationnelles avec des fonctions interprétatives.

Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0) Sens-Public, 2014

This document is protected by copyright law. Use of the services of Érudit (including reproduction) is subject to its terms and conditions, which can be viewed online.

https://apropos.erudit.org/en/users/policy-on-use/

This article is disseminated and preserved by Érudit.

Érudit is a non-profit inter-university consortium of the Université de Montréal, Université Laval, and the Université du Québec à Montréal. Its mission is to promote and disseminate research.

https://www.erudit.org/en/

Sens [public]
Revue internationale
International Web Journal
www.sens-public.org

Humanités numériques ou computationnelles : Enjeux herméneutiques

JEAN-GUY MEUNIER

Humanités numériques ou computationnelles : Enjeux herméneutiques

Jean-Guy Meunier

Les humanités dites numériques

Dans les années 2000, en cooccurrence avec l'émergence des technologies informatiques, est apparu un programme de recherche original et novateur habilement appelé les *Digital Humanities*. Pour le monde anglophone, ce nom orientait des recherches dans une voie plus générale que ne le faisait la dénomination antérieure de *Computers and the Humanitie*s. Dans le monde francophone, l'expression anglaise, traduite par l'expression « les *humanités numériques* » est plus récente et n'est pas sans poser des questions, non pas relativement au qualitatif « digital », mais plutôt au substantif « humanités ». Comme on le sait ces termes *humanités* et les *humanities* ne couvrent pas les mêmes territoires disciplinaires dans les deux langues. En anglais : *humanities* est un terme traditionnel pour couvrir une partie de l'ensemble complémentaire des sciences dites « dures ». Il touchera surtout les sciences liées à la culture telles la sociologie, l'anthropologie culturelle, les sciences du langage, le droit, les pratiques artistiques (musique, théatre, peinture, danse) la philosophie critique, la littérature, et même le journalisme et le cinéma. Dans une acception plus précise, les *humanities* couvriront les arts, les lettres, les media mais incluront plusieurs disciplines en sciences humaines critiques, par exemple la sociologie, la philosophie, etc.

Dans le monde francophone, italien, espagnol, les *humanités* désignent généralement plus une tradition intellectuelle et même éthique de type *humaniste* d'inspiration érasmienne qu'un champ disciplinaire universitaire.

On accepte cependant de plus en plus le recouvrement des deux extensions de ces termes *Humanities* et *Humanités*, surtout si on accole les qualificatifs *digital* au premier et *numérique* au second. Aux fins de notre analyse, nous admettrons ici ce néologisme des *humanités numériques* et de *digital humanities.* Notre analyse de ce territoire ne portera pas sur la pratique effective

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

qu'on peut trouver dans les diverses disciplines académiques[1]. Notre objectif est plutôt conceptuel. Nous interrogeons non pas les termes humanities ou humanities mais ce que signifie théoriquement les qualificatifs de *numérique* et de *digital* lorsqu'ils sont appliqués *aux Humanities et aux Humanités.* Nous demanderons si la propriété d'être numérique est ce qui caractérise le mieux ce programme de recherche que sont les *humanités numériques* ou les *digital humanities*. Et nous chercherons à voir en quoi ces concepts de *numérique/digital* se distinguent de celui de computationnalité et quels sont les enjeux herméneutiques qui en découlent.

Commençons par analyser quelques définitions prototypiques de ces termes h*umanités/numériques ou digital humanities.* Une première définition nous est donnée par le centre de recherche de Digital Humanities de l'Université d'Oxford:

> By digital humanities, we mean research that uses information technology as a central part of its methodology, for creating and/or processing data…The use of the term digital Humanities] reflects a growing sense of the importance that digital tools and resources now have for humanities subjects[2].

On retrouve des définitions similaires dans la communauté francophone.

> Les humanités numériques recouvrent un ensemble de pratiques de recherche à l'intersection des technologies numériques et des différentes disciplines des sciences humaines. (Dacos et Mounier, 2014).

Certains, comme Doueihi (2011, 2014), lui ajoutent une dimension culturelle proche de la tradition érasmienne et peut-être aussi de certaines positions critiques derridiennes et foucaldiennes.

Malgré des variantes, toutes ces définitions interprètent le digital et le numérique en termes de relation/médiation à un *outil technologique* c'est-à-dire l'ordinateur ou le « computer ». Cependant, ces définitions s'intéressent plus aux impacts culturels et humains de cette technologie qu'à l'analyse technique des liens entre le numérique, la « technologie informatique » et la théorie de la computation qui lui est sous-jacente. On suppose ces liens connus du lecteur[3].

[1] De fait, la pratique concrète des Humanités numériques est devenu problématique. Comme le souligne tout semble pouvoir être l'objet des humanités numériques. Ainsi, écrit-il dans *Defining Digital Humanities*, « the term can mean anything from media studies to electronic art, from data mining to edutech, from scholarly editing to anarchic blogging, while inviting code junkies, digital artists, standards wonks, transhumanists, game theorists, free culture advocates, archivists, librarians, and edupunks under its capacious canvas. »

[2] Voir http://digital.humanities.ox.ac.uk/Support/whatarethedh.aspx pour lire l'article « What are the Digital Humanities ? » publié sur le site du centre de recherche de *Digital Humanities* de l'Université d'Oxford.

[3] Il est intéressant de remarquer que Père Busa dans ses écrits ne parlait pas de Digital humanities mais bien de *humanities computing.*

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

Mais comme on le verra, ceux-ci ne sont pas aussi si limpides qu'ils le semblent. Sur ce point, il est intéressant de noter qu'avant l'apparition de ces expressions *Digital/Humanities ou Humanités numériques*, le territoire de recherche était nommé autrement. Par exemple, le Père De Busa ne parlait pas dans ses écrits de *Digital humanities* mais bien de humanities *computing.* Mais il interprétait ce *computing* en termes d'automatisation de toutes les analyses des expressions humaines :

> " Humanities computing is precisely the automation of every possible analysis of human expression (therefore, it is exquisitely a "humanistic" activity), in the widest sense of the word, from music to the theater, from design and painting to phonetics, but whose nucleus remains the discourse of written texts ". Nous soulignons. Busa (2008) p. 1.

Pour Mc Carthy (2008), l'un des artisans importants de l'émergence des Humanités numériques, ces définitions qui voient les DH comme : *the application of the computer to the disciplines of the humanities,* sont trop générales. En fait, elles réduisent le numérique à de la technologie en focalisant l'attention sur la portée de celle-ci dans la culture ou dans toutes les expressions des activités humaines. Et pour lui, ces types de définition masquent le rôle effectif de l'ordinateur. Rôle qui ne réside pas dans des opérations numériques mais bien dans une médiation épistémique, à savoir : être une machine modélisante qui intervient entre un sujet cognitif et le monde : « *Computers are essentially modeling machines, not knowledge jukeboxes* » *(Mc Carthy, 2008 : chap. 19, p. 2).*

Nous partageons cette vision (Meunier, 2009). Tout comme bien d'autres technologies (par exemple, la photographie, le cinéma, le livre, etc.) l'ordinateur sert dans l'interaction cognitive et pratique d'un agent cognitif humain avec le monde et elle affecte sa culture et son environnement. Mais encore faut-il préciser davantage le type particulier de médiation qui est réalisée. Pour Mc Carthy, cette médiation de l'ordinateur en est une de précision *mécanique* interne propre à une modélisation computationnelle : « *the mechanical precision of the computational model* ». Et c'est à ce type de modélisation qu'il donne une finalité épistémique et épistémologique. L'ordinateur devient un outil de perception et un instrument de pensée : « *an agent of perception and instrument of thought* ». Il permet de traduire des données originelles souvent imprécises en des données évidentes et renouvelantes :" *the flawless precision of the machine makes mechanically imprecise data obvious and obviously" .* Cependant ces notions de *mécanisme* et de *computation* sont peu définies. Elles sont de fait des notions complexes et appartiennent au fondement des mathématiques et de l'informatique. Et il nous apparait nécessaire de les expliciter davantage surtout dans le cadre de leur application aux humanités.

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

La modélisation dans la technologie informatique

La caractérisation de la médiation qu'est la technologie informatique proposée par Mc Carthy nous semble juste, mais elle demeure encore générale. Il faut voir qu'elle participe d'une dynamique explicative complexe où interviennent comme le suggèrent, Dennett (1978), Pylyshyn (1984) et Marr (1982) plusieurs modélisations. De fait, ils en identifieront trois types distincts.

Une première sera dite *intentionnelle* ou *représentationnelle*. Elle visera à identifier dans un domaine quelconque, les artefacts et les tâches ou pratiques qui y sont effectuées. Mais cette identification sera faite en regard de l'observation et de la compréhension qu'en ont les utilisateurs, les spécialistes ou les experts. Et l'explication qu'on en donnera en appellera à des concepts issus du discours ordinaire tels par exemple ceux de *représentations* et d'*intention*. Une deuxième modélisation, appelée *fonctionnelle,* traduira les objets et tâches décrits dans le premier modèle en termes d'intrant et de *fonctions.* Les fonctions posséderont cependant une propriété mathématique très spécifique à savoir, être *computables.* Enfin, la troisième modélisation dite *physique ou matérielle* présentera les *mécanismes* physiques qui sous-tendent la réalisation de ces fonctions computables. En informatique, ces mécanismes seront ceux d'une machine physique de type électronique communément appelé : un ordinateur.

Pour illustrer l'interrelation entre ces trois modèles, nous prendrons ici l'exemple de la version électronique du jeu de tic-tac-toe. Pour expliquer comment ordinateur peut jouer au tic-tac-toe, il faut, dans un premier modèle, décrire le jeu tel que les joueurs humains se le représentent et le comprennent. Ils parleront ainsi des *pièces, des règles, des mouvements admissibles (bouger à droite, faire échec, conquérir, etc.)*, *des buts du jeu*, *d'une joute gagnante*, etc. Dans le deuxième modèle, on traduira les objets ou artefacts et les tâches précédentes en termes d'intrants et d'extrant ou encore de fonctions/opérations computationnelles, c'est-à-dire en précisant les données et les algorithmes (une machine abstraite de Turing). Enfin, dans le troisième modèle, il faudra réaliser ces fonctions dans un ordinateur. Toute explication d'une technologie informatique présentera, explicitement ou implicitement ces trois types modèles. Et dans la mesure où les humanités numériques sont intimement liées à une technologie informatique, elles seront amenées aussi à produire ces trois types de modèles pour expliquer leur fonctionnement et justifier leur programme de recherche.

Notre objectif ici n'est pas d'approfondir le modèle représentationnel ou le modèle physique comme tel. Nous voulons plutôt préciser plus en détail la modélisation fonctionnelle/computationnelle pour analyser les éléments techniques qui y sont en jeu afin de mieux comprendre leur relation avec le programme de recherche des humanités numériques.

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

Aussi explorerons-nous les questions suivantes : a) où se situe le numérique dans la modélisation fonctionnelle ? b) quelle est la nature de la computationnalité en regard du numérique ? c) et quels sont les enjeux herméneutiques de cette modélisation pour les humanités numériques ?

La modélisation fonctionnelle et le numérique.

La première question nous invite à préciser le concept de *numérique*. Normalement ce concept n'apparait ni dans le modèle représentationnel ni dans le modèle physique. Par exemple, pour modéliser le jeu tic-tac-toe, on ne parlera pas des entités et du mouvement du jeu en termes numériques. On utilisera plutôt des mots du langage ordinaire tels : des *pièces, des grilles, des croix, des cercles des barres, des carrés, aller en haut, au milieu, en bas, à droite, à gauche,* etc. De plus, le modèle physique, à savoir celui pertinent pour l'ordinateur, n'utilisera pas non plus ce terme de numérique. De fait, contrairement à ce qu'on pense, l'ordinateur comme machine physique ne manipule pas du numérique, mais bien des configurations électriques[4].

C'est dans le modèle fonctionnel/computationnel que le concept de *numérique* apparait. Et il s'y retrouve à un endroit très spécifique. Le numérique désigne une forme sémiotique d'encodage de configurations électriques. Pour faciliter l'appariement avec les mathématiques, deux symboles de type « chiffres » (non des nombres!) ont été choisis pour cet encodage, à savoir les symboles 0 et 1 ou leur combinaison. Mais ils auraient pu être tout autre comme, par exemple, les symboles §et #.

Cette appellation de « numérique » pour parler d'un système manipulant des symboles est cependant problématique. En effet, ces deux symboles ne désignent pas nécessairement des *nombres* c'est-à-dire des entités dans l'univers numérique comme cela est habituellement le cas en mathématique. Ils désignent, dans un langage de programmation machine que des configurations électriques. Or ce que manipulera concrètement un ordinateur ne seront pas ces symboles eux-mêmes, mais les configurations électriques elles-mêmes que ces deux symboles, 0 et 1, encodent.

Par ailleurs le choix du terme *numérique* n'est pas sans conséquence théorique. En effet, il laisse supposer que ces symboles eux-mêmes, 0 et 1, garantissent que les fonctions qui les manipuleront possèdent la propriété d'être computable. Une telle supposition, comme nous le

[4] De fait, ces chiffres 0 et 1 se retrouvent presque uniquement dans le langage dit « machine » lesquels seront par compilation reliés a des configurations électriques spécifiques. Ce sont ces dernières que l'ordinateur comme machine physique manipulera. Les langages de programmation de haut niveau ne contiennent que rarement ces symboles.

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

verrons, va à l'encontre du concept même de *computation*. Ce n'est pas parce qu'un intrant est représenté par des symboles par 0 et 1 que le traitement fonctionnel qui pourrait lui être appliqué serait computationnel. Une telle remarque nous mène directement à notre deuxième question : quelle est la nature de la computationnalité ?

Le computationnel.

La vision populaire ce concept de computation n'est pas toujours des plus clairs. En effet, pour plusieurs personnes, il n'y a pas vraiment de différence entre un ordinateur et la computation. Pourtant, il y en a bien une. La *computation* est un terme qui désigne en mathématiques une propriété formelle d'une fonction mathématique, alors que le terme *ordinateur* désigne la machine physique particulière parmi plusieurs autres possibles peut effectuer concrètement le calcul d'une fonction. Mais ces deux notions ne sont pas sans liens importants : Si une fonction, en soi, n'est pas computable alors, elle ne peut être « calculée » par un ordinateur. Cependant, ce n'est pas l'ordinateur qui détermine si une fonction est en soi computable. Celle-ci doit posséder des propriétés spécifiques. Ainsi, la computationnalité est une notion logiquement antérieure à celle d'ordinateur[5].

La computationnalité comme propriété formelle est de nature mathématique. Elle est liée au problème de Hilbert sur la *calculabilité* c'est-à-dire de la manipulation formelle (sans signification) de symboles pour *décider* si une formule particulière (une équation, une formule logique) *appartient ou n'appartient pas à un système mathématique (celui de nombres).* Church reprendra ce concept en le définissant en termes de *récursivité*. Et Turing[6] (1936) démontrera qu'elle est *équivalente* à la " computation " effectuée par une machine physique mécanique dite machine de

[5] Il faut nuancer cette formulation ici. Le fait pour une fonction d'être réalisée par une machine physique comme la machine de Turing ou sa variante une machine de Van Neuman, ou son équivalent un ordinateur, garantit que la fonction est computable. Mais il n'en demeure pas moins que sur un plan intensionnel, l'ordinateur et la Machine de Turing ne sont pas identiques à une fonction computationnelle. Elles lui sont simplement extensionnellement équivalentes . Ce que l'un peut computer l'autre peut aussi le computer.

[6] De manière spécifique et précise, cette thèse soutient « Que la même classe de fonctions partielles (et donc de fonctions totales) est obtenue dans chaque cas. » (Rogers 1987 :19) Autrement dit, pour chaque classe de fonctions récursives, une machine de Turing peut être trouvée. *Il s'est avéré aussi que tous les modèles de machines computationnelles proposés ultérieurement définissent aussi une classe de fonctions qui peuvent toutes être décrites par des instructions définissant exactement le même type de fonctions computationnelles* .

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

Turing[7]. Plus tard, elle sera définie en termes algorithmiques (Markov 1960), génératifs (Post 1944) ou de combinatoires (Curry et Fey, 1958). Von Neuman (1945) démontrera à son tour que certaines architectures physiques des ordinateurs sont des machines de Turing concrètes. Ainsi, dans une formulation plus générale, on dira qu'une fonction est calculable ou computable, algorithmisable, etc. s'il existe *une procédure effective* pour traiter des intrants et produire systématiquement des résultats. L'ordinateur est une architecture physique adéquate pour effectuer ce calcul d'une fonction. Il y a donc une différence formelle entre un ordinateur et la computation.

> A computation is a process whereby we proceed from initially given objects, called inputs, according to a fixed set of rules, called a program, procedure, or algorithm, through a series of steps and arrive at the end of these steps with a final result, called the output. The algorithm, as a set of rules proceeding from inputs to output, must be precise and definite, with each successive step clearly determined. The concept of computability concerns those objects which may be specified in principle by computations, and includes relative computability (Soare, 1996 : 286)

Pylyshyn et Fodor, (1988) proposent une reformulation qui permettra une compréhension plus facile de la computationnalité par les chercheurs en sciences cognitives. Dans cette reformulation, une fonction sera reconnue comme computationnelle si elle possède au moins les trois propriétés suivantes : l'atomicité, la systématicité et la productivité. Ils ajoutent, pour le domaine de sciences cognitives : l'interprétativité.

La première propriété touche l'admissibilité des intrants *(appelés arguments)* à une fonction computable. Ceux-ci doivent être discrets ou « atomiques » c'est-à-dire non continus[8]. Habituellement cet encodage numérique 0 et 1 exprime sémiotiquement cette propriété. La deuxième et la troisième propriété sont les plus importantes pour notre propos. Elles spécifient la nature de la procédure ou les opérations qui seront appliquées à ces intrants. En effet, celles-ci doivent être *systématiques* et *productives*, c'est-à-dire que les procédures doivent permettre de *produire* de manière *systématique* des séquences de symboles de plus en plus complexes ou de les réduire en des séquences de symboles plus simples. Ces deux propriétés de systématicité et

[7] On a ainsi démontré que toutes les fonctions effectivement calculables investiguées « à savoir : les fonctions lambda définissables et récursives, les formes canoniques normales de Post, la logique combinatoire de Schönfinkel, les algorithmes de Markov et la notion de reconnaissance de Gödel's, les langages de Kleene sont tous interdéfinissables et traduisibles l'une dans l'autre. Plus spécifiquement ils peuvent tous être caractérisés par une machine de Turing. »

[8] Lacombe (2014) a par ailleurs démontré que des systèmes qui traitaient du continus pouvaient être des systèmes computables.

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

de productivité sont des manières plus simples de parler de la *récursivité*, de *l'algorithmique* et de la *combinatiorialité*. Mais ultimement ce sont ces deux propriétés qui permettront à une machine de Turing physique de manipuler physiquement les intrants. Et l'ordinateur est une forme électronique de cette machine de Turing. Enfin, il faut remarquer que si un système possède uniquement ces trois propriétés, il est de ce fait un système formel au sens de Hilbert. Et comme le souligneront plusieurs chercheurs, mais que resouligneront régulièrement Searle (1980) et Harnad (1990), pour le domaine cognitif, un tel système de fonctions computables est de nature essentiellement " syntaxique ". Il est certes interprétable, mais sa sémantique lui est externe. D'où l'importance pour les sciences cognitives que soit ajoutée la propriété d'interprétativité.

Pour illustrer ce concept de computationnalité, nous prendrons deux exemples simples. Le premier illustre les dimensions atomiques et récursives d'une fonction calculable. Prenons ainsi une surface rectangulaire dont il faut « calculer » le nombre nécessaire de tuiles carrées pour la couvrir. Pour ce faire, il faut que les tuiles soient représentées par des symboles atomiques. Le chiffre 1 désignera la surface d'une tuile, les chiffres 3 et 2 désigneront respectivement les côtés du rectangle. On appliquera à ces symboles une fonction récursive d'addition. Une première addition sera appliquée successivement au chiffre 1 du premier coté pour donner le résultat noté 3. Ensuite, la même fonction sera appliquée au second côté (3) pour donner encore le résultat noté 3. Enfin, la même fonction d'addition sera appliquée au premier résultat noté 3 ainsi qu'au second 3 pour obtenir le résultat noté 6. Ainsi, le résultat final de toute la fonction sera appliqué récursivement sur des symboles. Dans les mathématiques classiques, cette fonction récursive sera représentée de manière très synthétique par des formules symboliques et exprimée dans une équation dite de multiplication : *La surface S* $_{déf.}$: S (x, y) = (x × y). Cette formulation est une manière conventionnelle de dire que la fonction est récursive, algorithmique et combinatoire. Et, dans la terminologie de Fodor et Pylyshin, elle est *atomique, systématique et productive*. En conséquence une machine de Turing pourra « calculer » cette fonction. On remarquera que ce calcul ne fait que manipuler symboles/chiffres et qu'il est indépendant de l'interprétation concrète qu'on pourrait leur donner: par exemple, surface d'une tuile en céramique, en tapis, en bois, etc.

Un deuxième exemple montre que la computation peut s'appliquer à autre chose que des symboles strictement numériques du type 0 et 1. Ici, des procédures systématiques et productives seront en effet appliquées à des symboles (ici alphabétiques). Et elles seront présentées sous la forme de règles :

Soit trois symboles *a* et *b* auquel on ajoute le méta-symbole → qui signifie : *se réécrit*

Le tout est présenté sous forme de grammaire :

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

Règles 1) : V → {a, b} ;

que l'on peut combiner aux 2 règles suivantes :

Règle 2) : a → b ;

Règle 3) : b → ba

Cet ensemble de règles permettra alors de *produire systématiquement* la séquence suivante de symbole

- b (par règle 1)
- bab (par règle 3)
- ababb (par réapplication de la règle 2

et ainsi de suite pour obtenir la séquence

- *babbababbabbababbabab*…

On peut complexifier cet exemple en écrivant une grammaire syntaxique plus sophistiquée du genre de celle que proposent les grammaires génératives. Et toute grammaire possédant ce type de règles systématiques et productives pourrait permettre à une machine de Turing de « computer » de telles séquences. On remarquera encore que cette combinatoire est sans sémantique : la grammaire y est purement syntaxique.

Comme on le voit par ces exemples simples, une fonction est computable si les intrants, représentés par des symboles, sont atomiques et si les opérations sont systématiques et productives, c'est-à-dire récursives, algorithmiques, ou combinatoires. En conséquence, ce n'est pas le fait qu'une fonction admette des symboles de chiffres qui permet de dire qu'une fonction est computable. Certes, il y aura des cas où la computation se réalisera sur des *symboles* de types chiffres, et qui seront interprétables comme des *nombres*. Mais il existe de nombreux autres cas où elle pourra s'appliquer à des symboles qui ne sont pas des chiffres et qui sont sans lien avec des nombres. En conséquence, bien que chiffres ou symboles numériques 0 et 1 soient des intrants atomiques pour une fonction, il ne s'ensuit pas formellement que la fonction qui les manipulera sera computable. Être numérique n'est pas une condition nécessaire ni suffisante de la computationnalité.

Le non computable.

Cet approfondissement du concept de la computationnalité nous permet maintenant de mettre en évidence une première aporie des théories de la computation. En effet, il est possible de construire des modèles fonctionnels qui contiendront des fonctions applicables à des intrants atomiques et encodables en des symboles numériques, mais qui de par leur nature ne seraient

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

pas des fonctions computables. Nous serions dans un modèle fonctionnel numérique mais aucunement dans un modèle computationnel !

En mathématiques, l'existence de ces types de fonctions non computables n'est pas un fait rare. De fait, comme l'ont montré plusieurs mathématiciens et informaticiens, les fonctions computables ne sont qu'un sous-ensemble très restreint des fonctions mathématiques. Les fonctions computables sont en nombre infini $\aleph_0$ alors que les fonctions non computables sont en nombre infiniment plus grand ($\aleph_0^2$). Autrement dit, elles ne sont pas dénombrables. Et pour certains informaticiens contemporains, les fonctions non computables sont théoriquement beaucoup plus intéressantes que les fonctions computables !

The subject (of computation) is primarily about incomputable objects not computable ones. (Soare, 2009 : 59)

Pour illustrer la non-computationnalité de certaines opérations ou fonctions, nous reprendrons ici des variantes de nos exemples précédents. Le premier nous est inspiré de Penrose (1994). Imaginons que pour couvrir la surface de notre rectangle précédent dont les côtés étaient de 3 par 2, nous ayons non plus des tuiles de forme carrée, mais des tuiles ayant des formes irrégulières atypiques comme celle-ci :

Penrose prouva qu'il n'existait pas de fonctions récursives ou d'algorithmes pour montrer si ces formes pouvaient couvrir un plan entier. Autrement dit, il n'est pas possible de trouver une fonction algorithmique du genre f(x)=(x × y) permettant de calculer le nombre de tuiles nécessaires pour recouvrir cette surface !

Le second exemple est mathématique. Il se présente sous la forme d'un système de plusieurs équations ayant toutes la forme suivante :

Ax+by=c où a, b, c sont des constantes.

Ce type de système d'équations est appelé *diophantien*. Intuitivement, il semble facile de lui trouver des solutions algorithmiques. Au contraire, Matiyasevich (1970) a démontré qu'il n'en existait pas. Autrement dit, bien que les valeurs données aux variables de la fonction puissent être des nombres entiers et donc atomiques, ce type de système d'équations est non

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

computable. Bref, ces deux exemples choisis parmi une infinité d'autres montrent que certains systèmes de fonctions mathématiques ne sont pas computables.

Cette question de la non-computationnalité est encore plus problématique qu'on le pense. En effet, il existe des systèmes, qui bien qu'opérant avec des fonctions computables peuvent, s'ils présentent une grande complexité, se transformer en systèmes non computables. Chaitin (1998) a récemment montré que plus les systèmes sont complexes, plus ils incluront de la redondance (ce qui assure une stabilité au système). Or, plus celle-ci est forte, plus elle risque de générer de l'aléatoire, ce qui, à son tour risque de diriger le système vers une probabilité absolue appelée *Omega* qui, selon un théorème de Chaitin (1998), rendra le système non computable ! Autrement dit, bien qu'au départ un système complexe puisse sembler computable, il peut paradoxalement devenir un système non computable ! En conséquence, plus ce qui doit être modélisé fonctionnellement est complexe, plus les probabilités qu'il soit non computable sont fortes. Se pose alors la question importante suivante : si un ordinateur ne peut traiter que des fonctions computables, comment pourrait-il computer des fonctions en soi non computables ?

Déjà, Turing avait été confronté à ce même problème. Et en 1939, il proposait une solution originale pour rendre computables des fonctions arithmétiques non computables. Il en appelait à des « oracles » :

> " With the help of the oracle we could form a new kind of machine (call them o-machines), having as one of its fundamental processes that of solving a given number-theoretic problem. "
> (Turing, 1939 : 161)

A sa sortie, cette solution ne fut pas vraiment remarquée, mais aujourd'hui elle est remise à l'avant-scène des théories de la computation. Ce fut une solution astucieuse. Certains font même l'hypothèse que ce type de système existe dans le monde physique. Et il existerait des machines, ou des hyper-ordinateurs dits hypercomputationnels qui pourraient calculer effectivement des fonctions en soi non computables. Mais plusieurs doutent de cette possibilité. Et le tout demeure un objet de débats importants en informatique théorique (Davis, 1982 ; 2006 ; Feferman, 2006 ; Shagir, 2002).

Comme l'on voit cette notion de computation est complexe. Mais, au-delà des détails techniques, l'important pour nous ici est de voir que la distinction entre le numérique et le computationnel n'est pas uniquement une question terminologique ou sémantique. Chacun de ces termes aborde des objets et des problèmes théoriques spécifiques. Le *numérique* touche un problème de représentation sémiotique, c'est-à-dire d'encodage formel – applicable à divers types d'intrants admissibles pour une machine de Turing, et ultimement pour un ordinateur. Le *computationnel* pour sa part s'intéresse à un type particulier de fonctions qui peuvent être

appliquées de manière récursive, algorithmique à des intrants atomiques. De cette différence des problèmes auxquels renvoient le numérique et le computationnel, il s'ensuit donc ce n'est pas le fait que des intrants soient numériques qui garantit que les fonctions qui pourraient les traiter soient elles-mêmes computationnelles. Autrement dit, un système numérique n'est pas toujours un système computationnel.

Le computationnel et le non computationnel dans les humanités numériques.

Les analyses précédentes sont des plus importantes pour comprendre un programme de recherche qui vise à étendre ces notions de *numérique* et de *computationnalité* à l'« univers » des humanités. Comme on le voit depuis quelques décennies maintenant, les technologies informatiques s'insèrent dans les multiples tâches et objets des humanités, même les plus classiques. Mais, quel que soit le projet dans lequel les humanités s'engageront, elles devront, dans leur méthodologie, explicitement ou implicitement construire une triple modélisation. Ainsi, un modèle représentationnel devra identifier des « entités », c'est-à-dire des artefacts, des faits, des objets et des tâches à réaliser qui sont spécifiques aux humanités. On peut penser ici aux textes et documents et à leur écriture, lecture, analyse, distribution, archivage*, etc.,* à la musique et à sa composition, interprétation, écoute, à la danse, aux rituels, à l'architecture, au cinéma, aux arts visuels, etc. Un modèle fonctionnel pour sa part devra traduire ces entités et tâches en des intrants et des fonctions. Et le modèle physique déterminera quels processus un ordinateur réalisera pour effectuer concrètement la computation de ces fonctions. Or, dans cette modélisation, les problèmes théoriques soulignés plus haut réapparaitront nécessairement. Et la question importante qui aura retenu notre attention plus haut réapparaitra : dans ces humanités, est-il possible de traduire la majorité des artefacts et des tâches en des unités atomiques ou discrètes et en des fonctions computables ? Autrement dit, *les artefacts et tâches des humanités que l'on retrouve dans un modèle représentationnel et que l'on traduit dans un modèle fonctionnel comme des intrants et des fonctions, pourront-ils se voir attribuer les propriétés d'atomicité, de systématicité, de productivité pour qu'éventuellement ils puissent être traités par un ordinateur ?*

Une telle question ne surgit évidemment que très peu chez les chercheurs des humanités numériques. On semble plus intéressé à ce qui devrait être encodé numériquement qu'aux fonctions computationnelles qu'on peut appliquer dans ce territoire. Serait-ce la littérature, l'histoire, la philosophie, la musique, les arts visuels, etc. Ce qui n'est pas sans entrainer des

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

conflits au sein de certaines disciplines (voir Kirch A, 2004). Pourtant la question du computationnel est fondamentale pour le programme de recherche des humanités, quel que soit le projet spécifique qui est retenu. Et une réponse à cette question doit être fournie si on veut justifier l'existence même de ce programme de recherche. Nous tenterons ici d'entamer une réponse à cette question en analysant ces notions d'atomicité et de récursivité lorsqu'elles sont utilisées dans le contexte des humanités.

Malgré ce qu'on peut penser, ces deux propriétés se retrouvent régulièrement appliquées aux artefacts et tâches des humanités. Ainsi, pour la condition d'atomicité, on peut identifier de nombreuses unités de types discrètes. Par exemple, en musique on aura *la note, la portée, la pièce, le compositeur*, *l'œuvre*, etc. ; en littérature, on aura *le mot, la ligne, le chapitre, l'auteur, le livre,* etc. La peinture choisira *la toile, le peintre, les périodes*, etc. Toutes ces entités discrètes sont facilement traduisibles en des formes atomiques encodables par des symboles, 0 et 1, dits binaires ou « digital », c'est-à-dire numériques. Un livre numérisé traduit chaque lettre et mot en une suite de tels symboles.

De plus, on trouvera aussi de nombreuses tâches qui peuvent aussi être traduites en des fonctions computationnelles. Souvent, ces tâches sont si intégrées qu'elles sont devenues des « automatismes ». Leur structure récursive est transparente et, parce qu'elles auront été réalisées à la main, leur structure algorithmique n'est pas évidente. Certaines d'entre elles sont très simples. Par exemple, on peut penser à *copier, imprimer, paginer, indexer un texte, une pièce de musique, compter des mots,* etc. Certaines auront même des approches quantitatives cachées et floues. Par exemple, dans les partitions musicales, on aura des indicateurs quantitatifs flous comme *piano, pianissimo, allégro,* etc. Et en littérature, en peinture, en danse, dans les films, les rituels certaines analyses encore plus complexes. Pensons par exemple à la comparaison de manuscrits. Une partie de cette tâche reposera sur des comparaisons entre parties de texte dont le but sera de découvrir des différences de formes atomiques (mots, séquences, etc.) et qu'on appellera des « variantes ». Bref, si on porte attention, on trouvera un grand nombre de tâches, mêmes complexes, qui sont traduisibles en des algorithmes c'est-à-dire en des fonctions computables.

Bien que ces artefacts et tâches existent effectivement dans les humanités, celles-ci, du moins pour les chercheurs experts, seront vues comme relativement simples et élémentaires. Elles ne sont pas, pour eux, les tâches les plus intéressantes. En effet, dans leur pratique, ces chercheurs voudront travailler sur des tâches autrement plus complexes que ces dernières, comme, par exemple : *l'identification d'un style musical, la chorégraphie d'une danse, la formation des concepts, l'évolution de thèmes, la structure narrative, les liaisons intertextuelles,*

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

les biais épistémiques, les tendances architecturales, les conditions de production de discours, les structures argumentatives, la production d'une toile, la *préparation d'une exposition, la transmission des mêmes,* etc. Et il existe une infinité de tels types de tâches dans chaque sous-domaine des humanités. Aussi faut-il se demander s'il est possible de construire un modèle fonctionnel computationnel pour ces tâches très complexes. De quelle nature seront les fonctions ? Posséderont-elles toutes les conditions nécessaires pour qu'elles soient computationnelles, algorithmisables, récursives, combinatoires, etc. ? Les réponses à ces questions ne sont pas faciles. Postuler que la modélisation fonctionnelle computationnelle pour traduire adéquatement tant la multiplicité et la complexité de ces artefacts et de ces tâches que veulent réaliser les experts des humanités, est une hypothèse que peu de chercheurs seraient prêts à soutenir. Voyons cependant ce qu'il en est vraiment.

Commençons par étudier la propriété d'*atomicité*. Pourquoi est-ce si difficile d'atomiser les artefacts complexes des humanités ? Peut-on, par exemple, atomiser les mouvements subtils d'une danse, les multiples variations d'une suite musicale, les intrigues d'un roman, ou encore la sculpturalité des statues géantes de l'Ile de Pâques ? Une telle atomisation de ces entités ou artefacts semble très difficile à réaliser. Ceux-ci sont d'une nature particulière.

Dans les humanités, les « entités » identifiées ne sont pas, comme dans d'autres sciences, des entités du monde physique c'est-à-dire, des « choses » ou des « objets matériels ». Les « entités » intéressantes sont avant de nature *sémiotique.* Leur qualité première n'est pas d'être dénombrable, énumérable mais avant d'avoir un *sens* ou une *signification.* Dans les termes classiques de la phénoménologie ou de la philosophie du corps situé, cette propriété de sémioticité trouve son origine dans la relation que ces entités entretiennent avec les actions intentionnelles humaines. Hors de cette relation, ces entités ne sont que des « choses » d'un monde quelconque. Et on ne saurait même pas dire quelle chose elles sont dans ce monde. Par exemple, une tige circulaire et soudée de 1 millimètre d'épaisseur et de 1.5 centimètres de diamètre, dont la structure moléculaire est celle de l'or ou de titanium, n'est pas en soi un objet sémiotique. Mais elle peut le devenir dans un rituel de mariage, un engagement de couple. À ce moment elle devient une *alliance*, une *bague,* etc. Et ce n'est que comme entité sémiotique que cet objet physique devient intéressant dans une recherche dans les humanités. Et un artiste ne verra pas cet objet comme un anthropologue.

Si, dans le cadre des humanités numériques, il faut construire un modèle fonctionnel pour le traitement informatique de tels objets, des difficultés importantes surgiront. D'une part, leur traduction en des intrants atomiques risque de faire disparaitre cette sémioticité. Par exemple, un geste dont la signification repose sur la fluidité, la souplesse et la continuité, risque de perdre sa

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

signification lorsqu'il est déconstruit en mille et une unités discrètes. Un anneau qui est déconstruit en n propriétés (circonférence, épaisseur, structure moléculaire) aura peine à y superposer une propriété sémiotique. Le sens n'est pas une propriété de même type ou catégorie logique que des propriétés physiques. Ce serait comme décrire l'or en lui assignant les propriétés suivantes : est jaune, brillant, *nom commun* ! On voit la conséquence immédiate que peut entrainer une atomisation qui ne pourrait s'en ternir qu'à des propriétés non sémiotiques : elle fait disparaitre immédiatement l'intérêt d'un chercheur des humanités. Celui-ci ne s'intéresse pas aux « tiges de métal en or » mais aux « alliances » ! L'intérêt d'un littéraire pour un mot dans un poème écrit n'est pas sa quantité d'encre mais son sens.

Par ailleurs, si on tente de respecter la sémioticité de ces entités dans un modèle fonctionnel alors les intrants risquent de perdre leur atomicité. Et de ce fait, ils ne sont plus aussi facilement admissibles à des fonctions computationnelles. Une *alliance-en-or-ayant-le –sens-d'engagement-d'un-couple-dans-une-communauté* est un tout complexe difficilement décomposable en unités atomiques. La vibration sonore en hertz d'une tige métallique est autre chose qu'une *tonalité-musicale–en-harmonique-mineure–d'une-perfomance-musicale-pour-une-cérémonie–funéraire*.

Abordons maintenant la deuxième propriété : celle de la récursivité, c'est-à-dire de la systématicité et de la productivité des fonctions. Ici le problème se complique. En effet, si le modèle fonctionnel est confronté à des entités où la sémioticité ne peut être éliminée et qu'il faut les traiter de manière computationnelle, on risque de rencontrer des problèmes encore plus difficiles que celui que pose l'atomicité. Ainsi, si ces fonctions computables qu'on veut appliquer aux intrants doivent être récursives, elles seront par définition purement syntaxiques. La sémantique leur échappera. Comment alors peut-on concevoir un modèle fonctionnel dont le traitement syntaxique doit s'appliquer à des entités où la sémantique est essentielle ?

Des solutions techniques existent. Il est en effet formellement possible de construire des fonctions computationnelles sensibles à une sémantique. Mais le coût pour ce faire sera extrêmement lourd. Un exemple simple et amusant nous introduira ce problème. Supposons qu'un chercheur des humanités veut construire un modèle intentionnel représentationnel de l'événement suivant :

> Plusieurs êtres vivants de l'espèce biologique humaine causent le frappement de leurs mains l'une contre l'autre mais collectivement et répétitivement devant un autre être vivant de la même espèce biologique humaine qui, lui, cause la vibration de boyau de fils de métalliques tendus sur une caisse allongée et en bois vernis à l'aide d'une tige sur laquelle sont tendus quelques cheveux de la queue d'un cheval.

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

Ce fait ou évènement auquel réfère cet énoncé peut se voir appliquer plusieurs modèles représentationnels. Un premier pourrait être exprimé par la précédente description qui, elle-même, on le voit, n'est pas théoriquement neutre. Elle est, de fait, une modélisation représentationnelle spécifique de type « physicaliste-naturaliste » que pourrait inventer un physicien biologiste de ce que causent des agents sur des objets. Mais elle n'est pas la seule possible. Par exemple, un neurologue pourrait décrire la même chose, mais en insérant une autre signification. Il pourrait y voir des « signes » précurseurs de la maladie collective de Parkinson ! Ces deux formulations modélisantes sont peut-être pertinentes pour ces types de chercheurs mais elles sont sans intérêt pour un chercheur des humanités. Celui-ci devra fournir une autre modélisation représentationnelle qui sera probablement plus pertinente et explicative pour notre propos. Par exemple, un musicologue pourrait y voir tout simplement des « applaudissements » devant une performance musicale d'un violoniste! Bref, sur un même fait ou événement, plusieurs modèles représentationnels ou descriptions intentionnelles sont possibles lorsqu'on y reconnait du sémiotique.

Cet exemple, amusant, fait apparaitre la difficulté technique d'appliquer un modèle fonctionnel/computationnel à des objets ou faits où la signification doit rester attachée. La modélisation fonctionnelle est confrontée à une multiplicité de modèles représentationnels différents. Les uns sont aussi justifiables que les autres. Que faire alors ? Construire un modèle fonctionnel unique pour chaque modèle représentationnel ou encore construire plusieurs modèles fonctionnels pour chaque modèle représentationnel ? Les deux solutions sont possibles mais elles sont aussi problématiques l'une que l'autre.

Dans la première, il faut créer un modèle fonctionnel pour chacun des modèles représentationnels. Une telle situation est élégante mais problématique. Cela signifie que chaque modèle fonctionnel porte une signature spécifique. Elle vient du constructeur du modèle représentationnel. Par exemple, un modèle computationnel pour le traitement des modèles représentationnels des statues des Iles de Pâques, recevra une signature différente selon que celui-ci vient d'un anthropologue comme, par exemple un Metraux[9] (1935, 1941) ou d'un archéologue, comme par exemple, d'un Pitts M. J.[10] (2014). Et qu'en serait-il s'il venait d'un Levis

[9] A. Métraux fut l'un des premier anthropologue a avancé des thèses célèbres sur les l'existence de ces statues.

[10] Voir les découvertes faites par L archéologue Pitts par des moyens informatiques dans Pitts, M. et *al.* (2014) et voir l'article « New and surprising discoveries about Mysterious Easter Island Statue : Origin of the Birdman Cult » (2013) sur Science World Report via http://www.scienceworldreport.com/articles/6178/20130411/new-surprising-discoveries-mysterious-easter-island-statue-origin-birdman-cult.htm.

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

Straus ou un Descola. Or ces constructeurs de modèles sont des agents cognitifs marqués par leur langue, leur culture, leurs disciplines, etc. Et les résultats de la computation en seront nécessairement marqués par ces signatures.

Dans la deuxième situation, contrairement la première, il se pourrait que, pour chacun des modèles représentationnels, il existe plus qu'un modèle fonctionnel[11]. Dans ce cas, il existerait une multiplicité de modèles fonctionnels pour chaque modèle représentationnel. Par exemple, si on utilisait un modèle représentationnel à la Levis Straus pour décrire les statues de l'Ile de Pâques, il se pourrait que le modèle computationnel réalisé par un informaticien de la tradition intelligence artificiel comme Minsky (1975) soit radicalement différent de celui construit par un informaticien de la tradition connexionniste comme Rumelhart et Mc Lelland (1986). Encore une fois, les résultats de la computation seraient différents dans les deux cas.

Mais on peut aussi complexifier ces deux situations. Imaginons que l'ensemble de ces nombreux modèles repose sur des sémantiques totalement autonomes et étanches l'un à l'autre : par exemple, si la sémantique des statues construite par un archéologue était complètement autonome et close par rapport à celle proposée par un anthropologue. Ainsi, ce que signifierait le mot « socle » dans un modèle serait radicalement différent dans les deux systèmes. Ou, encore, imaginons que leurs sémantiques se recouvrent ou se croisent. Par exemple, si pour ces statues géantes, la sémantique de l'archéologue et de l'anthropologue partageait certains éléments tels par exemple, la périodisation, l'usage, les visages, etc.

Dessin de Julien Viaud alias Pierre Loti Statues géantes des 'Iles de Pâques'[12].

[11] En termes formels, pour chaque modèle représentationnel MRi correspond un infinité de modèle Mf_{ij}, c'est-à-dire pour un MR_i spécifique , il peut y avoir j modèles fonctionnels possibles !

[12] http://www.bloghotel.org/Errance/Visites+culturelles/ consulté le 12 novembre 2014.

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

Dans les deux cas, la computationnalité qui y serait effectuée poserait des problèmes graves de compréhension et de comparaison des résultats. Ainsi, dans la première situation, parce les sémantiques sont autonomes et étanches, les résultats obtenus par un traitement computationnel réalisé sur les statues géantes à partir du modèle d'un archéologue ou d'un informaticien de la tradition connexionniste seraient incompréhensible et non comparables à ceux réalisés par l'anthropologue ou un informaticien de la tradition IA. Autrement dit, la sémantique des deux systèmes de modèles est hermétique l'une à l'autre.

Cependant, comme dans la deuxième situation où la sémantique des modèles se recouvre, il se pourrait que pour assurer et soutenir ce partage, il faille augmenter la redondance. Si tel est le cas, il y aurait certes plus de compréhension et la comparaison mutuelle des modèles. Mais comme le soutien Chaitin, la redondance[13] dans ces modèles augmentera l'aléatoire. Et donc cela pourrait diriger le système vers du non computable. Et si tel est le cas, la belle fiabilité et validité qu'assurait la computation disparait. Un résultat computationnel en vaudra un autre.

On voit, je l'espère, la complexité qu'il y a à penser le positionnement de la technologie informatique dans le domaine des humanités. Approcher ce domaine dans le paradigme du numérique demeure pertinent certes, mais il masque des problèmes d'une grande complexité. Par ailleurs, ce domaine à travers le paradigme de la computationnalité ouvre une immense boite de Pandore. On voit le défi qui est présenté au programme de recherche des humanités numériques si la modélisation doit respecter la sémioticité des objets et des tâches mis en œuvre dans de ce programme. Il est difficile, si ce n'est que pour des exemples simples, de penser que ces objets sémiotisés moindrement intéressants sont facilement atomisables et que les tâches à effectuer sont toujours algorithmisables. C'est là une critique des humanités numériques qui est certes théorique. Mais elle nous semble sérieuse parce que fondée sur les théories contemporaines de la computationnalité qui fondent la technologie informatique elle-même.

Ces problèmes ne sont évidemment pas toujours apparents dans la pratique concrète des humanités. Prenons par exemple l'opération si privilégiée dans les HN de la numérisation. Elle est, malgré des apparences contraires, traversée par cette problématique de la computationnalité, de la sémioticité et de la complexité. On accepte depuis des décennies maintenant de numériser des textes, des images, de la musique, des danses, des architectures, etc. Mais cette opération, si efficace soit-elle, ne peut éviter ces problèmes. Notamment, la discrétisation des objets qu'impose la numérisation implique des décisions importantes quant à la sémioticité

[13] On peut comprendre que la redondance intervient lorsqu'il devient nécessaire de répéter totalement ou en partie un élément pour diminuer le bruit qui aurait pu s'insérer dans le système.

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

omniprésente dans plusieurs moments du processus. Par exemple, si ces objets sont des textes, des décisions auront à être posées eu égard au choix des manuscrits, des éditions originales, de la précision du scanner, de leur diffusion, de leur destinataire, etc. Le processus de numérisation est traversé de choix théoriques, culturels, etc. Il y aura possiblement une différence entre l'objet l'original et de l'objet numérisé. Il y a bien une computation effective qui fut réalisée mais au prix de décisions qui auront affecté la sémioticité. Il ne faut donc pas se surprendre que les éditions techniques critiques soient prudentes face à la numérisation des textes.

On trouve aussi dans les HN des applications de fonctions computationnelles complexes, souvent spectaculaires, qui n'en sont pas moins confrontées à ces mêmes questions. Les technologies informatiques de l'information nous en présentent plusieurs telles la webisation, l'archivage, la fouille, la description structurale, statistique, la classification, la visualisation, la fouille, le forage, etc. Toutes ces opérations mettent en œuvre de multiples interventions dans la signification de ce sur quoi elles sont appliquées. Par exemple, il leur faudra prendre des décisions sur des paramètres qui déterminent ce qui est admissible à un traitement, ou encore il faudra resserrer des critères pour délimiter la portée des algorithmes, choisir un mode de présentation des résultats, etc. Bref, toutes les applications des modèles fonctionnels computationnels insèrent dans leur processus une couche particulière et complexe de médiations qui affectent la sémiotique de ce qui est « computé ». Quiconque a déjà analysé une visualisation (liste, graphes, réseaux, nuages) de mots extraits d'une base de données ou d'un texte voit vite la médiation mise en œuvre pour analyser les thèmes, les indices, les structures, etc.

Ce type de problème pourrait s'appliquer à la majorité des tâches que rencontrent les disciplines des humanités. Et pour être encore plus radical, nous dirons que, parce que le nombre des fonctions/opérations non computationnelles est infiniment plus grand que les fonctions/opérations strictement computationnelles, la probabilité que les humanités numériques rencontrent ce type de fonctions/opérations est évidemment très grande. Comment alors les projets pourront-ils traiter informatiquement ces cas ? La seule réponse possible à cette question sera similaire à celle que les autres disciplines donnent lorsqu'elles sont confrontées à des modèles fonctionnels où émerge le non computable. Et la réponse nous vient évidemment de Turing : ces systèmes, comme il le suggérait, doivent consulter des oracles.

Comment peut-on comprendre dans le cadre des HN ce rôle des oracles ? Récemment, Soare (1996, 2006, 2009, etc.), un des théoriciens importants de la théorie de la computation, proposa une interprétation originale de ces oracles, et des plus intéressantes pour notre propos. Il faut comprendre l'oracle, suggère-t-il, comme une *source d'informations* et de solutions ad hoc que les fonctions non computables peuvent consulter pour résoudre leur problème. Pour les

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

ordinateurs, Soare verra ces oracles comme des bases de données. Qui plus est, l'internet serait pour lui le prototype d'un tel oracle. Et pour les systèmes naturels, les oracles se trouveraient tout simplement dans l'environnement.

In real world computing the oracle may be called a database or an environment. " The oracle may be called a database or an environment. " (Soare, 2009: 41)

On peut traduire cette interprétation des oracles pour la positionner en regard du travail des chercheurs dans les humanités. En effet, s'il existe dans les humanités de nombreuses tâches qui sont intrinsèquement non-computables, on peut penser que leurs modèles fonctionnels devront eux aussi inclure une consultation d'oracles. Et comme le suggère Soare, ces oracles seront, pour les humanités, des sources complexes de formes sémiotiques dépositaires de telles informations. De fait, dans les humanités, ces formes sémiotiques ne sont pas si mystérieuses qu'elles le semblent. Nous dirions même qu'elles y sont omniprésentes. Elles prendront la forme *de texte, d'images, de cartes, de rituels, de chants, de films, des structures architecturales, etc.* Autrement dit, lorsque des modèles fonctionnels s'avèrent effectivement ou possiblement non computables, il faudra prévoir la possibilité de consulter des oracles constitués de formes sémiotiques où est déposé un savoir stabilisé et socialement partagé.

En fait, cet oracle n'est pas nouveau pour les humanités. Il est connu depuis des siècles. Son nom est Hermès. Et il a souvent joué ce rôle d'oracle! Et lorsqu'il s'est mis à travailler sur les objets qu'on lui remettait, on appelait son travail *interprétation* ou plus techniquement " herméneutique ". Ceci signifie plus concrètement que ces oracles sont des interprètes ou agents cognitifs, qui, eux-mêmes, possédant aussi un savoir pertinent, interviennent pour les assister dans la " calculabilité " des tâches à réaliser. Et ces agents-interprètes sont toujours sensibles à la dimension de signification en jeu dans la tâche à réaliser.

Ainsi, loin de disparaître, l'herméneutique renait dans le cadre computationnel. Hermès doit troquer la plume, le stylet, le papyrus, le papier pour l'écran d'ordinateur, le disque dur, la tablette. Il se réjouira probablement de retrouver ces vieux « nuages ». Il aura certes encore de nombreuses tâches qui, comme de vieux supports (exemples : de papier, de pierre, etc.) seront toujours réalisées avec un certain automatisme. Mais on peut penser que la majeure partie des tâches des humanités seront non-computables et ceci en raison soit de leur non-computationnalité intrinsèque soit de leur immense complexité.

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

Conclusion générale

Comme on le voit, le défi des humanités lorsqu'il opère dans un horizon informatique est immense. Nous avons tenté de montrer que, pour ce programme de recherche, la caractéristique d'être numérique est secondaire. La plus importante est la computationnalité. Dans cette perspective, le défi des humanités numériques ne consiste pas avant tout à numériser les artefacts si typiques, mais bien à identifier les tâches qui sont les siennes et à les traduire en des fonctions computationnelles. Or un tel travail n'est pas en soi facile à réaliser est n'est pas sans risque. Il n'en demeure pas moins au cœur de ce programme de recherche.

Aussi, il ne faut pas confondre le numérique avec le computationnel. En effet, ce n'est pas parce que des projets numérisent de plus en plus des objets sémiotiques complexes, au point d'en faire d'immenses reposoirs – des big data – que l'on trouvera facilement des algorithmes pour les traiter. Il n'est même pas assuré que ces algorithmes existent pour les tâches complexes qui traversent les humanités. Peut-être que les opérations effectuées dans les humanités parce qu'elles sont réalisées par des humains ou qu'elles portent sur des activités humaines sont tout simplement non computables.

Comme le dit Penrose : *Human intuition and insight... cannot be reduced to any set of computational rules. (Penrose, Shadows of the Mind :*

Une telle position ne serait pas originale. Même en mathématiques, tout n'est pas computable: " *undecidability and incompleteness are everywhere in mathematics " Chatin, Costa, Doria ; 2012 : xiii)*

Si tel est le cas, quels ajustements doivent faire les humanités humériques pour que le programme soit réaliste, réalisable, et ne tourne pas au vide ? Et il est essentiel que ce programme de recherche, original et novateur se donne de bons fondements théoriques. Sinon, ce « buzz word » " Humanités numériques " aux effets marketing des plus efficaces risque de lui jouer des tours. Lors de son apparition, il séduira certes par des prophéties irréalisables mais très vite, les véritables problèmes liés à la computationnalité et la non-computationnalité émergeront et le dirigeront vers des cul-de-sac. Le programme de recherche s'étiolera rapidement.

Qui plus est, il faut que Hermès trouve une alliance réaliste et efficace avec l'informatique. L'ordinateur devra apprendre à se positionner correctement dans la chaîne des opérations interprétatives réalisées dans les humanités et accepter de travailler avec Hermès. Il pourra, par exemple, offrir des banques d'algorithmes efficaces pour réaliser un grand nombre d'opérations typiques des humanités, mais surtout il devra offrir reposoirs d'oracles, qui seront utilisables dans les chaines de traitements. Il faudra toujours cependant voir ces oracles comme des assistants

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

dans les opérations interprétatives complexes qui dominent dans les humanités numériques. Reste cependant qu'un des meilleurs oracles sera toujours le lecteur humain lui-même, qui par sa culture, son expertise, finalisera les interprétations. Aussi cet humain, et dans certains cas le lecteur expert, devra-t-il avoir une place privilégiée dans les chaines de traitement interprétatif.

Se rappeler l'omniprésence de Hermès et lui attribuer un rôle correct dans les tâches des humanités numériques est la seule manière de garantir le sérieux de ce programme de recherche. Et si l'ordinateur y a une place ce sera justement celui d'être un assistant dans l'interprétation et non l'inverse. Dans une telle vision des choses, ce n'est pas de l'informatique que naitra un nouvel humanisme. Au contraire c'est de l'humanisme assurément le plus classique que peut-être apparaitra une adaptation de l'ordinateur à l'humain. Il n'existera jamais d'interprète automatique. Mais un bon interprète apprendra à être assisté d'un ordinateur… et non dans toutes ses tâches. Fort heureusement d'ailleurs!

Bibliographie

CHAITIN, Gregory John, " Information Randomness and Incompleteness ", *Scientific American*, n°232, Mai 19751.

CHAITIN, Gregory John, *The Limits of Mathematic*s, Springer, 1998.

CHAITIN, Gregory John ; DA COSATA Newton et DORIA Francisco Antonio, " Godel's Way Exploits into an undecidable world ", Taylor & Francis Group / Balkema Book, Londres (UK), 2012.

CHURCH, Alonzo, " An unsolvable problem of elementary number theory ", *American Journal of Mathematics*, n° 58, 1936, p. 345-363.

CURRY, Brooks Haskell et FEYS, Robert, *Combinatory Logic*, Vol. I., Amsterdam, North Holland Publishing Compagny, 1958.

DACOS, Marin et MOUNIER, Pierre, *Les Humanités numériques : état des lieux et positionnement de la recherche française dans le contexte international*, Paris, Inst. Français., 2014.

DAVIS, Martin, *Computability and Unsolvability*, New York, McGraw-Hill, 1958.

DE BUSA, Roberto, " Foreword : Perspectives on the Digital Humanities " in SCHREIBMAN Susan, SIEMENS Ray, UNSWORTH John, *A Companion to Digital Humanities*, Wiley-Blackwell, 2008.

DENNETT, Daniel, *Brains Storms*, Cambridge, MIT Press, 1978.

DOUEIHI, Milad, *Pour un humanisme numérique*, Paris, Seuil, 2011.

DOUEIHI, Milad, *Qu'est ce que le numérique ?*, Paris, PUF, 2014.

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

FEFERMAN, Solomon, " The impact of the incompleteness theorems on mathematics ", *Notices American Mathematic Society*, n°53, April 2006, p. 434-439.

FODOR, Jerry and ZENON W. Pylyshyn, " Connectionism and Cognitive Architecture : A Critical Analysis, " In *Connections and Symbols*, ed. Pinker, Steven and Jacques Mehler, Cambridge, MIT Press, 1988.

HARNAD, Stevan, " The Symbol Grounding Problem ", Physica, n°42, 1990, p. 335-346.

KIRSH, Adan, *The false promise of the digital humanities New Republic*, 2014, [url : http://www. newrepublic. com/article/117428/limits-digital-humanities-adam-kirsch] (consulté le 19/10/2014).

LACOMBE, Daniel, " La théorie des fonctions récursives et ses applications ", *Bulletin de la Société Mathématique de France*, n°88, 1960, p. 393-468.

MARKOV, Andreï, " The Theory of Algorithms ", *American Mathematical Society Translations*, series 2, n°15, p. 1-14, 1960.

MARR, David, *Vision : A Computational Investigation into the Human Representation and Processing of Visual Information*, New York, W. H. Freeman and Company, 1982.

McCARTY, Willard, " Modeling : A Study in Words and Meanings " in SCHREIBMAN Susan, SIEMENS Ray, UNSWORTH John, *A Companion to Digital Humanities*, Wiley-Blackwell, 2008.

METRAUX, Alfred, *Ethnologie de l'Île de Pâques*, Paris, Gallimard, 1935.

METRAUX, Alfred, *L'Île de Pâques*, Paris, Gallimard, 1941, coll. "Idées".

MEUNIER, Jean-Guy, *CARAT–Computer-Assisted Reading and Analysis of Texts : the appropriation of a technology Digital Studies*, 2009 [url :http://www.digitalstudies.org/ojs/index.php/digital_studies/article/view/161] (consulté le 19/10/2014).

MEUNIER, Jean-Guy, *Computor and computers : Epistémologia*, XXXVI, 2013, p. 18-36.

PITTS, Mike, MILES, James, PAGI, Hembo and EARL, Graeme, " Hoa Hakananai'a : a new strudy of an easter Island statue " in the *British Museum The Antiquaries Journal*, n°94, 2014, p. 291-321.

MINSKY, Marvin, " A framework for representing knowledge ", in *The psychology of Computer vision*, P. Winston (Ed.), McGraw Hill, 1975.

RUMELHART, David, McCLELLAND, James and the PDP Research Group, *Parallel Distributed Processing : Explorations in the Microstructure of Cognition*, Cambridge, MIT Press, 1986.

ORD, Toby, " The many of hypercomputation ", *Applied Mathematics and Computation* n°178, 2006, p. 143–153.

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121

PENROSE, R., *Shadows of the Mind : A search for the Missing Sicence of Consciousness*, Oxford University Press, 1994.

POST, Emil, " Recursively enumerable sets of positive integersand their decision problems ", *Bulletin of the American Mathematical Society*, n°50, 1944, p. 284–316.

PYLYSHYN, Zenon, *Cognition and computation*, Cambridge, MIT Press, 1984.

RAMSEY, Stephen, " Who'in who'out " in VANHOUTTE, NYHAN, Julianne, TERRAS, Melissa, *Defining Digital Humanities*, Ashgate Publishing, 2011.

ROGERS, Hartley, *Theory of Recursive Functions and Effective Computability*, New York, McGraw-Hill, 1967.

SCHREIBMAN, Susan, SIEMENS, Ray, UNSWORTH, John, *A Companion to Digital Humanities*, Wiley-Blackwell, 2008.

SEARLE, John," Minds, brains and programs ", *Behavioral and Brain Sciences*, n°3, 1980, p. 417-57.

SEARLE, John, The Rediscovery of the Mind, Cambridge, MIT Press, 1992.

SHAGRIR, Oron, " Effective Computation by Humans and Machines ", *Minds and Machines*, n°12, p. 221–240, 2002.

SOARE, Robert, " Turing Oracle Machines, Online Computing, and Three Displacements " in *Computability Theory, Annals of Pure and Applied Logic*, 160(3), 2009, p. 368-399.

TURING, Alan, " On computable numbers with applications to the entscheidung–problem ", *Proceedings of the London Mathematical Society*, n°42, 1936, p. 230–265.

TURING, Alan, " Systems of Logic Based on Ordinals ", *Proceedings of the London Mathematical Society* (Series 2), n°45, 1939, p. 161–228.

Publication de l'article en ligne : 2014/12
http://www.sens-public.org/spip.php?article1121