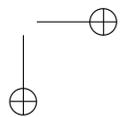
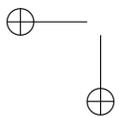
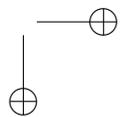
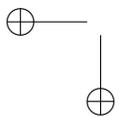
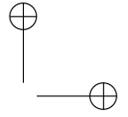
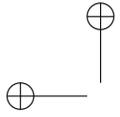


Coline Ferrarato

Philosophie du logiciel  
Dialogue entre Simondon et un objet technique  
numérique





**Coline Ferrarato**

**Philosophie du logiciel**  
**Dialogue entre Simondon et**  
**un objet technique numérique**

Collège Eötvös József ELTE  
Budapest, 2018

Philmaster,  
*sous la direction de Mathias GIREL et Éric GUICHARD*  
École des hautes études en sciences sociales,  
École normale supérieure, Juin 2017

Responsable de l'édition : Dr. László Horváth,  
Directeur du Collège Eötvös József ELTE  
Rédacteur : Gyula Mayer  
Conception graphique : Emese Egedi-Kovács  
© Collège Eötvös József ELTE  
© L'auteur

Tous droits de traduction et de reproduction réservés.

Imprimé en Hongrie par Pátria Nyomda Zrt.  
1117 Budapest, Hunyadi János út 7.  
Directeur : Simon László  
ISBN 978-615-5897-06-1

## Remerciements

A Mathias GIREL et Éric GUICHARD, qui m’ont appris à poser les bonnes questions (et à tenter d’y répondre).

A Mete DEMIRCIGIL, compagnon de navigation en eaux troubles.

A toutes celles et ceux qui m’ont initié aux méandres de l’informatique avec pédagogie et bienveillance :

Elie MICHEL, pour la ”découverte” de la marge d’indétermination,

François TAIANI et Davide FREY, pour leurs explications pointues sur les navigateurs,

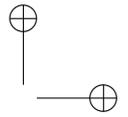
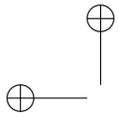
Mathieu et Lunar, pour leur extrême disponibilité et la découverte du monde de la programmation,

Rémi HUBSCHER, pour les explications sur le fonctionnement de Mozilla Firefox,

Emmanuel ST-JAMES, pour la rigueur intellectuelle et les conversations,

Hellekin, pour les Rencontres Mondiales du Logiciel Libre,

Stéphanie OUILLO, pour la disponibilité et les perspectives sur les études de genre en informatique.



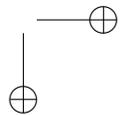
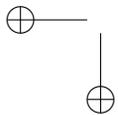
6

---

A celles et ceux qui m’ont éclairé sur Simondon : Irlande SAURIN et Jean-Yves CHATEAU.

A Alicia BASSO BOCCABELLA, Juliette FLEURANT, Lucie LESZEZ, Lucile MARION, qui ont beaucoup trop entendu parler de ce mémoire.

A Pascale, Dino et Léo FERRARATO.



# Sommaire

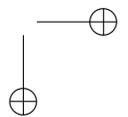
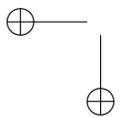
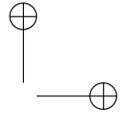
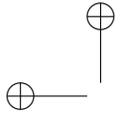
<b>Introduction</b>	<b>11</b>
<b>I La philosophie de la technique de Simondon comme programme de travail</b>	<b>25</b>
I.A Une philosophie de la technicité . . . . .	26
I.A.1 Le fonctionnalisme simondonien . . . . .	29
I.A.2 La question de la localisation de la technicité . . . . .	38
I.A.3 La question des représentants de la technicité . . . . .	45
I.B La méthode simondonienne : au plus près de l’objet technique . . . . .	50
I.B.1 Les enjeux épistémologiques : une méthode inductive . . . . .	51
I.B.2 Etude de cas d’un exemple techno- logique . . . . .	54
I.B.3 Reproduire le geste simondonien . . . . .	59
I.C Confronter la pensée de Simondon à l’in- formatique . . . . .	61
I.C.1 Etat des travaux sur Simondon et l’informatique . . . . .	61
I.C.2 Positionnement de l’étude . . . . .	69

<b>II Etude de technologie génétique : le logiciel est-il un objet technique ?</b>	<b>71</b>
II.A Définition et problématisation de l’objet numérique . . . . .	71
II.A.1 L’objet technique pour Simondon .	73
II.A.2 Le navigateur, objet numérique représentatif du logiciel . . . . .	75
II.B Construire le logiciel à partir de la marge d’indétermination . . . . .	88
II.B.1 La machine-ordinateur et la marge d’indétermination . . . . .	89
II.B.2 La complexification du code informatique . . . . .	99
II.B.3 Trois hypothèses concernant le statut du logiciel . . . . .	104
II.C Les niveaux de technicité logiciels . . . . .	105
II.C.1 La genèse du navigateur . . . . .	107
II.C.2 L’élément, le milieu associé . . . . .	109
<b>III Etude psychosociale du logiciel libre</b>	<b>117</b>
III.A Le problème de l’objet technique industriel	120
III.A.1 La question de la commensurabilité de la technique . . . . .	121
III.A.2 La double aliénation des objets techniques industriels . . . . .	123
III.A.3 Sauver l’objet technique en le des-tituant . . . . .	131
III.B La promesse d’ouverture du logiciel en tant qu’objet technique post-industriel . . .	134
III.B.1 Un système complexe . . . . .	135
III.B.2 Une configuration post-industrielle .	137
III.B.3 Le logiciel libre, garant de la tech-nicité logicielle . . . . .	141

SOMMAIRE 9

---

III.C Bricoler l’objet technique numérique . . . .	146
III.C.1 Enjeux et extension du concept de bricolage . . . . .	147
III.C.2 Le bricolage informatique . . . . .	157
<b>Conclusion</b>	<b>169</b>
<b>Annexes</b>	<b>179</b>
<b>Glossaire</b>	<b>193</b>



## Introduction

*« La plus forte cause d’aliénation dans le monde contemporain réside dans cette méconnaissance de la machine, qui n’est pas une aliénation causée par la machine, mais par la non-connaissance de sa nature et de son essence, par son absence du monde des significations, et par son omission dans la table des valeurs et des concepts faisant partie de la culture. »*

*– Simondon, Le mode d’existence des objets techniques, Introduction.*

Un jour, alors que nous voulions reprendre la rédaction de ce travail, la page blanche de notre logiciel de traitement de texte a refusé de « s’ouvrir ». La licence dudit logiciel avait expiré. Elle ne pouvait être renouvelée sans paiement, et nos travaux écrits étaient gardés en otage. Cette situation est une expérience-limite. Elle montre que l’objet numérique « page blanche » n’est pas entièrement ce qu’il semble être – autrement dit, que notre perception directe ne suffit pas pour juger de l’identité de l’objet. Nous sommes dupes de l’apparence que des développeurs logiciels ont choisi de donner à leur service de traitement de texte : celle d’une feuille blanche<sup>1</sup>.

---

1. En effet, le traitement de texte sur un ordinateur peut revêtir

Ce qui est présenté comme une simple chose est en fait une interface, c'est-à-dire l'image qu'on a choisi de donner à un mécanisme technique complexe et sous-jacent. L'expérience-limite révèle ainsi notre inculture quant à ce support que l'on avait, par facilité, pris pour allié. Le travail que l'on y inscrit est conditionné par le paiement de la licence ; il est aliéné, ne nous appartient pas en propre.

Un tel déboire est commun parmi les universitaires. Il est symptomatique du double visage de ce que l'on nomme « numérique » : une nébuleuse d'objets techniques et de protocoles qui s'avancent masqués.

En son sens technique général, le numérique renvoie à l'ensemble des appareils réticulés dont l'information échangée est réductible à du langage binaire<sup>2</sup>. Cela correspond à tous les terminaux connectés au réseau Internet, ou à d'autres réseaux<sup>3</sup>. L'informatique, dont nous traitons dans cette étude, est une des branches du numérique<sup>4</sup>.

Le versant informatique du numérique est problématique en ce qu'il recouvre une réalité sociale et technique qui ne se recoupent que très peu. Une telle définition est propre à notre époque : notre rapport aux ordinateurs a évolué historiquement. Lors du début de leur diffusion, dans les années soixante-dix, une partie des mouvements contestataires de la gauche américaine rejetait les ordinateurs comme symboles de la bureaucratie et de l'ordre établi<sup>5</sup>. Les étudiants craignaient l'appa-

---

d'autres formes que celles de la feuille blanche : fenêtre blanche avec un simple curseur, fenêtre noire dans certains cas, etc.

2. Voir glossaire, "Code informatique, code source".

3. Réseaux plus locaux, à l'échelle d'une entreprise, par exemple.

4. Voir glossaire, "Numérique/Informatique".

5. Ainsi, Fred Turner [103] cite le discours de l'étudiant Mario

rition d’imposantes machines<sup>6</sup>, dont le développement était intimement lié à la seconde guerre mondiale et aux affrontements de la guerre froide. Cet exemple états-unien montre que l’informatique, à ses débuts, faisait partie du débat public ; il était sujet à réflexion pour la culture.

Avec la miniaturisation des ordinateurs et leur commercialisation à grande échelle<sup>7</sup>, les discussions autour

---

Savio prononcé à Berkeley le 2 décembre 1964, contre le président du Conseil des Régents de l’université qu’il considérait comme un “gérant” : “Arrive le jour où le calcul de la machine devient si odieux, vous donne tellement la nausée que vous ne pouvez plus en être, vous ne pouvez plus tacitement en être l’un des opérateurs. C’est alors que vous devez peser de tout votre corps sur ses engrenages, ses rouages, ses manettes et toute sa mécanique. Vous devez l’arrêter coûte que coûte. Et vous devez donner à entendre aux machinistes et aux propriétaires que leur machine ne sera remise en état que lorsque vous aurez retrouvé la liberté.” Une telle prise de position est représentative du militantisme contre-culturel des années soixante aux Etats-Unis. F. Turner en analyse les deux tenants, qui se sont tous deux positionnés face à l’informatique : la “nouvelle Gauche” et les “nouveaux communalistes”.

6. Les années soixante sont pour l’informatique une période de transition entre l’ère des gros systèmes et celle de leur miniaturisation progressive. Les gros systèmes ont initié, depuis le début des années cinquante, d’importants programmes de recherches et commencent à être commercialisés. Ils sont également liés au complexe militaro-industriel : ainsi, le Whirlwind, développé au Massachusetts Institute of Technology en 1951, était relié à des stations radars pour alerter la défense américaine en cas d’attaque aérienne russe. Ce qui s’appelait à l’époque des mécanographes remplissait des pièces entières, et était totalement inaccessible au grand public ; un ensemble mécanographique valait quatre millions d’anciens francs, l’équivalent de quatre Citroën DS. [71]

7. La période de miniaturisation des ordinateurs s’ouvre à la fin des années soixante. Le premier ordinateur miniature ayant connu un succès commercial est l’Apple II (Altair 8800), conçu en 1975. Ce dernier initie une longue lignée d’ordinateurs de plus en plus compacts.

des enjeux soulevés par la technicité de l’informatique ont peu à peu disparu du débat public. Les ordinateurs n’étaient plus d’imposantes machines et se présentaient de plus en plus comme des auxiliaires du quotidien, objets de marketing et de désir. La réalité sociale de l’informatique aujourd’hui est la feuille blanche : nous côtoyons les ordinateurs au quotidien, pour leur usage, sans questionner les dessous de leur apparence. En ce sens, le numérique<sup>8</sup> est un « milieu technique »<sup>9</sup> dans lequel nous baignons, avec lequel nous entretenons un rapport conditionné par l’habitude.

Le numérique ne peut se définir uniquement par des critères techniques. Il doit être compris au prisme de la tension interne qui le traverse. Il s’agit d’un système technique extrêmement lourd que l’on a peu à peu réduit au rapport quotidien de l’usage, sans en mesurer tous les enjeux. Ce système technique est devenu un impensé de notre culture.

Notre époque est prise au piège d’un paradoxe majeur. Alors que la technicité numérique est éminemment pervasive<sup>10</sup> et structure notre existence, la plupart d’entre nous sommes des « illettrés » du numérique<sup>11</sup>. Il nous est impossible de démonter les machines que nous utilisons au quotidien, ou de comprendre les lignes de code de nos logiciels habituels. Soixante ans plus tard, le diag-

---

8. Nous utiliserons dans la suite de cette introduction le terme « numérique » et « informatique » de façon indifférenciée, puisque nous avons spécifié au préalable que l’expression « numérique » fait référence à une de ses branches, celle de l’informatique.

9. Voir G. Friedmann, *Sept études sur l’homme et la technique*. [54]

10. Voir B. Bachimont, “Arts et sciences du numérique : ingénierie des connaissances et critique de la raison computationnelle”. [4]

11. Voir E. Guichard, “Culture numérique, culture de l’écrit”. [61]

nostic de l’introduction du *Mode d’existence des objets techniques*<sup>12</sup> est toujours d’actualité. Rédigé en 1958, l’ouvrage, thèse secondaire de l’auteur Gilbert Simondon, souhaitait « susciter une prise de conscience du sens des objets techniques »<sup>13</sup>. Une telle prise de conscience était rendue nécessaire suite au rejet des sphères culturelles face à la réalité technique. Simondon s’élève dans son introduction contre l’hypocrisie d’une culture qui, de plus en plus dépendante de la technique, la traite comme une « réalité étrangère ».<sup>14</sup> Tout le but de son ouvrage est de conférer une dignité ontologique aux objets techniques, afin de les réconcilier avec la culture. C’est, selon lui, un rôle qui incombe à la pensée philosophique. Pour ce faire, l’auteur déploie une approche particulière. Il s’intéresse aux machines en elles-mêmes, et tente d’établir leur mode d’existence en s’appuyant sur la biologie ; il participe en cela d’un « axe naturaliste »<sup>15</sup> qui le situe par rapport aux auteurs de son époque dans la continuité d’A. Leroi-Gourhan et de J. Laffitte<sup>16</sup>. La philosophie de la technique de Simondon est avant tout fonctionnaliste : un objet est un objet technique s’il *fonctionne*. La particularité irréductible d’existence au monde d’un objet technique est son fonctionnement, qui se traduit par une genèse et un processus de concrétisation.

12. Noté à présent MEOT. [90]

13. MEOT, première phrase de l’introduction.

14. MEOT, *ibid.*

15. La classification est de R. Le Roux [72], qui oppose ainsi Simondon à l’axe dit « formaliste » (représenté notamment par Babbage et Reuleaux et par Couffignal et Riguet en France), axe axiomatisant dans un langage ou en algèbre les mécanismes et éléments qui constituent les machines.

16. Pour une mise en perspective des thèses de Simondon par rapport aux autres thèses de la philosophie de la technique classique, voir I.A.1.

La mécanologie de Simondon est également un dialogue direct avec la cybernétique de Wiener<sup>17</sup> - ce qui le rapproche des problématiques informatiques. Simondon n'a pas pensé ces dernières frontalement, même s'il avait conscience de leur développement. L'introduction du MEOT prend pour exemple les « machines à calculer »<sup>18</sup>, et le glossaire de ce dernier renvoie à une entrée "Basculeur" mentionnant le circuit Eccles-Jordan<sup>19</sup>. Quelques autres réflexions englobant l'informatique apparaissent dans son oeuvre, mais sont peu nombreuses<sup>20</sup>. Les ordinateurs auxquels fait allusion Simondon sont encore des machines à calculer assez peu connues du grand public : il s'agit de grands ensembles mécanographiques qui produisent des calculs<sup>21</sup>. Pourtant, et ce dès l'introduction du MEOT, l'analyse des « machines à marge d'indétermination » laisse entendre que l'auteur a laissé des cadres conceptuels adéquats pour penser notre réalité technique contemporaine.

17. Voir N. Wiener, *Cybernetics*. [106]

18. MEOT, p.13 : "Les machines à calculer modernes ne sont pas de purs automates ; ce sont des êtres techniques qui, par-dessus leurs automatismes d'addition (ou de décision par fonctionnement de basculeurs élémentaires), possèdent de très vastes possibilités de commutation des circuits, qui permettent de coder le fonctionnement de la machine en restreignant sa marge d'indétermination."

19. Il s'agit du premier système de basculeur électronique, qui est à la base du codage binaire.

20. Ainsi, le mot "ordinateur" apparaît deux fois dans le cours *L'invention dans les techniques* Des réflexions sur le calcul et le codage apparaissent dans "Art et nature (La maîtrise technique de la nature" et dans "Trois perspectives pour une réflexion sur l'éthique et la technique", dans le recueil *Sur la technique* [93]. Le cours *Imagination et invention* [92] fait quant à lui référence aux données et aux règles des machines complexes.

21. Les fonctionnalités des ordinateurs commenceront à se diversifier lors de leur miniaturisation, dans les années soixante-dix.

Cette étude se propose suivre les pistes laissées par Simondon pour penser une réalité technique qui nous est contemporaine. Puisque la culture continue de se poser en « système de défense contre les techniques »<sup>22</sup>, il faut adresser à notre époque les mêmes questions que celles que Simondon adressait à la société des années soixante. Que sont nos objets techniques ? Répondre à une telle question impliquait de délimiter un périmètre précis dans la pensée de l’auteur : celui de sa philosophie de la technique. Impossible alors de ne pas prendre en compte l’histoire éditoriale mouvementée de l’oeuvre simondonienne<sup>23</sup>. Il y a quelques années encore, on ne connaissait de la philosophie de Simondon que sa thèse secondaire, le *Mode d’existence des objets techniques*<sup>24</sup>. Depuis une dizaine d’années cependant, le corpus simondonien s’est élargi de nouveaux textes<sup>25</sup>, qui permettent

22. MEOT, p.1.

23. Les deux thèses de Simondon sont *L’individuation à la lumière des notions de forme et d’information* (noté à présent ILFI, thèse principale) [91] et le *Mode d’existence des objets techniques* (thèse secondaire). Elles sont toutes deux soutenues en 1958, mais seul le MEOT est publié à cette date. La thèse principale connaît une histoire plus mouvementée : en 1964 en est publiée une première partie, *L’individu et sa genèse physico-biologique* (IGPB), et en 1989 une seconde, *L’individuation psychique et collective* (IPC). Elle n’est publiée dans son entièreté qu’en 2005 sous son titre original. Le reste des cours, articles et interventions est publié peu à peu, au cours des années 2000 ; ainsi, la dernière publication posthume de Simondon date de 2016 ; il s’agit du recueil de cours *Sur la philosophie*, chez PUF, regroupant des interventions de 1950 à 1980. [99]

24. Les spécialistes de Simondon avaient une connaissance plus générale de l’œuvre, mais pour le néophyte, le manque de travail d’édition conduisait à des lacunes quant à la philosophie de l’auteur.

25. Ainsi, chez PUF, la série de volumes regroupant des textes inédits : *Sur la technique* (2014) [93], *Sur la psychologie* (2015) [98] et *Sur la philosophie* (2016). [99]

une réévaluation globale de l’œuvre<sup>26</sup>. De nouvelles correspondances peuvent ainsi être tracées pour éclairer la pensée du philosophe.

Nous avons fait le choix de nous appuyer sur deux ouvrages : le *Mode d’existence des objets techniques* et la *Psychosociologie de la technicité*<sup>27</sup>. Notre corpus se situe dans cette perspective générale de relecture de l’œuvre à la lumière des « nouveaux » écrits - la PST est un cours qui n’est paru que récemment. Le dialogue de ces deux livres permettra de redéfinir la philosophie de la technique de Simondon, en l’élargissant à la méthode psychosociale. Notre hypothèse de départ est qu’ils forment un tout cohérent : par leurs échos théoriques et leur complémentarité, mais également par les aspects problématiques que leur mise en regard soulève.

La philosophie de la technique de Simondon est un constat : celui du divorce entre la technique et de la culture. C’est un cadre conceptuel défini par le dialogue des deux ouvrages que nous avons sélectionnés. C’est enfin, en dernier recours et avant tout, l’emploi d’une méthode particulière qui a valu à la publication du MEOT un retentissement dans la sphère universitaire.

26. La parution de la thèse primaire dans son entièreté, en 2005, en est le meilleur exemple. De nombreuses monographies ont insisté sur la nécessité de lire le MEOT en ayant en tête la thèse de L’individuation à la lumière des notions de forme et d’information. J-H. Barthélémy, dans *Simondon ou l’encyclopédisme génétique* [10], mène une réflexion exemplaire en ce sens. Tout l’ouvrage est dirigé vers la philosophie génétique de la thèse principale, et analyse en dernier lieu le *Mode d’Existence* à la lumière de cette dernière.

27. Notée à présent PST. Le MEOT a été soutenu en 1958, et est paru la même année, tandis que la PST est un cours donné en 1960-1961, et n’est paru qu’en 2014 chez PUF, dans le recueil de cours inédits *Sur la technique*. [93]

John Hart en explique la cause dans la préface du livre : « En tant qu'étude universitaire exposant la réalité humaine contenue dans la machine, il était unique en son genre dans tout le corpus philosophique concernant ce sujet, c'est-à-dire qu'il n'y avait encore rien qui associe un traitement philosophique à une telle proximité de l'objet technique. »<sup>28</sup> En dépit de tout impératif académique, Simondon insère dans sa thèse complémentaire de longs développements sur des moteurs, des diodes et des triodes ; sur des objets techniques concrets. C'est en ce sens que l'on peut parler de geste simondonien dans la première partie du MEOT : le philosophe postule théoriquement la dignité ontologique des objets techniques et, simultanément, la démontre en leur donnant la parole en tant qu'arguments à part entière. Ce que nous appelons méthode simondonienne est ce geste fort d'insertion de l'objet technique comme argument dans le raisonnement discursif.

Malgré son statut radical dans le champ de la philosophie de la technique, cette dernière fait pourtant l'objet d'un vide historiographique. De nombreuses études sur Simondon en mesurent les enjeux<sup>29</sup>, mais rares sont celles

28. J. Hart, Préface du MEOT.

29. Les approches à ce propos diffèrent, mais elles sont toutes périphériques et non épistémologiques à proprement parler. On peut évoquer à ce sujet, du plus épistémologique au plus général :

-L'étude comparée de G. Carrozzini confrontant la mécanologie de Simondon à celle de Lafitte [32], mais sans toutefois analyser dans le détail les particularités argumentatives de la méthode simondonienne.

-Les grandes monographies explicitant la technologie simondonienne en détaillant certains de ses exemples : celle de P. Chabot [33] fait par exemple appel à des exemples techniques déployés par Simondon pour appuyer le commentaire de la première partie du MEOT.

qui ont étudié en détails le déploiement des exemples dans la pensée de l’auteur. Un tel vide historiographique fait écho à un trait paradoxal des études simondoniennes ; le geste de Simondon a été commenté et loué pour sa pertinence, mais non reproduit. Ce geste se voulait pourtant l’initiateur d’une longue série de philosophies réconciliées avec la réalité de leur objet.

Il ressort des études menées sur Simondon et l’informatique<sup>30</sup> que « l’ambition d’appréhender techniquement l’objet informatique à partir de Simondon réclame une redéfinition du schème technique inséparable d’une étude minutieuse des objets techniques en eux-mêmes.<sup>31</sup> Comprendre le numérique à l’aune de Simondon implique de mettre les schèmes techniques simondoniens à l’épreuve d’un objet technique. Reproduire la méthode simondonienne nécessite de la théoriser réflexivement, puis de faire l’effort de la mettre en application à un domaine qui est *a priori* étranger au philosophe, celui de la technique informatique.

C’est pourquoi nous avons décidé, pour faire dialoguer Simondon et l’informatique, de présenter un objet technique concret, et d’en étudier le fonctionnement. L’objet

---

-Les articles qui traitent de la méthode simondonienne et des schémas notamment sous l’angle plus général de la problématique encyclopédique, comme Bontems dans son article “Encyclopédisme et crise de la culture”. [22]

Une approche non directe, mais épistémologique, et sur laquelle nous nous appuyerons, est celle que déploie Barthélémy pour comprendre l’effort inductif de Simondon dans sa philosophie génétique (ILFI) à partir de Bachelard, dans *Simondon ou l’encyclopédisme génétique* notamment. [10]

30. Que nous commenterons dans la partie I.C. Il s’agit principalement d’études grammaticales.

31. J. Grosman, « Simondon et l’informatique II ». » [58]

devait être issu du champ numérique précédemment décrit, qui recouvre sous le même nom un panel de réalités hétérogènes. Le logiciel nous est alors apparu comme un point cardinal de la technique numérique contemporaine, en tant que programme indiquant à la machine-ordinateur les actions qu'elle se doit d'effectuer<sup>32</sup>.

Reproduire le geste de Simondon impliquait de reproduire sa méthode, c'est-à-dire d'entretenir avec notre objet d'étude une proximité assez grande pour que ce dernier puisse informer et être informé par les catégories d'analyse du philosophe. Il nous fallait donc sélectionner un exemple qui puisse illustrer notre analyse du logiciel dans sa généralité. Nous avons choisi le navigateur web<sup>33</sup>, en tant qu'il s'agit d'un logiciel complexe et central pour tout utilisateur du Web<sup>34</sup>; c'est à la fois la technicité pointue et l'aspect nodal d'un tel objet numérique qui nous a incité à l'analyse. Nous avons opté pour le navigateur Mozilla Firefox<sup>35</sup> car ce dernier est un logiciel libre<sup>36</sup>: il nous était donc possible d'accéder à son code source et il présentait des formes de production technique intéressantes.

Nous avons pris le parti de nous familiariser avec le fonctionnement de l'objet technique étudié par le biais de l'écriture du code et d'entretiens avec des programmeur·euse·s<sup>37</sup>. Il nous fallait comprendre de l'intérieur le

32. Renvoi au début de la partie II.

33. Voir partie II.A.2 pour une définition du navigateur.

34. Voir glossaire, "Web/Internet".

35. Nos exemples sont tirés de la version 53.0.3 du navigateur, parue le 22.05.17. Mozilla Firefox n'est pas le seul navigateur libre; nous l'avons également choisi pour sa popularité et l'importance des documents produits éclairant son fonctionnement.

36. Voir glossaire, "Logiciel (du point de vue de sa production)".

37. Bien que Simondon considère "la Femme" comme faisant par-

fonctionnement technique du logiciel (au prisme du navigateur web) de la même façon que Simondon étudiait les objets techniques de son époque. Dans un second temps, il s’agissait de faire émerger un tel savoir dans le discours réflexif de la philosophie, tout en le rendant accessible au plus grand nombre : à l’instar de Simondon, nous avons donc inséré des outils pédagogiques permettant à un public de non technicien·ne·s de comprendre les arguments techniques. Nous espérons ainsi que ce travail permettra à qui le lira de sensibiliser aux problématiques numériques tout en permettant une prise de recul critique et réflexive sur ce dernier.<sup>38</sup> On pourra se référer, à la fin de l’ouvrage, à un glossaire des termes principaux, à une table des figures, ainsi qu’à une bibliographie thématique. Pour que ce travail soit crédible, il fallait que sa forme en exemplifie le fond ; il fallait que l’injonction de l’étude concrète d’un objet et l’appel à la pédagogie soit doublés de leur réalisation effective.

L’intérêt de l’étude du logiciel réside en ce que sa configuration technique très particulière ne pouvait être envisagée par Simondon<sup>39</sup>. Se pencher sur un objet

---

tie d’un “sous-groupe dominé” (PST, p.46), dont l’étude est menée après celle de l’enfant et avant celle du “groupe rural”, nous avons décidé d’utiliser une écriture inclusive pour qualifier les personnes travaillant autour de l’informatique. Beaucoup de femmes ont eu et ont un rôle important dans l’histoire de l’informatique. Il ne nous semblait pas pertinent de différencier les rapports de l’homme et ceux de la femme quant à l’objet technique - et encore moins de traiter les femmes, donc la moitié de l’humanité, comme un “sous-groupe”.

38. Le travail d’annexe avait également pour avantage d’alléger le corps du raisonnement de longues définitions préliminaires.

39. Puisque les ordinateurs présents à son époque ne dissociaient pas le *hardware* du *software* : une telle dissociation s’est effectuée à la

numérique qui n’était pas constitué en tant que tel à l’époque du philosophe permet de jeter une lumière renouvelée sur ses catégories d’analyse. Le logiciel questionne directement la notion d’objet technique chez Simondon. Ce dernier s’attache en effet à étudier, on l’a vu, des machines et des objets qui sont exclusivement matériels : or le logiciel, considéré comme objet numérique, n’est pas à proprement composé de matérialité, mais d’information binaire.

Dans ce cadre, les catégories d’analyse simondoniennes permettent-elles de considérer le logiciel comme un objet technique ? Autrement dit, peut-on radicaliser le fonctionnalisme simondonien au point de lui faire assumer la pensée d’un objet technique immatériel ?

Une telle question revêt un triple enjeu. Au niveau de la philosophie de la technique en général, elle permettrait, via le fonctionnalisme de Simondon, de mettre sur pied la définition d’un objet technique numérique qui n’ait pas (entièrement) besoin de matérialité pour être considéré comme un objet technique. Cela serait le signe d’une conception de la technicité modifiée, en partie détachée de la matérialité. Au niveau de la philosophie simondonienne, réussir à édifier la définition d’un objet numérique immatériel aurait des conséquences quant à la seconde facette de la philosophie de la technique de Simondon, à savoir son aspect psychosocial ; la configuration particulière du logiciel viendrait en modifier les contours. Enfin, notre objet d’étude, le logiciel, pourra être remis en perspective grâce au prisme simondonien.

Notre cheminement sera le suivant : effectuer de nouveau le geste de Simondon est un parti-pris risqué et

---

fin des années 1970. Voir à ce propos partie III.B.3.

demandait une fréquentation assidue de l’auteur, pour éviter de le détourner. Il nous faudra définir, par un commentaire précis, le double cadre de ce que nous nommons « philosophie de la technique de Simondon » : cadre conceptuel (philosophie de la technique génétique et psychosociale) et cadre méthodique (démonstration par l’exemple). Nous tenterons ensuite d’appliquer ce cadre à l’objet numérique que nous avons sélectionné – le logiciel (et l’exemple qui nous servira de fil rouge, le navigateur web), tant sur le plan de la technologie génétique (II) que sur celui de la psychosociologie (III).

Plutôt que de commenter l’auteur, nous avons décidé de reproduire son geste. Tout notre travail s’efforce de correspondre au programme sous-jacent de l’affirmation de Deforge, dans la postface du MEOT<sup>40</sup> :

*« Notre conclusion : rééditer Simondon c’est bien. Avoir beaucoup de Simondons ce serait encore mieux. »*

---

40. DEFORGE Y., Postface du MEOT, fin de la huitième question ouverte, p.325.

## Partie I

# La philosophie de la technique de Simondon comme programme de travail

Reproduire le geste simondonien implique de se situer clairement vis-à-vis de sa philosophie de la technique. Le premier temps de notre réflexion est donc une entreprise de caractérisation conceptuelle de la philosophie de Simondon à partir du dialogue entre les deux ouvrages de notre corpus. Notre propos, dans sa généralité la plus grande, est le suivant : la philosophie de la technique de Simondon est une philosophie de la technicité et du geste méthodique pour mettre à jour cette technicité.

## I.A Une philosophie de la technicité

Les deux textes qui constituent notre corpus permettent de comprendre plus précisément la spécificité de la philosophie de la technique de Simondon : il s’agit d’un champ dynamique, ouvert aux questionnements et à la réappropriation.

Ainsi, le *Mode d’existence des objets techniques* et la *Psychosociologie de la technicité* étudient de deux points de vue différents les rapports de l’homme à l’objet technique ; du point de vue objectif de la genèse de l’objet lui-même (MEOT), du point de vue objectal<sup>1</sup> de la vie des objets dans leur cadre psychosocial (PST).

Précisons : la nécessité de réconciliation entre la culture et la technique est le postulat de départ du MEOT. Ainsi, les premières phrases de l’introduction posent la raison d’être de tout un livre qui tend à légitimer, sur le plan ontologique, les objets techniques<sup>2</sup> : « Cette étude est ani-

---

1. La distinction objectif/objectal est instituée par J-Y.Chateau dans l’introduction du recueil *Sur la technique* [93] : tandis que le MEOT traite de l’essence des objets techniques, de leur objectivité, la PST traite de la prise d’indépendance de ces mêmes objets une fois qu’ils sont projetés dans le monde social, sur un mode différent quoique dépendant du premier, le mode « objectal ». Lorsque l’objet a fini d’être produit techniquement, qu’il a atteint le comble de son objectivité, il devient détachable de son producteur et est investi de significations psychosociales.

2. La philosophie du MEOT s’inscrit dans la continuité de la thèse primaire de Simondon, qui développe une ontologie génétique du vivant en général. Voir à ce sujet J-H. Barthélémy, *Penser l’individuation. Simondon et la philosophie de la nature*. [20] Nous tentons de partir d’une problématisation de la recherche technologique en tant qu’elle essaie de réconcilier technique et culture, sans nous placer au départ sur le plan de l’ontologie de la nature.

mée par l’intention de susciter une prise de conscience du sens des objets techniques. » Si une telle prise de conscience est nécessaire, c’est qu’il y a, on l’a dit, un divorce effectif : « La culture s’est constituée en système de défense contre les techniques<sup>3</sup> ». Un dialogue se crée alors entre la pensée réflexive du MEOT et celle de la PST. La PST prend comme point de départ la pensée ontologique sans la développer, et vient approfondir l’aspect psychosocial des relations entre l’homme et l’objet technique. Dans les deux ouvrages, la réflexion psychosociale et la réflexion philosophique se font écho.

On pourrait nous objecter que le rapprochement entre le MEOT et la PST est arbitraire, et qu’il aurait été tout aussi efficient de rapprocher la thèse secondaire d’un autre cours du même recueil ; nous répondrons à cela que la PST, contrairement aux autres cours, déploie une pensée systématique très complète<sup>4</sup>. Son architecture, bien

3. MEOT, Introduction, p.1.

4. Selon J-Y. Chateau dans l’introduction à la PST, cette dernière fournit le point de vue manquant à l’ontologie « en trois volets » de Simondon, qui est destinée à « replacer l’individu dans l’être selon les trois niveaux physique, vital et psychosocial » (ILFI, p.32). Bien que la PST ait en principe un statut mineur vis-à-vis des deux thèses (il s’agit d’un cours, donc d’un format oral, moins précis, et plus court), cette dernière prolonge effectivement les deux premiers points de vue autour de concepts principaux (objectalité, objet ouvert/fermé) qui s’ouvrent sur des considérations plus larges, notamment la notion de réseau.

L’importance de la PST est d’ailleurs souligné par les éditeurs du recueil. La PST en est le cours d’ouverture. J-Y. Chateau y ajoute une démarcation typographique. Il cite dans son introduction différents textes du recueil et, malgré le fait que la plupart (hormis les fragments, notes et entretiens) soient aussi des cours, seule la PST est en italiques (donc considérée comme un ouvrage à part entière), tandis que les autres textes sont entre guillemets. Ces choix éditoriaux sont intéressants à relever si on considère ce qu’ils pointent :

que moins développée du fait du format, est tout aussi ambitieuse que celle du MEOT. La portée théorique du cours sur la psychosociologie vient prolonger la portée théorique des thèses du MEOT ; mieux, elle questionne et module certaines analyses de ce dernier. Cette remise en question, trois ans après la publication du MEOT, est fondamentale. La cohérence du corpus sélectionné tient aussi en ce que les deux textes ne sont pas seulement complémentaires ; leur mise en regard est problématique. La PST questionne les acquis du MEOT sur deux points qui sont liés l’un à l’autre : celui de la technicité d’abord, celui du statut du représentant de la technicité ensuite.

Une double problématique émerge de la confrontation de ces points de vue. Les deux questions ouvertes que nous allons mettre à jour font de notre corpus un ensemble cohérent, dont on peut reprendre les interrogations comme point de départ pour une nouvelle réflexion. La philosophie de la technique de Simondon est un « programme de travail »<sup>5</sup>, en ce sens qu’il y a certains problèmes que l’auteur pose plus qu’il ne les résout.

on a considéré que la PST, dans ces nouveaux textes, avait un statut majeur. Les autres textes du recueil ont effectivement un contenu plus local que les thèmes développés par la PST, et ne dialoguent pas frontalement avec les deux thèses principales. « L’effet de halo en matière technique », par exemple, vient compléter la PST sur un point précis, celui de la communauté quasi religieuse et asymétrique (il y a « quelque chose de religieux » dans le halo) des utilisateurs qui se forme autour de l’objet technique. Cela vient compléter le cadre théorique général institué par la PST.

5. Selon l’expression de J-Y. Chateau dans « Technophobie et optimisme technologique moderne et contemporain » [35]. Analysant la structure du MEOT, J-Y. Chateau considère que la progression du local au global ménagée par les trois parties du MEOT ne propose pas des solutions de plus en plus solides, mais *pose de plus en plus solidement le problème de la technique.*

### I.A.1 Le fonctionnalisme simondonien

L’essence des objets techniques réside pour Simondon dans leur technicité, et leur technicité est fonctionnement.

#### Intérêt du fonctionnalisme simondonien par rapport aux conceptions hylémorphiques et utilitaristes

La compréhension de l’objet technique comme fonctionnement est une thèse radicale qui confère aux analyses de Simondon un statut particulier dans le champ de la philosophie de la technique.

Simondon se positionne explicitement par rapport à deux conceptions classiques de la technique. Il s’agit d’une part de la tradition de pensée initiée par la philosophie aristotélicienne, qui emploie ce que Simondon nomme le “schème hylémorphique” pour comprendre les objets techniques, et de l’autre de la conception d’une technique utilitaire et arraisonnable défendue par Heidegger. Ces deux courants de pensée tendent à définir la technique au prisme des objets qu’elle produit. Pour la philosophie aristotélicienne, l’objet technique est ce qui relève de la tekhnê. Dans la *Physique*, Aristote fait intervenir la distinction entre tekhnê et phusis pour consacrer la supériorité ontologique du fait naturel. Tandis que les êtres naturels contiennent en eux le principe de mouvement et de repos « immédiatement et à titre essentiel<sup>6</sup> », les produits de l’art, les choses artificielles existent essentiellement sur un mode contingent :

6. Aristote, *Physique*, Livre II. [2]

« [elles n’ont pas] en elles-mêmes le principe de leur production ; les unes l’ont en d’autres choses et hors d’elles, tels une maison et tout objet fait de main d’homme ; les autres l’ont bien en elles-mêmes, mais ce n’est pas par essence, (savoir) toutes celles qui peuvent être par accident causes d’elles-mêmes. »<sup>7</sup>

C’est le sens du schème hylémorphique : l’essence de l’objet technique réside en ce qu’il est une matière à laquelle un agent humain a imposé une forme de l’extérieur<sup>8</sup>. Les objets techniques ne sont pas caractérisés pour eux-mêmes (ils appartiennent à la classe plus large des artefacts, des choses artificielles) et, s’ils le sont, ce n’est que « par accident ». Leur essence est définie uniquement sur le mode contingent<sup>9</sup>. Le second schème vis-à-vis duquel Simondon se démarque est le schème utilitariste ; c’est ce qu’il nomme l’usage « ustensile » des objets techniques. Ici, Simondon vise directement Heidegger. Dans *La question de la technique*, ce dernier affirme que “la vérité du monde de la technique se trouve dans une puissance qui n’est pas une réalité technique elle-même.”<sup>10</sup> La puis-

7. *Ibid.*

8. Ainsi, les artefacts sont « [ce] dont la forme est dans l’esprit de l’artiste. » (*Métaphysique Z 7, 1032 b*). [3]

9. On trouve la prégnation d’un tel schème hylémorphique chez Kant : « Lorsqu’en faisant des fouilles dans un marécage, comme c’est arrivé parfois, on trouve un morceau de bois taillé, on dira qu’il s’agit non d’un produit de la nature, mais de l’art ; sa cause efficiente s’est accompagnée de la pensée d’un but auquel l’objet doit sa forme. » (*Critique de la faculté de juger*, paragraphe 43) ou chez Marx : « [...] ce qui distingue dès l’abord le plus mauvais architecte de l’abeille la plus experte, c’est qu’il a construit la cellule dans sa tête avant de la construire dans la ruche. » (*Le Capital*, Livre I, troisième section, chapitre VII, paragraphe I). [75]

10. J.Y. Chateau, “Technophobie et optimisme technologique mo-

sante arraisonnable de la technique est indépendante de toute volonté et de tout pouvoir humain. Elle est ”un dispositif (*Einrichtung*), en latin un *instrumentum*”<sup>11</sup>. L’unité des objets techniques tient en ce qu’ils participent d’une essence commune, le *Gestell*, que l’on peut traduire par le ”Dispositif”<sup>12</sup>. Heidegger définit la technique par son absence d’essence : ”L’essence de la technique n’est absolument rien de technique.”<sup>13</sup> Elle est tout au plus un symptôme, un indice du dévoilement de la métaphysique.

Le point commun de la tradition hylémorphique et de la tradition utilitaire réside en ce qu’elles définissent toutes deux l’essence de la technique en négatif. Cette dernière est contingente, secondaire : elle n’existe pas pour elle-même.

Contrairement à ces deux études, Simondon tente d’assigner à l’objet technique une essence qui lui soit propre, et qui lui soit irréductible ; c’est une d’une démarche de caractérisation positive. Notre hypothèse est que le positionnement simondonien, par son ouverture et sa positivité, permet la réappropriation. Nous estimons que la pensée de l’auteur est adaptée pour penser la technicité qui nous est contemporaine, alors que les deux traditions précédentes évoluent dans un cadre conceptuel *a priori* trop limité. Nous chercherons à établir en quoi le fonctionnalisme de Simondon se trouve radicalisé et validé *a posteriori* par sa confrontation à un objet technique

---

dernes et contemporains” [35].

11. Heidegger, *La question de la technique* [62].

12. Il s’agit d’une proposition de traduction différente de ”l’arraisonnement”, qui renvoie mieux à la réalité du terme allemand. Voir Janicaud, *La puissance du rationnel*, p.271[66] (cité par J.Y. Chateau, [35] )

13. Heidegger, *op. cit.* [62].

contemporain, alors que les deux autres théories semblent insuffisantes pour le comprendre.

### Une pensée de la genèse des objets

La technologie<sup>14</sup> déployée dans la première partie du MEOT s’inscrit dans la continuité de la thèse primaire. Elle postule le lourd appareillage conceptuel de *L’individuation à la lumière des notions de formes et d’information*. Ainsi, l’objet technique est « ce dont il y a genèse »<sup>15</sup>. Il ne s’agit pas ici d’une genèse d’ordre général, mais bien d’un processus génétique particulier : la concrétisation, qui est passage du mode « abstrait », analytique, au

14. La technologie est entendue par Simondon comme discours étudiant les objets techniques.

15. L’objet technique est « ce dont il y a genèse » (MEOT, p.20) « selon des modalités qui distinguent la genèse de l’objet technique de celles des autres types d’objets : objet esthétique, être vivant » (MEOT, p.20, note 1). Toute la philosophie simondonienne consiste en l’établissement d’une ontologie générale dont le fondement est une pensée génétique. Voir à ce propos J.-H. Barthélémy, *Simondon ou l’encyclopédisme génétique* : l’encyclopédisme de Simondon consiste dans l’unification des savoirs dans le processus de genèse, dont procède toute réalité. Le système simondonien est pour Barthélémy une « nouvelle ontologie génétique de l’individuation ». Chaque mode d’existence correspond à une genèse particulière. Replacer le MEOT dans la lignée de l’ontologie générale est donc primordial ; c’est pourquoi nous nous inscrivons dans le sillage des études de J.-H. Barthélémy sur le sujet, sans relever pour notre part les problématiques propres à l’ontologie et à la cohérence du système général de Simondon (conditions et limites d’une transposition du vital vers le technique, statut de l’ontologie comme « philosophie inachevée » sont par exemple deux problématiques d’ordre plus global traitées par Barthélémy dans l’ouvrage déjà cité). Nous nous limitons à l’étude de sa philosophie de la technique – aux enjeux qu’elle soulève et à la méthode qu’elle déploie.

mode « concret ». L’objet concrétisé est la solution d’un problème : il est cette entité qui a réussi à surmonter tous les obstacles et incompatibilités inhérents à sa réalisation. De fait, l’objet technique ne peut être compris du point de vue de son état présent, car il ne se différencierait pas fondamentalement d’une chose. Ce qui permet de dire qu’un objet est réellement objet technique, c’est l’évolution de son fonctionnement. L’objet technique n’est pas une chose mais l’aboutissement temporaire d’une lignée de fonctionnement, qui constitue sa genèse. C’est pourquoi il est doté d’un « mode d’existence »<sup>16</sup> propre. Il existe dans le monde d’une façon particulière et différenciée des êtres humains et de la vie biologique, puisqu’il dispose de ce qui, du point de vue général de l’ontologie simondonienne, lui donne de plein droit une dignité ontologique : un processus génétique propre. Examiner la genèse des objets techniques, c’est découvrir leur mode d’existence particulier, et, ce faisant, ce qui constitue leur

---

16. La notion de « mode d’existence » provient d’un philosophe aujourd’hui peu connu, Souriau, qui a rédigé en 1943 un livre intitulé *Les différents modes d’existence* (réédité chez PUF en 2009) [100]. L’ouvrage défend la thèse d’un pluralisme existentiel : plusieurs manières d’exister (voire de « sur-exister », ou de « sous-exister ») sont possibles. Simondon reprend cette idée. Ainsi, la troisième partie du MEOT décrit la réalité comme découlant d’un mode d’existence unique, « le mode magique », qui se dédouble en « mode d’être religieux », tenant de la subjectivité, et « mode d’être technique », tenant de l’objectivité (MEOT, p.160). Il y a donc plusieurs modes d’existence (dont découlent plusieurs pensées : la pensée esthétique et philosophique notamment), dont chacun doit être respecté pour lui-même. Par ailleurs, on voit aussi que Simondon réduit l’opposition classique de l’objet et du sujet à des effets tardifs d’une histoire, elle première, des modes d’existence (voir MEOT, p.168 et l’article de B. Latour sur la notion de mode d’existence, « Prendre le pli des techniques »).

technicité<sup>17</sup>. Pour Simondon, est objet technique ce qui *fonctionne*.

C’est pourquoi la technicité, comprise en tant que fonctionnement, est un concept large qui n’épuise pas toute sa réalité dans son objectivation. Comprendre : la technique n’est pas seulement l’objet technique, bien que ce dernier soit un prisme d’analyse privilégié<sup>18</sup>.

Ainsi, la technicité s’étudie pour Simondon sous deux aspects principaux. D’abord via l’étude de la lignée « phylogénétique »<sup>19</sup> (chap. I, « Genèse de l’objet technique : le processus de concrétisation). Est objet tech-

---

17. C’est ce à quoi s’engage la première partie du MEOT. Cette démarche a des résonances éthiques et politiques dès le départ. Elle répond concrètement, par la méthode, à l’appel initial de l’introduction. En effet, Il y a un développement concret, propre à l’objet technique : si l’objet abstrait est « la traduction physique d’un système intellectuel », l’objet concret devient peu à peu indépendant de l’acte d’invention, et se réalise dans la synergie progressive des fonctions qui le composent, et auquel l’humain s’adapte.

18. Comme le note J-P. Sérís dans *La technique* [88]. Nous problématiserons plus avant les liens entre Simondon et l’objet technique dans la partie II.A.1.

19. La phylogénétique est, en biologie, la science de la genèse de l’espèce (phylogenèse). Elle s’oppose à l’ontogenèse, qui est genèse de l’individu. Simondon exporte cette notion dans le champ de la philosophie de la technique : la « lignée phylogénétique » d’un objet technique est tout le développement qui a conduit jusqu’à lui, la genèse de son « type » technique : elle est la « dimension temporelle d’évolution » de l’objet technique (MEOT, p.66). Il y a pourtant des différences propres à l’objet technique que Simondon souligne fortement : la lignée phylogénétique de l’objet technique « n’est pas identique à l’évolution biologique » (MEOT, p.66). La différence réside en ce que l’évolution technique est suit des lignes moins « continues » (MEOT, p.66) que l’évolution biologique, car, contrairement à cette dernière, ses éléments sont directement *détachables*. « Dans le domaine de la vie, l’organe n’est pas détachable de l’espèce ; dans le

nique non la lampe que nous avons devant nous, mais « l’unité de devenir » (p.20) qui a évolué de bout en bout en lampe de plus en plus cohérente et fonctionnelle dans un mouvement de « synergie fonctionnelle » ; l’objet est évolutif ; il nécessite une étude diachronique. L’objet technique n’est pas seulement cet objet sous nos yeux ; il est le terme, l’accomplissement (provisoire, si le système n’est pas arrivé à saturation) d’une évolution propre à l’objet, qui est de plus en plus cohérent avec lui-même et avec le milieu qui l’entoure : c’est le processus de concrétisation technique.

La technicité s’étudie ensuite via un examen scalaire<sup>20</sup> : la concrétisation des objets techniques est différente selon le niveau de technicité étudié. Trois types de technicité coexistent : l’élément, l’individu, et l’ensemble. Ces derniers sont d’abord des niveaux d’analyse classiques : un ensemble comprend et coordonne plusieurs individus techniques, qui font eux-mêmes fonctionner ensemble plusieurs éléments ; tandis que « les objets techniques infra-individuels peuvent être nommés éléments techniques » (MEOT, p.65), l’individu (dont le représentant est la machine naissant à l’âge industriel) est « ce qui porte ses outils et les dirige » (MEOT, p.78-80) ; enfin l’ensemble est celui « qui comprend tous les sous-ensembles » (MEOT, p.63). A ce jeu d’échelle s’ajoute des âges tendanciels ; ainsi, l’objet technique gagne graduellement en technicité selon un processus qui joue à

---

domaine technique, l’élément, précisément parce qu’il est fabriqué, est détachable de l’ensemble qui l’a produit ; là est la différence entre l’engendré et le produit. » (MEOT, p.67)

20. Nous reprenons ici les analyses éclairantes de J.-H. Barthélémy dans son Glossaire Simondon [19], et dans son article « Sur l’architectonique de Du mode d’existence des objets techniques »[17].

trois niveaux :

- Le premier niveau est celui de la concrétisation, qui se joue au niveau de l'élément. Un élément qui se concrétise est un élément qui passe d'une fonction à plusieurs fonctions dans l'objet technique (Simondon emploie l'exemple de l'ailette à refroidissement dans un moteur, sur lequel nous reviendrons) : c'est la « plurifonctionnalité ». Les éléments sont de plus en plus dépendants les uns des autres. L'objet technique, via la concrétisation des éléments, acquiert une plus grande « résonance interne ».
- Le second niveau est celui de l'individualisation, qui se joue au niveau de l'individu, dont le représentant-type est la machine à l'âge industriel. La cohérence de l'objet technique se renforce non par « résonance interne » mais par « résonance externe » : il développe une relation de « causalité réciproque » avec le milieu associé dans lequel il évolue<sup>21</sup>.
- Le troisième niveau est celui de la naturalisation. Ici, l'individu technique continue de s'individua-

---

21. Le milieu associé est « ce par quoi l'être technique se conditionne lui-même dans son fonctionnement » (MEOT, p.56-57). Le milieu associé est, par le biais de l'invention humaine, la concrétisation « d'un milieu techno-géographique ». Il est « la mise en relation de deux milieux l'un et l'autre en évolution » (p.53) : c'est un « milieu mixte », « technique et géographique » (p.54). Par exemple, dans une locomotive, « Le moteur de traction ne transforme pas seulement l'énergie électrique en énergie mécanique ; il l'applique à un monde géographique varié, se traduisant techniquement par le profil de la voie la résistance variable du vent, la résistance de la neige que l'avant de la locomotive repousse et écarte. Le moteur de traction rejette dans la ligne qui l'alimente *une réaction qui traduit cette structure géographique et météorologique du monde* » (p.53).

liser à l’intérieur d’ensembles techniques qui sont devenus eux-mêmes le « milieu associé » de chaque individu technique.

La notion « d’âge tendancier » utilisée par J-H. Barthélémy s’explique par le fait que l’ensemble, et la naturalisation qui lui est corrélatrice, est le lieu prépondérant de la technicité dans le MEOT. La normativité technique est située historiquement à l’âge des ensembles, qui est aussi celui de l’information : « aujourd’hui, la technicité tend à résider dans les ensembles »<sup>22</sup>. Cela a une conséquence directe quant à la résolution du problème posé par l’introduction du livre : la technicité « (...) peut alors devenir un fondement de la culture à laquelle elle apportera un pouvoir d’unité et de stabilité, en la rendant adéquate à la réalité qu’elle exprime et qu’elle règle »<sup>23</sup>.

La technicité peut donc être considérée sous plusieurs angles.

1. Elle est avant tout, on l’a vu, fonctionnement. Toutefois, elle est rarement définie de façon générale dans le MEOT, si ce n’est page 71, où Simondon la décrit positivement : « La technicité de l’objet est le degré de concrétisation de l’objet. » Toujours à la même page, Simondon précise :

*« La technicité de l’objet est donc plus qu’une qualité d’usage ; elle est ce qui, en lui, s’ajoute à une première détermination donnée par un rapport de forme et de matière. »*

---

22. MEOT, p.16

23. Voir aussi MEOT, p.126.

La technicité est donc, de ce point de vue, ce qui se surajoute à l’objet (assemblage de la matière et d’une forme). C’est un fonctionnement de plus en plus cohérent (que Simondon nomme concrétisation) découlant à la fois du geste d’invention humain initial et de la synergie des composants entre eux.

2. Autrement, la technicité est *qualifiée* (elle intervient pour caractériser les différents niveaux d’échelle : l’élément, l’ensemble) et pensée dans l’optique d’âges *tendanciels* : ainsi, la technicité « tend à résider dans les ensembles » pour le MEOT<sup>24</sup>.

Au-delà de son aspect fonctionnel, la technicité est donc intrinsèquement liée à sa situation dans les âges tendanciels de la technique : pour le MEOT, elle est liée aux ensembles. La PST vient mettre en tension cette définition : tout en prenant comme point de départ le cadre d’analyse de la technologie simondonienne (genèse des objets, échelles de niveaux, technicité qualifiée), permettant ainsi un cadre commun pour la comparaison, elle vient modifier la définition de la technicité en attribuant sa prépondérance à un autre niveau d’échelle.

## I.A.2 La question de la localisation de la technicité

De fait, l’ouvrage se place d’un point de vue différent, celui de la méthode psychosociologique. La PST ne s’in-

---

24. MEOT, p.16

téresse pas à l'essence de la technicité en elle-même mais caractérise « un ensemble de représentations et d'attitudes concernant la technique »<sup>25</sup>. Elle est ce qui étudie l'objet une fois « libéré » de sa production objective, et projeté dans l'espace social – l'objet n'est plus perçu comme objet technique, mais comme objet d'usage. Positivement, la technique du point de vue psychosocial se définit comme « une activité de l'homme en groupe, et une activité qui suppose et provoque des représentations, des sentiments, des mouvements volontaires. »

Pour ce faire, la psychosociologie use d'un prisme d'analyse particulier, dont le but est de prendre en compte à la fois les représentations individuelles (psychologiques) et collectives (sociologiques). Le postulat de la méthode psychosociologique est que pour comprendre le lien de l'homme à sa réalité technique, on ne peut se satisfaire ni d'une psychologie pure ni d'une sociologie pure<sup>26</sup>, car ces deux dernières découlent de la présupposition d'une existence substantielle de l'individu, qui serait séparable de son existence sociale. Partant du principe, au contraire, qu'il est impossible de dissocier individu et société, Simondon entend mener une étude des relations transindividuelles<sup>27</sup> qui rende compte des

25. J-Y. Chateau, Introduction à la PST.

26. ILFI, p.315 et 534.

27. Simondon définit ainsi le transindividuel dans l'ILFI : « L'individuation psychique et collective sont réciproques l'une par rapport à l'autre ; elles permettent de définir une catégorie du transindividuel, qui tend à rendre compte de l'unité systématique de l'individuation intérieure (psychique) et de l'individuation extérieure (collective). Le monde psychosocial du transindividuel n'est ni le social brut ni l'interindividuel ; il suppose une véritable opération d'individuation à partir d'une réalité pré-individuelle » (p.29). Ainsi, c'est

liens intrinsèques entre l’individu et la collectivité<sup>28</sup>. La méthode psychosociologique est, comme la méthode de la technologie génétique, directement déduite de la réalité de son objet. Elle n’est pas un choix arbitraire ; de même que l’objet technique appelle naturellement des schémas pour la compréhension de son fonctionnement<sup>29</sup>, qui n’est pas directement relié au langage, la réalité psychosociale analyse de façon indifférenciée l’individuel et du collectif, car les relations à la technique sont « des phénomènes qui sont simultanément psychologiques et sociaux »<sup>30</sup>

Ce que Simondon nomme psychosociologie de la technicité est donc une méthode visant à rendre compte des représentations transindividuelles à l’égard des techniques. Ces dernières sont diverses et entourent la technicité d’un « halo » psychosocial<sup>31</sup>.

---

« l’être comme relation qui est premier et qui doit être pris comme principe ; l’humain est social, psychosocial, psychique, somatique, sans qu’aucun de ces aspects puisse être pris comme fondamental alors que les autres seraient jugés accessoires » (p.297). Cela fait écho à toute l’ontologie déployée par Simondon : il faut prendre un point de vue « psychosociologique » à l’égard de l’humain parce « sa nature est génétique transindividuelle » (J-Y. Chateau). Ainsi, « Le psychosocial est du transindividuel » (p.303).

28. En cela, Simondon correspond à l’orientation de la psychosociologie française, énoncée clairement par S. Moscovici quelques décennies après le cours sur la PST : la méthode psychosociologique étudie, d’après S. Moscovici, « le conflit entre l’individu et la société » (p.7), les phénomènes d’idéologie, et de communication. Toujours selon S. Moscovici, la psychosociologie a pour originalité de « mettre en question la séparation de l’individuel du collectif, de contester le partage entre psychique et social dans les domaines essentiels de la vie humaine. » [80]

29. Partie I.B sur la méthode simondonienne

30. Voir MOSCOVICI S., *Psychosociologie* [80].

31. Voir « L’effet de halo en matière technique : vers une stratégie

Une telle méthode psychosociale permet de préciser le problème dont les termes généraux ont été posés en introduction du MEOT : celui du divorce entre technique et culture. Tandis que le MEOT énonçait un état de fait, la PST fournit un cadre conceptuel pour le comprendre.

Le divorce entre technique et culture se traduit par des phénomènes d’aliénation de la culture à l’objet technique<sup>32</sup> via des représentations erronées. La culture perd la conscience de la technicité de l’objet en le transformant en objet d’usage, et en l’investissant d’un sens déterminé par des intérêts principalement économiques : l’objet technique est « ostracisé ». Simondon dresse des couples d’opposition pour appuyer cette rupture : il oppose la cryptotechnicité (objets dont la technicité est dissimulée) de la plupart des objets techniques à la phanérotechnicité (objets dont la technicité est apparente, exhibée) des autres – les premiers étant acceptés au détriment des seconds<sup>33</sup>. De même, la surhistoricité des objets

---

de la publicité (1960) », in *Sur la technique* [94].

32. Et inversement. Les lignes qui suivent visent à donner un cadre conceptuel général des enjeux de la PST, sans les développer en détail. Elles servent à introduire la problématique de la localisation de la technicité. La question de l’aliénation sera développée plus longuement en III.A

33. Les objets pharénotechniques peuvent être eux aussi acceptés, mais cela n’aide pas à la connaissance de l’objet technique ; lorsque l’on accepte la vue de la technicité, il s’agit plus d’une perspective « technophani[que] » que d’une perspective rationnelle (par exemple, montrer la puissance d’un moteur). Cette technophanie est la voie par laquelle l’objet technique ostracisé peut reprendre, tant bien que mal, une place dans la culture qui le rejette ; elle est à la fois irrationnelle et problématique (l’objet rentre dans la culture par « ritualisation ») mais peut être également le biais d’une ré-introduction de l’objet technique dans la culture (via les êtres « néoténiques », les amateurs de technologies, qui peuvent s’intéresser réellement aux objets techniques à la suite de cette porte d’entrée).

s’ajoute à leur historicité première (celle de leur conception) au moment où l’objet est acheté. L’acte d’achat est pour Simondon synonyme d’aliénation de l’objet, qui perd la reconnaissance de sa dimension technique.

Le couple conceptuel objet ouvert/fermé va en ce sens<sup>34</sup> : Simondon distingue l’objet ouvert, dont la technicité est apparente, et l’objet fermé, qui est une boîte noire empêchant de « lire en lui l’opération constructrice ». Tandis que l’objet ouvert permet la connaissance des schèmes techniques et autorise la reprise, l’objet fermé stérilise tout rapport à l’objet ; il clôture une bonne fois pour toutes sa technicité et favorise l’inculture à son égard. L’objet fermé est une des causes du divorce entre technique et culture.

C’est la théorie de l’objet industriel « ouvert » qui engage un changement dans la localisation de la technicité par rapport au MEOT. Pour que l’objet industriel puisse être toujours considéré comme ouvert, il doit l’être en tant que garant de pièces indépendantes les unes des autres, porteuses d’une technicité perfectible. C’est « la pièce détachée », l’élément, qui devient fondamental dans la vision de la technicité – car l’individu au niveau industriel est condamné à la clôture. Par ailleurs, l’importance de l’élément engendre une seconde conséquence : celle de la relation constante entre producteur et l’utilisateur via la *mise en réseau* de « dépositaires possédant les pièces nécessaires ». La réhabilitation de l’élément engendre l’importance capitale de la réticulation : « Il ne peut y avoir déploiement d’une réelle ouverture des objets techniques sans création d’un réseau de technicité. »<sup>35</sup>.

Les catégories d’analyse psychosociales, différentes de

34. PST, chapitre II.

35. PST, p.69

celles de la technologie du MEOT, viennent affiner et enrichir la vision de la technicité de Simondon. Si l'on se place du point de vue de l'ouverture des objets, « c'est ici l'élément *et non plus l'ensemble* qui est dépositaire du pouvoir d'ouverture »<sup>36</sup>. Autrement dit, c'est dans l'élément, plus protégé de la virtualisation à l'ère industrielle que ne l'est l'individu, qu'il faut chercher la potentialité de non-aliénation de la culture à sa réalité technique. Pour que l'utilisateur ne soit pas complètement coupé de la réalité technique à l'ère industrielle des objets fermés et pollués de « surhistoricité »<sup>37</sup>, il fallait résoudre le problème par une question d'échelle. C'est ce à quoi s'emploie Simondon : l'objet industriel est fermé en tant qu'objet d'usage ; « l'automobile ou le récepteur de télévision sont appelés à se fermer au niveau du véhicule ou du meuble » (p.70), certes. Mais la fermeture des objets au niveau de « la dimension corporelle de l'homme » (le « niveau pratique d'utilisation »), c'est-à-dire celui des individus techniques, correspond à leur ouverture à deux autres niveaux :

— L'échelle « *microtechnique* des pièces détachées », c'est-à-dire le niveau de l'élément

36. PST, p.68.

37. Pour Simondon, la surhistoricité est définie comme « la zone extérieure » de l'objet technique, au niveau de « ce qui, dans l'objet technique, est l'équivalent du vêtement pour l'être humain » (PST, p.58). L'historicité, par opposition, est la réelle date d'une invention, et ses évolutions techniques dites « majeures ». La généralisation de la surhistoricité a alors des conséquences néfastes sur la technicité inhérente aux objets techniques : « L'existence généralisée de la surhistoricité crée chez les constructeurs la nécessité de devenir producteurs de surhistoricité, en créant assez fréquemment des modèles nouveaux, ce qui revient à fractionner volontairement les réformes de structures correspondant à un réel progrès de technicité, parfois même à les différer. » PST, *ibid.*

— L'échelle « *macrotechnique* des réseaux de distribution et d'échange »

Simondon opère donc un raffinement d'analyse par rapport à la prépondérance affichée des ensembles dans le MEOT<sup>38</sup> : plus que les ensembles, sont garants de la technicité le couple élément / réseau. Il y a « dédoublement des ordres de grandeur des supports de technicité. » Simondon parvient alors à sauver l'ordre industriel d'un point de vue psychosocial : « Lorsqu'on recherche l'unité de la culture, il ne convient donc pas de déplorer que la vie industrielle ne soit pas à l'échelle humaine. »<sup>39</sup>, car cette dernière « libère la réalité technique d'un asservissement à l'ordre humain de grandeur. »<sup>40</sup> Ainsi, l'unité de la culture est sauvegardée via une modification des vues du MEOT : on passe de la technicité de l'ensemble à celle du couple élément et réseau.

38. Il s'agit bien d'un raffinement d'analyse, et non d'un renversement complet ou d'une contradiction interne à la pensée de l'auteur. Simondon caractérise dans le MEOT les éléments comme ce à partir de quoi la technicité essaime. Par ailleurs, la distinction réseau/ensemble reste ambiguë : même s'il s'agit *a priori* de deux réalités différentes (L. Duhem caractérise d'ailleurs le réseau comme « quatrième stade de technicité » [46] ), on peut avancer que le terme d'ensemble dans le MEOT préfigurait, de façon moins précise, ce que la PST (et le cours de 1968-1969 "L'invention et le développement des techniques" à sa suite [89] ) caractérise comme une réalité technique réticulée. Le MEOT et la PST entretiendraient alors un rapport de continuité plus que d'opposition dialectique. Cette thèse « continuiste » est défendue notamment par J-Y. Chateau (entretien, 6 juin 2017).

39. PST, p.71

40. PST, *ibid.*

### I.A.3 La question des représentants de la technicité

La technicité simondonienne est, dans son sens le plus général, fonctionnement. Cette généralité lui permet d’être située : Simondon établit qu’elle réside, à l’ère industrielle, dans les éléments réticulés. Une troisième problématique caractérisant la technicité, soulevée par la mise en regard des deux textes, est celle du statut de ses représentants.

Dans le MEOT, Simondon annonce que les deux types de pensée paradigmatiques du rapport de la culture à la technique ne sont « pas cohérents ». Ce manque de cohérence est « en partie responsable des contradictions que renferme la culture actuelle dans la mesure où elle juge et se représente l’objet technique en rapport avec l’homme ». Nous sommes donc face à l’opposition de deux paradigmes dont l’impossible réconciliation est en partie responsable du divorce entre technique et culture. Par ailleurs, cette disjonction représentative a son corollaire dans le monde des objets lui-même : « La disjonction entre la culture et la technique a sa condition dans la disjonction qui existe à l’intérieur du monde des techniques lui-même. »<sup>41</sup> Ce passage fait écho à dichotomie entre objets techniques ouverts et objets techniques fermés analysée dans la seconde partie de la PST<sup>42</sup> : les objets fermés et leur cryptotechnicité sont également responsables d’une forme d’aliénation de la culture vis-à-vis

41. MEOT, p.87

42. Voir PST, « Objet technique ouvert et objet technique fermé », p. 60 et suivantes.

de la technique réelle. On a donc une dichotomie de représentations qui correspond à une dichotomie de fonctionnement des objets ; cette dichotomie sous-tend une aliénation qui est la problématique principale de tout le corpus Simondonien. Il ne s’agit pas seulement d’une aliénation économique au sens marxiste du terme, mais bien d’une aliénation plus profonde, qu’il faudrait également désamorcer pour détruire celle économique ; celle des hommes à leurs objets techniques, via la représentation qu’ils s’en font. Or la représentation est conditionnée par la configuration technique des objets. Comment faire pour résoudre cette impasse ? Il faut des représentants adéquats de la technicité, des porte-paroles. Simondon recherche un « juste milieu » mais plusieurs points sont laissés problématiques. Les analyses du MEOT présentent l’artisan comme véhiculant un savoir fermé, exclusif, assimilé à la figure de l’enfant ; l’ingénieur est adulte, porteur d’un savoir souple, non rigide, ouvert à l’apprentissage. A la « rigidité » du premier s’oppose l’« encyclopédisme » du second<sup>43</sup>. Le MEOT déploie donc une hiérarchie très claire<sup>44</sup> entre le rapport à la technique de l’ingénieur et le rapport à la technique de l’artisan. Le critère de classification est celui de la passation du savoir : l’ingénieur est susceptible de transmettre ce dernier, et peut donc réduire le divorce entre culture

43. Simondon relève cette opposition comme symptomatique d’une époque, mais ayant existé de tout temps : s’ensuit un long développement historique pour comprendre le divorce entre la technicité et la culture à travers les âges.

44. Bien qu’il ne remette pas en question la « quantité d’information » véhiculée par le savoir artisanal : « la primitivité ne saurait être confondue avec la bêtise, pas plus que la conceptualisation avec la science. », MEOT, p.90.

et technique<sup>45</sup>, tandis que l’artisan, tout enfermé dans sa « confrérie », ne peut faire de même. Le savoir de l’artisan est « rigide » : il ne peut pas évoluer. Il est « initiatique et exclusif », ce qui le rend intrinsèquement fermé à la passation de savoir ; on le voit dans le glissement du terme « rigide » au terme « fermeture » pour désigner les techniques anciennes<sup>46</sup>. On a donc, dans le MEOT, une typologie qui oppose l’artisan et l’ingénieur.

Simondon laisse ouverte la question des représentants adéquats de la technicité ; le médiateur idéal entre technique et culture n’est pas l’ingénieur. Dans un souci de justice et d’équilibre, l’auteur explique que « la condition première d’incorporation des objets techniques à la culture serait que l’homme ne soit ni inférieur ni supérieur aux objets techniques »<sup>47</sup>. Il faudrait qu’il puisse « apprendre à les connaître en entretenant avec eux une relation d’égalité, de réciprocité d’échange » ; il faudrait « découvrir une voie moyenne », une « représentation qui incorporerait à la fois celle de l’artisan et celle de l’ingénieur »<sup>48</sup>.

Dans la PST, un retournement s’opère. Le statut de l’artisan est grandement revalorisé. Il diffère complètement de celui qui a été énoncé dans le MEOT : l’artisan fait partie, avec l’ingénieur, de ceux capable de produire des objets ouverts. On assiste à un renversement des catégories précédemment énoncées par Simondon. A la formation technique « rigide », se substituent des objets

45. L’ingénieur est celui qui véhicule l’encyclopédisme, le savoir des signes réfléchis et des symboles.

46. La technique artisanale correspond ainsi à un « régime fermé de vie », MEOT, p.90.

47. MEOT, p.88

48. MEOT, *ibid.*

« ajustables et réparables ». L'idée initiale est la même : dans les deux cas, l'artisan peint par le MEOT comme celui de la PST a une relation privilégiée avec la matière brute, il en a l'intuition. Pourtant, cette intuition de la matière conduit à une « fermeture » des techniques anciennes dans le MEOT, et à la production d'objets « ouverts », où la matière est vue comme « réformable et prolongeable », dans la PST. L'artisan est capable de produire des objets présentant la même caractéristique que ceux des ingénieurs de pointe : l'ouverture. L'opposition ne se situe plus alors entre l'ingénieur dominant son objet et l'artisan dominé par la matière ; du point de vue des objets, elle se situe entre l'objet commercial à la chaîne, fermé et inconnaissable, aliénant plus encore la culture et la technique, et l'objet ouvert, artisanal ou industriel avancé.

La « voie médiane » recherchée par le MEOT entre le statut de l'artisan et celui de l'ingénieur n'est pas encore résolue. Deux questions émergent de la comparaison du MEOT et de la PST. La première, interne au MEOT et posée ouvertement, est celle d'un « juste milieu » entre la représentation de l'ingénieur et celle de l'artisan. Il ne pourrait s'agir que d'un effet d'attente dans l'économie argumentative du MEOT, qui ménage une progression vers la troisième partie de l'œuvre, point culminant où la philosophie vient justifier la technologie développée dans les premiers chapitres. Si tel est le cas, nous choisissons tout de même d'emprunter une autre voie que celle de Simondon, et de chercher parmi les technicien·ne·s ce « juste milieu » que Simondon ne parvient pas à trouver parmi ceux qui pratiquent la technique. Cette problématique ouverte est redoublée par la confrontation des deux textes. L'évolution du statut de l'artisan dans la PST est

significative des hésitations de Simondon et de la question laissée ouverte à ce sujet - celle du statut de l'acteur dans le processus effectif de la concrétisation de l'objet technique, en tant que médiateur entre la technique et la culture. On a donc une problématique initiale, explicite, celle d'un « juste milieu » entre les représentations de l'artisan et celles de l'ingénieur, et une problématique implicite qui émerge de la confrontation des deux textes et vient recouper la première : quelles représentations adéquates de l'objet technique ouvert ? Quel est le statut de la représentation de l'artisan, partiellement réhabilité par rapport au MEOT ? En quoi la réhabilitation de l'élément avec la mise en réseau entraîne-t-elle une nouvelle vue de la technicité et, ce faisant, de nouveaux représentants ?

On le voit : la confrontation de la méthode psychosociale à la technologie du MEOT fait émerger des problématiques fécondes. Ce champ de réflexion pose, à partir d'une problématique commune (celle des rapports entre la technique et la culture), plusieurs éléments de réponse à la fois complémentaires et problématiques. Ces éléments de réponse sont apportés par des méthodes différentes (la méthode génétique et la méthode psychosociale) qui portent sur des objets différents (l'évolution des objets techniques en eux-mêmes et leur évolution dans le milieu social).

La philosophie de la technique chez Simondon est une problématisation progressive du concept de technicité. Ce dernier est de loin le geste le plus radical de Simondon : il établit l'essence des objets techniques dans leur fonctionnement, et non dans leur ustensilité ou dans leur aspect artefactuel.

La technicité est problématisée au niveau de sa localisation et de ses représentants. Nous avons ainsi soulevé

deux interrogations qui émergent de notre corpus :

1. Peut-on parler d'une technicité nouvelle, d'une « libération de l'élément » créant des objets ouverts autrement que dans le cadre du système industriel ? Plus généralement, comment prolonger la pensée de la technicité de Simondon via la notion de « libération de l'élément » ?
2. Quel serait le représentant de la technicité adéquat, la « voie médiane » entre l'artisan et l'ingénieur, propice à notre technicité contemporaine ?

Ainsi considérée comme programme de travail, la pensée de Simondon nous permettra de questionner un objet technique contemporain. L'auteur déploie la problématisation de la technicité en suivant une voie bien précise, qu'il nous faut explorer avant de reproduire son geste.

## **I.B La méthode simondonienne : au plus près de l'objet technique**

La méthode simondonienne est le second point sur lequel s'appuiera notre travail. Le dialogue entre Simondon et notre objet technique numérique doit s'instaurer selon des critères bien définis, que nous tirerons de la façon de procéder simondonienne.

La technologie génétique de Simondon opère un véritable retournement théorique. En refusant de considérer les objets comme des acquis, il les réhabilite en tant qu'éléments de culture, en tant qu'éléments connaissables – différents d'une œuvre d'art ou d'un livre, mais

comme porteurs d’un *sens* véritable et encore trop peu questionné. La technicité peut être soumise à investigation génétique. Simondon redonne une épaisseur aux objets en les faisant siège d’un processus de fonctionnement qui leur est propre, et qui mène jusqu’au point de perfection limite le geste inventif de l’homme qui y a été cristallisé.

### **I.B.1 Les enjeux épistémologiques : une méthode inductive**

La méthode simondonienne est un acte extra-langagier. Elle est un geste retranscrit : le philosophe des techniques s’échappe de son périmètre philosophique (discursif et rationnel) pour plonger dans une réalité de fonctionnements (celle des objets techniques). Cette sortie de la philosophie conditionne la production d’une philosophie des techniques informée et juste - comment parler de quelque chose dont on ne connaît pas la teneur réelle ? Simondon initie ce geste, en appelant à sa reproduction. C’est une tâche éthique qui incombe au philosophe : celle de plonger dans des domaines inconnus de la philosophie pour les mettre à disposition du lecteur, et ainsi réduire le fossé entre la technique et la culture.

Le fonctionnement argumentatif d’une telle méthode est l’induction. Simondon y fait directement référence à la fin du chapitre 1 de la première partie du MEOT : « le mode d’existence de l’objet technique concrétisé étant analogue à celui des objets naturels spontanément produits, on peut légitimement les considérer comme des

objets naturels, c’est-à-dire les soumettre à une *étude inductive*.<sup>49</sup> » La méthode de l’étude par induction des objets techniques est directement issue de celle utilisée pour les objets naturels, menée dans l’ILFI, qui a permis l’analyse du processus d’individuation. Simondon déploie un raisonnement analogique qui sert à justifier le transfert de la méthode inductive du vivant vers le technique. Les correspondances sous-jacentes de l’individuation (schèmes physiques) et de l’individualisation (schèmes techniques)<sup>50</sup> font qu’il est légitime d’appliquer la méthode de la thèse primaire à la thèse secondaire. Le MEOT reprend à la fois le postulat de l’ontologie génétique et sa méthode car « par la concrétisation technique, l’objet, primitivement artificiel, devient de plus en plus semblable à l’objet naturel. » L’analogie a ici un rôle bien précis : elle permet à Simondon d’étendre la méthode inductive au MEOT. Elle ne permet pas, en revanche, le glissement théorique qui associerait définitivement l’objet technique à l’objet vivant. L’objet technique occupe « une place intermédiaire » entre l’objet naturel et la représentation scientifique ; il n’est ni complètement naturel, ni complètement humain. C’est sur ce point que Simondon se démarque le plus radicalement de la cybernétique, qui tend à associer l’humain aux schèmes

49. MEOT, p.47-48.

50. Voir IPC, p.13 : « (...) l’être au sein duquel s’accomplit une individuation est celui en lequel une résolution apparaît par la répartition de l’être en phases, ce qui est le devenir ; le devenir n’est pas un cadre dans lequel l’être existe ; il est dimension de l’être, mode de régulation d’une incompatibilité initiale riche en potentiels. L’individuation correspond à l’apparition de phase dans l’être qui sont les phases de l’être. » Il y a là des points communs évidents avec la technologie du MEOT, explicitée plus haut ; dans la technologie de Simondon, l’objet technique est d’ailleurs régulièrement assimilé au vivant.

technologiques : ainsi, ce qui risque de rendre le travail de la cybernétique inefficace, c’est « le postulat initial de l’identité des êtres vivants et des objets techniques auto-régulés »<sup>51</sup>. Simondon nuance : « il ne faut pas confondre la tendance à la concrétisation avec le statut d’existence entièrement concrète »<sup>52</sup>. Il n’opère pas de « passage à la limite » ; le raisonnement analogique entre individuation et individualisation permet le passage de la méthode inductive de l’un à l’autre, il permet de leur donner un cadre commun d’étude ; il ne permet pas de « parler des objets techniques comme s’ils étaient des objets naturels »<sup>53</sup>. La méthode inductive est donc importée de la thèse primaire, via un raisonnement analogique dont Simondon délimite nettement la portée : on ne peut pousser l’analogie jusqu’à l’identité, comme l’affirment les cybernéticiens.

Si certains commentateurs ont décrit l’induction dans la thèse primaire, aucun ne s’est intéressé à proprement parler aux mécanismes de l’induction dans le MEOT<sup>54</sup>.

---

51. MEOT, p.49.

52. MEOT, *ibid.*

53. MEOT, *ibid.*

54. L’inspiration bachelardienne a ainsi été relevée par J-H. Barthélémy dans *Simondon ou l’encyclopédisme génétique* [10]. L’ouvrage en question débute par un chapitre intitulé « Le « réalisme des relations » : préalable épistémologique » - ce qui souligne au passage l’importance de l’épistémologie pour étudier Simondon. J-H. Barthélémy explique que la physique a une portée philosophique autorisant l’ontologie à se construire à partir de son enseignement. La désubstantialisation de la physique contemporaine (trouvant sa source dans la relativité einsteinienne, la thermodynamique et la physique quantique) et le « réalisme des relations » qui en découle sont la source de la thèse anti-substantialiste au coeur de l’ontologie génétique de l’individuation de Simondon.

## I.B.2 Etude de cas d’un exemple technologique

En effet, les analyses sur l’épistémologie de la technologie restent d’ordre général, car elles ne sont pas directement dirigées vers la méthode inductive du MEOT. Les explications des commentateurs ont tendance à évacuer la question concrète du développement des exemples : ils citent les phrases conclusives sans montrer le développement qui les a amenées ; ils appellent la conclusion sans montrer le chemin menant vers elle<sup>55</sup>. Afin de donner droit de parole à la démarche inductive, nous nous proposons de nous livrer à un exercice d’analyse structurée de la technologie déployée par Simondon. Nous nous tiendrons ici à une brève analyse de l’argumentation d’un exemple tiré de la première partie du MEOT, attendu que le même procédé est dispensé dans le reste de la première partie, et dans les autres ouvrages abordant la technique<sup>56</sup>.

Ainsi, dans la « Genèse de l’objet technique : le processus de concrétisation »<sup>57</sup> puis dans « Evolution de la réalité technique ; éléments, individu, ensemble »<sup>58</sup>, les exemples sollicités ont valeur d’arguments ; ils permettent la montée en généralité du discours technologique. Alors que l’introduction du MEOT maintient un

55. Certains commentateurs le font, comme Chabot dans *La philosophie de Simondon* [33], mais ils restent rares.

56. Le MEOT déploie cette stratégie argumentative de bout en bout ; c’est également le cas, dans une moindre mesure, dans la PST et dans nombre d’autres textes de Simondon, comme ceux du recueil *L’invention dans les techniques* [89] ou les textes de *Sur la technique* [93].

57. MEOT, Partie I, Chapitre I, sur les lignées d’objets techniques.

58. MEOT, Partie I, Chapitre II, sur les échelles de technicité.

discours d’ordre général, le fonctionnement des objets, dans leur matérialité concrète et précise, intervient dès le début du chapitre I. Les exemples sont très présents dans la première partie ; la place qui leur est attribuée dans l’argumentation augmente quantitativement et leur développement peut atteindre jusqu’à trois pages<sup>59</sup>.

Intéressons-nous aux toutes premières pages du MEOT pour illustrer cette méthode : la première sous-partie du premier chapitre, « Genèse de l’objet technique : le processus de concrétisation », intitulée « objet technique abstrait et objet technique concret », aborde le processus de concrétisation via le déroulement d’un exemple technique trois pages durant. La technicité de l’objet choisi, le moteur à essence, y est déployée dans les moindres détails et pose peu à peu des arguments qui mène à l’idée de concrétisation, énoncée clairement dans les deux dernières phrases de la sous-partie.

Simondon commence par affirmer donc que l’objet technique est « soumis à une genèse »<sup>60</sup>, tout en soutenant que la simple description de ce dernier n’est pas le moyen adéquat pour le comprendre. Il énonce sa méthode : plutôt que de partir de « l’individualité de l’objet, ou même de sa spécificité, qui est instable » (postulat de

---

59. Voir MEOT, p.19, puis p.20-21 et 23 (deux pages et demie sur la lignée phylogénétique du moteur à essence), p.25-26 (sur le refroidissement du moteur à combustion interne pour illustrer le stade d’abstraction de l’objet technique), p.28-30 (sur l’évolution du tube électronique avec la lampe de radio, pour montrer l’évolution par bonds), p.32 (sur le passage du tube de Crookes au tube de Coolidge pour illustrer la spécialisation des unités fonctionnelles dans le processus de concrétisation). On trouve d’autres exemples, qui suivent tous le même procédé argumentatif, jusqu’à la fin de la seconde partie.

60. MEOT, p.19.

départ), il faudra donc « renverser le problème », c’est-à-dire étudier les « critères de la genèse » de l’objet. L’objet est alors défini comme « unité de devenir »<sup>61</sup>. L’étude génétique des objets techniques est ainsi postulée théoriquement.

L’exemple du moteur à essence intervient immédiatement après pour soutenir la prise de position philosophique :

*« Le moteur à essence n’est pas tel ou tel moteur donné dans le temps ou dans l’espace, mais le fait qu’il y a une suite, une continuité qui va des premiers moteurs à ceux que nous connaissons et qui sont encore en évolution. »<sup>62</sup>*

Succède à l’introduction de l’exemple une montée en généralité théorique. La notion de « lignée phylogénétique » comme indice de comparaison est introduite, et la genèse de l’objet est précisée : « l’être technique évolue par convergence et par adaptation à soi<sup>63</sup> ». Un exemple technique illustre immédiatement cette affirmation, en comparant les moteurs des années 1910 au moteur automobile d’« aujourd’hui » (années 1960). Ce dernier appuie le fait que le moteur ne se définit pas par son usage en tant que moteur, mais par le fait qu’il soit le siège d’un processus technique bien spécifique. Il pourrait en être fait un tout autre usage, cela resterait essentiellement un moteur. Un exemple technique très concret vient appuyer encore plus encore la mention des différents moteurs. Simondon passe alors dans

61. MEOT, *ibid.*

62. MEOT, *ibid.*

63. MEOT, p.20

le registre du technicien : il parle de « grippage », de « coulage », « d’allumage »; la théorie de la concrétisation pointe : « dans un moteur actuel, chaque pièce importante est tellement rattachée aux autres par des échanges réciproques d’énergie qu’elle ne peut pas être autre qu’elle n’est<sup>64</sup> ». En témoigne l’exemple qui suit de la « forme de la chambre d’explosion », de la forme « des soupapes », « du piston »<sup>65</sup>. Le fonctionnement du moteur corrobore la théorie de la concrétisation simondonnienne mieux qu’un argument tiré du domaine discursif, logique. Le terme « d’abstrait » et de « concret », mentionné comme programme de la première sous-partie du chapitre I<sup>66</sup>, apparaît alors dans le corps de l’exemple technique, au terme de son déploiement : « On pourrait dire que le moteur actuel est un moteur concret, alors que le moteur ancien est un moteur abstrait.<sup>67</sup> » Les deux concepts primordiaux de la technologie simondonnienne sont introduits pour la première fois dans un exemple, ce qui ménage l’induction de l’exemple technique précis à la théorie de technologie générale.

L’exemple des moteurs privilégie le moteur concret comme facteur de progrès technique. Pour appuyer son propos, Simondon analyse deux modèles de moteurs espacés dans le temps au prisme d’un problème technique précis : celui des ailettes à refroidissement. La recherche du compromis est le lot du moteur ancien. Les ailettes à refroidissement y sont des « structures de défense », elles sont « comme ajoutées de l’extérieur » à la structure

64. MEOT, p.21

65. MEOT, *ibid.*

66. Première sous-partie intitulée « Objet technique abstrait et objet technique concret ».

67. MEOT, p.21.

d'ensemble et servent uniquement au refroidissement ; elles n'ont « qu'une seule fonction ». Au contraire, la « convergence des fonctions » est illustrée par le moteur moderne : ici, les ailettes de refroidissement n'ont pas qu'une fonction de refroidissement ; elles jouent en plus un « rôle mécanique<sup>68</sup> » dans l'ensemble et deviennent indispensables (si on les supprime, non seulement le moteur n'est plus refroidi, mais en plus il ne peut plus fonctionner). Les éléments sont interdépendants ; la culasse dépend maintenant des ailettes. L'objet est plus concret, plus cohérent avec lui-même. Simondon insère au milieu de son développement technique une phrase qui prépare la conclusion finale : « le développement de cette structure unique n'est pas un compromis, mais une concomitance et une convergence ». Une phrase récapitulative vient conclure le tout : « le problème technique est donc plutôt celui de la *convergence des fonctions dans une unité structurale* que celui d'une recherche de compromis entre des exigences en conflit<sup>69</sup> ».

Par ailleurs, Simondon prend en compte l'existence de cas dits « mixtes », qui sont identifiés comme « résidus d'abstraction<sup>70</sup> » dans l'objet technique. Ici, l'objet technique a valeur d'argument mais aussi d'*objection* adressée à la théorie simondonienne. L'objection est émise par le fonctionnement de l'objet technique lui-même ; la concrétisation n'est pas toujours parfaite. La théorie de la concrétisation s'en accommode et se teinte de nuances : elle devient moins unilatérale.

Ce long développement sur les moteurs à essence, filé sur plusieurs pages, aboutit à un paragraphe très synthé-

68. MEOT, p.22

69. MEOT, p.22.

70. MEOT, *ibid.*

tique, résumant la théorie de la concrétisation :

*« L’objet technique existe donc comme type spécifique obtenu au terme d’une série convergente. Cette série va du mode abstrait au mode concret : elle tend vers un état qui ferait de l’être technique un système entièrement cohérent avec lui-même, entièrement unifié<sup>71</sup>. »*

Simondon utilise le même type de stratégie argumentative tout au long de la première partie de son ouvrage. Sa théorie s’affine au fur et à mesure que des « objections », émises par le fonctionnement des objets techniques eux-mêmes (ainsi le cas d’adaptation « mixte » qui donne lieu à un « résidu d’abstraction »), la font paraître moins massive. Dans la méthode simondonienne, es schèmes mécaniques servent de ligne épistémologique pour fonder une ontologie génétique de l’objet technique<sup>72</sup>.

### I.B.3 Reproduire le geste simondonien

L’objet technique a valeur argumentative ; c’est là la méthode simondonienne, la voie par laquelle nous souhaitons prolonger la pensée de l’auteur. Il est possible de questionner une telle méthode : le panel d’objets techniques que ce dernier étudie est constitué principalement de machines. Peut-on la prolonger en l’appliquant à un

<sup>71</sup>. MEOT, p.23.

<sup>72</sup>. Tout comme le « réalisme des relations » issu de la physique avait permis d’établir le processus d’individualisation dans la thèse primaire.

objet numérique ? Fournit-elle des cadres de compréhension acceptables, ou n'est-elle adaptée que pour les objets matériels ? Nous nous proposons de répondre à cette question en employant les cadres de ce que nous avons défini comme méthode simondonienne pour l'analyse du logiciel libre. Cette dernière est délimitée par :

- La reprise des schèmes généraux de la technologie simondonienne telle qu'énoncée et mise en œuvre dans le MEOT, savoir : 1. L'analyse de la lignée phylogénétique de l'objet, et 2. L'analyse des degrés de technicité présents dans cet objet (élément, individu, ensemble).
- La considération de la *valeur argumentative* du fonctionnement de l'objet technique ; le fait que ce dernier est potentiellement porteur d'*objections* pour la théorie qui le met en avant.

Il y a dans la philosophie de Simondon, au-delà de sa portée génétique et métaphysique, un geste fondamental. Toute l'œuvre de Simondon dialogue avec l'introduction du MEOT : il faut réconcilier la culture et la technique, et pour ce faire, il faut que le philosophe déploie une méthode d'analyse qui intègre les objets techniques et leur donne valeur d'argument à part entière, tout autant qu'une théorie. En témoigne la fin du chapitre I, où la précision de la méthode inductive est directement liée à l'énonciation du programme de la technologie simondonienne. C'est également tout le sens des liens de Simondon avec l'encyclopédisme : l'auteur incarne le modèle d'un pont entre culture et technique, qui prolonge celui des tenants de l'Encyclopédie.

La philosophie de la technique de Simondon est un programme de travail, en ce qu'elle soulève des problématiques liées à la conception d'une technicité fonction-

naliste, et en ce qu'elle présente une méthode particulière. Elle est à la fois cadre de réflexion et geste à prolonger. C'est pourquoi nous avons décidé de nous intéresser à un objet numérique contemporain en décrivant précisément certains points de son fonctionnement : il s'agissait de lui laisser la parole. Les cadres conceptuels d'un tel dialogue entre la philosophie de la technique simondonienne et le logiciel sont maintenant tous en place. Nous verrons comment le logiciel et son fonctionnement met à l'épreuve la philosophie de la technique simondonienne et sa définition fonctionnelle de la technicité.

## I.C Confronter la pensée de Simondon à l'informatique

*« En choisissant de penser les significations philosophiques de l'ordinateur, je voudrais donc donner raison à Simondon par-delà les objets techniques qu'il avait pensés en son temps. »*

*– J-H. Barthélémy, « Les significations philosophiques de l'ordinateur » [8].*

### I.C.1 Etat des travaux sur Simondon et l'informatique

Confronter un objet numérique à la philosophie de la technique simondonienne impliquait de prendre en compte les travaux déjà menés en ce sens. Il est apparu

alors qu’il existait des impensés de l’exégèse simondonienne.

De même qu’on ne s’intéresse pas aux explications proprement techniques et à leur déploiement (on considère Simondon comme original de par son *objet* mais non par sa *méthode*), on ne reproduit pas le geste simondonien. De fait, les plus enclins à appliquer de nouveau une telle méthode sont les ingénieurs plus que les philosophes, dont la praxis n’est pas encore le fort. Un article paradigmatique de cette situation est celui de J-H. Barthélémy, « Simondon et l’ordinateur », qui incite à la réévaluation de la philosophie de la technique simondonienne par rapport au constructivisme social de Feenberg. Ce qui est intéressant, c’est que selon l’auteur, cette réévaluation ne doit pas se faire par un point de commentaire particulier de Simondon, mais par l’étude d’un objet technique concret. La praxis, l’objet technique revient au premier plan pour dialoguer avec la philosophie de Simondon. J-H. Barthélémy appelle à l’étude de l’objet concret ordinateur, sans pour autant théoriser consciemment sa démarche. Il ne note à aucun endroit le glissement épistémologique : ce n’est pas la meilleure connaissance de tel ou tel texte de l’auteur qui en permettrait une meilleure compréhension, ou bien sa comparaison avec d’autres auteurs ; c’est bel et bien sa confrontation renouvelée à un objet technique contemporain, c’est-à-dire l’*emploi de la méthode simondonienne*. Les conditions de cette confrontation ne sont cependant pas explicitées ou détaillées ; encore une fois, ce sont plus les conséquences théoriques qui sont anticipées, sans laisser vraiment la parole à l’objet technique - cette responsabilité est transférée aux générations de futurs chercheurs, forts d’un appareillage déjà solide dans le commentaire simondonien.

Dans la continuité de cet appel, certains auteurs ont analysé Simondon pour comprendre le domaine informatique. Plusieurs approches notables ont été développées, vis-à-vis desquelles nous nous positionnerons.

L'effort le plus général pour penser Simondon et les réseaux est probablement celui de Ludovic Duhem, dont les travaux<sup>73</sup> posent de façon très précise les conditions pour construire une pensée du numérique à partir de Simondon. Son intervention au colloque de Cerisy<sup>74</sup> fait preuve d'une largeur de vue sur les problématiques simondoniennes, et d'une très grande lucidité sur les impensés simondoniens qui en découlent. L'approche de Ludovic Duhem est axée autour de la réticulation comme concept central – voire final – de la pensée de Simondon. Il insère la genèse des objets techniques de Simondon dans un processus de réticulation de plus en plus serré : tandis que l'élément n'est relié qu'à lui-même, les individus techniques réticulent les éléments et le milieu qui leur est associé, et au-dessus d'eux, les ensembles sont eux-mêmes des réticulations d'individus techniques. La technique complexifie peu à peu ses relations : son organisation progressive est décrite par L. Duhem comme « un monde de réticulations ». Elle tend à devenir « un réseau de réseaux à la fois indépendant et articulé au réseau naturel et humain »<sup>75</sup>. Ainsi, pour L. Duhem, la dernière phase d'évolution des objets techniques (non mentionnée dans le MEOT<sup>76</sup>) est celle du réseau. La réalité réticulée

73. Voir à ce propos l'article « Penser le numérique avec Simondon » [44].

74. « La réticulation du monde » [46] in *Gilbert Simondon ou l'invention du futur* [21].

75. "La réticulation du monde" [46].

76. Le MEOT, on l'a vu, fait apparaître pleinement l'essence de

consacre, aux dires de Simondon, « la technicité à l'état pur »<sup>77</sup>.

L. Duhem insiste sur l'importance du concept de réticulation dans la philosophie simondonienne, en tant que dernier stade de la technicité ; pourtant, il prend garde de ne pas assimiler hâtivement ce qu'il nomme « réseau technique postindustriel », et que Simondon appelle de ses vœux, avec Internet. Il se contente d'effectuer un travail de clarification conceptuelle, et énonce clairement les pistes qu'il resterait à parcourir pour « construire Internet comme paradigme de la réticulation du monde ». La démarche philosophique doit emprunter des chemins proprement techniques, à l'instar de ce que Simondon a effectué, pour prétendre ensuite penser la réalité réticulée dans son ensemble (et ses correspondances plus ou moins exactes avec la théorisation qu'en fait Simondon). Ainsi, avant que de se hâter aux analogies trop superficielles, les chercheurs contemporains doivent selon lui se pencher sur deux points cardinaux et « internes » à l'objet réticulé qu'est l'ordinateur :

- L'analyse de la « machine à information » spécifique qu'est l'ordinateur.
- L'analyse du langage informatique.

Ces deux points, relevés par L. Duhem mais non approfondis par lui, esquissent un mouvement d'investigation allant de plus en plus profond dans l'analyse de l'individu technique réticulé. A l'intuition philosophique brute de notre réalité contemporaine réticulée (provoquée par

---

la technicité à l'ère des ensembles techniques. La PST, entre autres, revient sur cette assertion et met le réseau au premier plan. Le mouvement théorique est encore plus clair dans le recueil de textes *L'invention dans les techniques*, texte sur lequel L. Duhem appuie son propos.

77. *L'invention dans les techniques*, p.101. [89]

les réflexions de Simondon sur le sujet, mais 50 années plus tôt, sans connaître les objets techniques qui nous sont contemporains), succède immédiatement une exigence de vérification de cette intuition. Pour ce faire, le programme que L.Duhem énonce s'apparente à une remontée vers les niveaux de technicité moins réticulés qui sont partie intégrante de cette mise en réseau générale des objets techniques. Il appelle d'abord à l'analyse de ce que l'on peut appeler sans trop d'erreur un individu technique, à savoir l'ordinateur, puis à ce qui le compose, le code informatique.

De telles réflexions font écho aux trois interventions de la table ronde autour de « Simondon et l'informatique », lors du même colloque de Cerisy. Ces dernières abondent dans le même sens : la pensée de Simondon, si elle n'aborde pas frontalement l'informatique, recèle de « pistes » pour étudier ce domaine.

Frédéric Pascal<sup>78</sup> relève le fait que l'informatique répond d'un « échafaudage de réseaux », et fait intervenir la distinction cardinale, dans l'étude de l'ordinateur, entre le *hardware* et le *software*<sup>79</sup>. Ainsi, au réseau formé d'une part « d'un système d'alimentation électrique, d'une transformation de l'énergie et de sa distribution, de son raffinement en un système électronique complexe composé de semi-conducteurs intégrés » (*hardware*, partie électronique), correspond d'autre part celui formé par « l'enchevêtrement de multiples systèmes de codages mis en œuvre par des communautés distinctes » (*software*, la partie du logiciel). Ces deux réseaux ne cohabitent pas côte à côte ; ils correspondent et sont inter-

78. In "Simondon et l'informatique I" [84].

79. Voir glossaire, "Opposition *Hardware/Software*."

dépendants ; ainsi « la bulle du *software* peut, à maints égards, être interrogée du point de vue de son rapport avec le *hardware* ».

J. Grosman<sup>80</sup>, enfin, apporte des pistes fondamentales pour étudier l'ordinateur à l'aune des concepts simondoniens. Ce dernier revient dans un premier temps sur l'importance de la théorie de l'information chez Simondon, que ce dernier s'emploie à refonder et à rendre plus universelle ; il y voit la possibilité d'un examen du codage informatique, compris non comme « un ensemble de lignes écrites dans un langage de programmation », mais comme « ensemble d'instructions et de valeurs correspondant directement à des différences de tensions significatives données à la machine ». Le codage est ce qui permet de restreindre la « marge d'indétermination » de la machine ouverte : « la programmation est ce qui permet de restreindre – temporairement- la forme de la machine, parmi les variations de forme initialement prévues. » Apparaît alors la figure du·de la hacker·euse, que J. Grosman met en tension avec celle du·de la technicien·ne énoncée par Simondon, car différente de ce dernier : elle suppose la notion de bricolage (« le hacker tient également quelque chose du bricoleur »), que la·le technicien·ne ne revêt pas. J. Grosman esquisse ici une mise en tension des concepts de technicité et de finalité, grâce à la comparaison du paradigme de la machine-outil et de celui de l'ordinateur, machine-ouverte. Le codage est pensé comme lien privilégié entre l'homme et la machine via la théorie renouvelée de l'information.

Le second point soulevée par J. Grosman concerne la notion de schème technique, ou plus généralement, celle de technicité : ce dernier part du principe que la techni-

80. In "Simondon et l'informatique II" [58].

ité est un concept ouvert, propre à évoluer, et que les critères de concrétisation ne correspondent plus exactement aux systèmes électroniques et informatiques – il faut en penser d'autres, grâce au contact étroit avec l'objet informatique. Il nous faut comprendre de nouveaux schèmes techniques pour énoncer sur eux une pensée réflexive et philosophique : « la connaissance de l'individuation est inséparable de l'individuation de la connaissance ».

J. Grosman préconise enfin la reproduction du geste simondonien. Pour connaître l'informatique, il faut nous rendre disponible à ce nouvel objet technique et à ses schèmes de fonctionnement. Le comprendre de l'intérieur nous permettra d'énoncer sur lui des choses qui n'auraient pu, sans ce geste *en dehors de la pensée philosophique et conceptuelle*, advenir dans le champ réflexif et langagier. J. Grosman rappelle que le propre de la méthode simondonienne est de s'extraire de la philosophie pour parler d'autres langages, et, fort de cette expérience, tenter de transmettre ces nouveaux modes de réflexion dans le langage philosophique.

Plusieurs constats émergent de ce rapide tour d'horizon. Des jalons d'importance ont été posés pour penser Simondon et l'informatique. L'apport de ces travaux précurseurs permet de justifier le sens dans lequel veut s'engager notre étude. Ils pointent ce qui, chez Simondon, relève d'une ébauche de réflexion sur le sujet, même si cela n'a pas été traité frontalement :

- Sa pensée des réseaux,
- Sa vision de la théorie de l'information, qui part d'une critique de l'automate et cherche à se démarquer de la théorie de Wiener. Ce qui différencie l'automate de la machine est le concept de marge d'indétermination, qui permet de recevoir

de l’information (l’automate, individu stable et déterminé, ne peut recevoir d’information<sup>81</sup>. Réciproquement, seul un individu métastable, qui contient une marge d’indétermination localisée, peut recevoir de l’information et transformer sa structure.)

— Cette notion de marge d’indétermination entraîne des réflexions quant au codage. La machine se restructure, se réorganise selon les informations qu’on lui apporte : « La programmation est ce qui permet de restreindre – temporairement – la forme de la machine, parmi les variations de forme initialement prévues. »<sup>82</sup>

On voit se dessiner la possibilité d’une relation ouverte entre la machine à calculer et la machine à calculer. L’homme se fait interprète des machines, entre en synergie avec elles : il y a couplage entre l’homme et la machine « lorsqu’une fonction unique et complète est remplie par deux êtres »<sup>83</sup> Cette synergie fonctionnelle pourrait être réalisée, dans le cadre de l’ordinateur, par le biais du codage.

La théorie de Simondon recèle donc d’intuitions qui appellent à être prolongées – les études citées sont le résultat d’une table ronde, et aucun ouvrage n’a à ce

---

81. J. Grosman mentionne à ce sujet la comparaison de Simondon : « l’exemple même de l’automate est la monade leibnizienne, pensée à partir du paradigme du pendule, dont le dieu horloger assure le devenir. Pour Simondon, la stabilité caractérise l’impossibilité de la transformation du système, de son devenir. » Le concept de marge d’indétermination est cardinal pour la suite de notre étude : nous y reviendrons lors de la seconde partie.

82. GROSMAN J., ”Simondon et l’informatique II” [58].

83. MEOT, p.173.

jour été consacré exclusivement à l'étude de Simondon et de l'ordinateur.

## I.C.2 Positionnement de l'étude

Suite à ces travaux déjà effectués sur les liens entre Simondon et l'informatique, notre travail se positionne sur plusieurs plans.

D'abord, il ne veut pas être une simple mise en parallèle des caractéristiques de la technicité chez Simondon à un objet contemporain ; pour réellement suivre la méthode simondonienne dans toute sa radicalité, nous nous devons d'effectuer un vrai travail de découverte et de compréhension de l'objet technique étudié – pour ensuite en rendre compte en des termes philosophiques. Le parti-pris méthodique pour lequel nous optons est celui d'un dialogue entre la théorie et l'objet concret.

Il nous semble ensuite que la distinction que pose F. Pascal entre le *hardware* et le *software* est fondamentale. Nos questionnements partiront non de l'objet technique dans sa matérialité qu'est l'ordinateur (*hardware*), mais de l'objet numérique qu'est le logiciel (*software*)<sup>84</sup>. Qualifier un *software* d'objet technique n'a rien d'évident dans le cadre de la théorie simondonienne. Dès lors que le logiciel est composé d'information binaire et de lignes de code, est-il seulement possible d'établir des liens effectifs avec la théorie simondonienne, qui porte sur des machines matérielles et concrètes ? Notre seconde partie s'attachera

84. Le fait de partir de l'étude du logiciel en tant qu'objet immatériel n'exclut pas de prendre en compte le *hardware* qui le sous-tend.

à tenter de résoudre les difficultés que posent une telle problématique : le logiciel est-il un objet technique ?

Pour répondre à cela, nous utiliserons les outils conceptuels développés lors de notre première partie, et que nous avons définis préalablement comme caractéristiques de la philosophie de la technique simondonienne :

- Celui de la méthode simondonienne : nous étudierons un objet numérique particulier, le logiciel. Notre analyse portera avant tout sur le logiciel dans sa généralité. Pour l’examen de points techniques plus précis, nous avons choisi de nous pencher sur le navigateur libre Mozilla Firefox<sup>85</sup>. Grâce au fil directeur de la méthode, nous verrons les apports réciproques entre la théorie simondonienne et l’objet numérique :
- L’objet numérique peut-il être construit comme objet technique à partir des cadres conceptuels de la philosophie de la technique simondonienne, précédemment énoncés ? (partie II) Quels sont les enjeux d’une telle définition pour la philosophie de la technique de Simondon ?
- Inversement, la philosophie de la technique simondonienne trouve-t-elle des réponses aux problématiques qu’elle laisse ouvertes ? Nous avons identifié deux problématiques principales, celle de la localisation technicité et celle des représentants de la technique. Si nous parvenons à identifier le logiciel comme objet technique, alors ce dernier pourra fournir de nouvelles réponses à ces deux questions. (partie III)

---

85. Dont le choix est justifié en II.A

## Partie II

# Etude de technologie génétique : le logiciel est-il un objet technique ?

### II.A Définition et problématisation de l’objet numérique

*« A modern computer system is like a huge iceberg, and most people get to see only the top. »*

*– Noam Nisan et Shimon Schocken, The Elements of Computing Systems [81].*

La confrontation de l’objet numérique logiciel à la pensée de Simondon fait émerger plusieurs problématiques fondamentales : nous nous efforcerons d’en dégager les pistes principales, sans pouvoir toutes les appro-

fondir<sup>1</sup>. Pour montrer les conséquences psychosociales d’une nouvelle technicité du numérique et voir comment notre objet d’étude dialogue avec les deux axes soulevés dans la première partie, il nous faut d’abord poser que le logiciel est effectivement un objet technique – ce qui est tout sauf une évidence. En cela, la question qui compose le titre de cette seconde partie est une question réellement ouverte ; elle n’est pas fermée d’avance par une réponse positive. A ce stade de la réflexion, deux hypothèses s’offrent à nous :

— Soit le numérique peut être effectivement considéré comme un objet technique<sup>2</sup>, mais alors il inaugure un nouveau régime de technicité (ce que la philosophie de la technique de Simondon, considérée comme programme de travail, autorise).

Nous pourrions alors évaluer les conséquences d’une telle conclusion sur la conception de la technicité mise en avant par Simondon.

— Soit le numérique ne peut être considéré comme un

---

1. Le II.C, « Essai de correspondance », déploie des hypothèses qui nous semblent valides, bien qu’elles mériteraient d’être encore approfondies ; s’y consacrer dans le cadre d’un travail de mémoire déséquilibrerait l’argumentation et demanderait plusieurs mois de plongée dans l’univers informatique. Comprendre ce dernier et parvenir à en différencier les objets nous paraît propice à un travail de thèse sur plusieurs années (que nous nous promettons d’effectuer si cela est possible).

2. Nous nous plaçons ici dans les cadres de la philosophie de la technique simondonienne, qui produit une définition très claire de l’objet technique que nous allons préciser dans le paragraphe suivant. La question posée ici est en réalité : « Peut-on dire que le logiciel est un objet technique à partir de la définition de l’objet technique de Simondon ? » Nous déduisons en conclusion les conséquences plus générales pour la philosophie de la technique d’un tel questionnement.

II.A. DÉFINITION ET PROBLÉMATISATION DE L’OBJET NUMÉRIQUE 73

objet technique dans le cadre de la pensée simondonienne.

Si la première hypothèse se vérifie, alors seulement nous pourrions utiliser le logiciel comme « objection », comme renouvellement de la pensée simondonienne en troisième partie ; si ce n’est pas le cas, l’entreprise est vaine, car les deux objets ne se situeraient pas au même niveau conceptuel.

Réaliser ce programme implique deux temps préparatoires : d’abord, problématiser ce que signifie l’objet technique pour Simondon, et ensuite, constituer un objet d’étude numérique qui présente un intérêt particulier pour étudier le logiciel.

### II.A.1 L’objet technique pour Simondon

Définir le logiciel comme objet technique implique de comprendre réellement ce que l’objet technique signifie pour Simondon. A ce niveau, une étrangeté subsiste : Simondon ne définit jamais réellement ce dernier.

On l’a vu<sup>3</sup>, la définition de l’objet technique pour Simondon est avant tout une définition fonctionnaliste. Pour Simondon, l’objet technique est l’objet qui déploie de la technicité. Cette technicité est avant tout fonctionnement, agencement et influence réciproque de trois niveaux d’échelles technique (l’élément, l’individu, l’ensemble) dans une synergie de plus en plus grande : la concrétisation. Autrement dit, c’est le fait de fonctionner qui distingue l’objet technique de la chose, du simple objet. En cela, Simondon se distingue à la fois d’Aristote,

---

3. Voir I.A.1.

qui définit l’objet technique comme imposition d’une matière à une forme, et de Heidegger, qui dénie toute qualité ontologique à l’objet technique en le réduisant à l’utilité. Ces deux conceptions s’appuient sur la matérialité de l’objet pour en définir la technicité.

Une autre facette, implicite, de la définition de l’objet technique chez Simondon, est qu’elle repose également sur un critère de matérialité. Les exemples utilisés par l’auteur le montrent : la matérialité est le présupposé sous-jacent de l’objet technique. Ce dernier est donc défini en partie par les caractéristiques de la matière qui le compose : il est perceptible, tangible, doté d’une forme unique, soumis aux lois physiques, et impénétrable<sup>4</sup>.

Une question se pose alors : quelle est l’importance du critère matériel dans la définition de la technicité simondonienne ? Peut-on s’en passer ? Peut-on affirmer que l’objet numérique logiciel, bien qu’immatériel, soit un objet technique ?

Une telle assertion aurait pour effet de radicaliser *a posteriori* le fonctionnalisme de Simondon : le critère de matérialité sous-tendant les exemples choisis serait finalement secondaire par rapport au fonctionnement de ces derniers. La conception fonctionnaliste serait alors plus appropriée pour comprendre notre technicité contemporaine que ne le seraient la conception hylémorphique ou la conception utilitaire, reposant toutes deux sur une définition matérielle conceptuellement insuffisante pour penser l’objet numérique.

---

4. L’impénétrabilité est une propriété physique selon laquelle deux corps ne peuvent occuper le même lieu dans l’espace.

## II.A.2 Le navigateur, objet numérique représentatif du logiciel

Nous établirons le navigateur en tant que sous-catégorie technique intéressante de la classe générale du logiciel à partir :

- De sa cohérence interne légitimant son appellation générique « d’objet numérique »<sup>5</sup>, c’est-à-dire une entité regroupant sous une même unité fonctionnelle plusieurs individus particuliers. Ainsi, parler « du navigateur » est quelque chose de cohérent : les différents navigateurs web recouvrent des principes de fonctionnement communs.
- Et de son importance dans le domaine des objets numériques en général, justifiant le choix de cet objet et non d’un autre.

### Le navigateur comme unité fonctionnelle

Cette première tentative de définition est dirigée par un double effort. Il s’agit d’abord de poser une contextualisation de l’objet à des fins de pédagogie<sup>6</sup>. Il s’agit ensuite

---

5. Dans un premier temps, nous définirons le navigateur comme un objet numérique, à défaut de pouvoir le caractériser directement d’« objet technique », au sens simondonien du terme. Nous entendons par « objet numérique » tout logiciel achevé, qui fonctionne correctement et propose à l’utilisateur une gamme de fonctionnalités précises. La notion d’« objet numérique » est une notion intermédiaire qui est garantie par l’unité fonctionnelle de ce qu’elle désigne.

6. Pour des raisons de pédagogie et de lisibilité pour le plus grand nombre, nous essayons d’épargner au lecteur l’entrée de plein pied dans un objet technique précis (par exemple : « Tel est le cas d e

de répondre à notre problématique grâce à cette contextualisation, en montrant que tous les navigateurs peuvent être définis via les mêmes fonctionnalités générales.

Le navigateur est un logiciel<sup>7</sup> complexe dont le rôle est de servir d’interface entre l’utilisateur et le Web<sup>8</sup>. Techniquement, c’est un visualisateur de fichiers distants.

La notion de logiciel est la plus fondamentale<sup>9</sup>. Le logiciel, aussi appelé *software*, fonctionne en couple avec la notion de *hardware*. Au versant logique, scriptural du *software*, correspond le matériel concret, électronique du *hardware*<sup>10</sup>. Partons de la plus grande abstraction, celle du logiciel<sup>11</sup>. Ce dernier se divise en deux parties : l’interface

---

l’évolution du tube électronique... » sans plus de préambules sur ce qu’est, en général, le tube électronique, MEOT, p.28). Nous avons voulu mettre à disposition des savoirs techniques pour le lecteur, afin de lui donner des outils critiques conséquents. D’où la nécessité de ces pages descriptives, inutiles pour un·e informaticien·ne, mais fondamentales pour qui ne connaîtrait pas le domaine informatique. Le glossaire et les annexes à la fin du travail vont également dans ce sens. Ces explications d’ordre général servent d’introduction au fonctionnement plus précis du navigateur qui sera développé en II.C.

7. Voir glossaire, “Logiciel (du point de vue fonctionnel)”.

8. Autrement dit, « The browser main functionality is to present the web resource you choose, by requesting it from the server and displaying it on the browser window. » Tali Garsiel, « Understanding how browser work »[56] .

Le présupposé sous-jacent que contient cette définition est le fait que le navigateur en général puisse être considéré comme objet numérique cohérent.

9. Pour un manuel d’introduction aux bases de fonctionnement de l’informatique, voir D. Ince, *The computer : A very short introduction* [65] .

10. Voir glossaire, “Opposition *hardware/software*”.

11. Ici, nous spécifions les liens entre le *hardware* et le *software* en remontant de la matérialité de l’un vers l’abstraction de l’autre, se-

utilisateur d’une part, et le code source de l’autre<sup>12</sup>. L’interface utilisateur constitue le rendu final du logiciel, ce qui le fait apparaître comme un objet technique entier et cohérent. Le code source est comparable au plan d’architecture du logiciel<sup>13</sup>. Il se matérialise sous la forme d’une série de fichiers contenant des instructions pour la machine, écrites en différents langages de programmation. Ces fichiers sont autant de sous-ensembles fonctionnels du logiciel. Le code source organise entre eux les différents sous-ensembles du logiciel ; il est une structure hiérarchique qui permet leur maniement par les programmeur·euse·s. Les lignes de code qui le composent sont autant de lignes d’instructions pour faire effectuer à la machine une action définie. Elles sont écrites en langage compréhensible par l’humain (syntaxe, grammaire, conventions diverses) ; une fois écrites et organisées, elles doivent être traduites en un langage qui soit compréhensible par le *hardware*, par la partie matérielle du système informatique. C’est le rôle d’un autre logiciel, le compilateur : ce dernier interprète puis traduit les lignes de codes des différents langages en langage binaire<sup>14</sup>, le seul qui

---

lon une logique « bottom-up ». Nous suivons en cela (en simplifiant considérablement l’explication) la méthode de N. Nisan et S. Schocken dans *The Elements of Computing Systems* [81] : « The multi-tier collection of abstractions underlying the design of a computing system can be described top-down, showing how high-level abstractions can be reduced into, or expressed by, simpler ones. This structure can also be described bottom-up, focusing on how lower-level abstractions can be used to construct more complex ones. »

12. Voir glossaire, “Code informatique, code source”.

13. Nous produisons ici une définition principalement descriptive du code source, à des fins pédagogiques. Nous tenterons en II.B.2 de problématiser la notion de code informatique.

14. En réalité, le processus varie et peut être un peu plus complexe, mais cela n’a pas une importance fondamentale pour le niveau où

soit compréhensible par l’ordinateur. Le langage binaire est constitué d’une succession de 1 et de 0 dont le décryptage serait impossible pour l’homme. Nous voyons là se déployer la logique booléenne au fait de tout système informatique. En dernier recours, ces 1 et ces 0 sont traduits en instructions physiques; 1 équivaut à un circuit ouvert suite à une décharge électrique, 0 à un circuit fermé. Les circuits sont le composant des puces informatiques, composées elles-mêmes de silicium, que contient notre ordinateur. L’information charriée par les lignes de code (par exemple : effectuer 1+1) se stocke alors dans une case de la mémoire et s’active à l’écran de l’ordinateur par chemin retour (si un affichage est requis)<sup>15</sup>.

Le logiciel est donc le plus haut niveau d’abstraction qui existe au sein d’un système informatique. Il renferme des logiques complexes, à la croisée de la matérialité et de l’abstraction<sup>16</sup>. Il est à la fois un objet cohérent du point de vue de l’interface utilisateur (le navigateur Mozilla Firefox lorsqu’il est lancé depuis le bureau) et un code source exécutable et exécuté, contenant une série d’instructions à la machine. Ce code source est modifié par les programmeur·euse·s, qui ont la maîtrise des différents (ou d’un des) langages composant le logiciel; il est exécuté par toutes les machines qui utilisent le logiciel, afin que ces dernières suivent ses instructions.

Le navigateur est donc un logiciel et, qui plus est, un lo-

---

se situe notre analyse.

15. Pour une illustration de ce mécanisme, Annexes, FIG – A.

16. C’est pourquoi nous n’emploierons pas le mot « virtuel » pour désigner un tel objet : si ce dernier est bien composé de langages qui n’ont a priori rien de matériel, les commandes du logiciel ont un effet direct sur le fonctionnement matériel de la machine, qui reconfigure sa mémoire en fonction de ses ordres.

II.A. DÉFINITION ET PROBLÉMATISATION DE L’OBJET NUMÉRIQUE 79

giciel extrêmement complexe. Il existe différents niveaux de complexité au sein des logiciels ; certains ont un rôle fixe, n’interagissent qu’avec un nombre d’objets numériques limités, et ont une tâche à effectuer qui est relativement restreinte. D’autres, et c’est le cas du navigateur, sont complexes : leur rôle principal est constant, mais leurs fonctions ne cessent de s’élargir ; leurs interactions avec d’autres objets numériques sont nombreuses, et les tâches qu’ils effectuent sont très étendues.

Au-delà de son appartenance générale à une classe d’objets numériques (les logiciels), le navigateur a une fonctionnalité spécifique : il s’agit d’un visualisateur de fichiers distants, qui fait donc office d’interface entre l’utilisateur et le Web<sup>17</sup>gl. Ce dernier récupère et affiche les pages HTML demandées par l’utilisateur (par le biais de la barre d’adresse) via le *networking*<sup>18</sup>. Le Web, à proprement parler, est un système de nommage via des standards prédéfinis (par des organisations, en tête desquelles la W3C<sup>19</sup>). Il sert à communiquer de façon uniforme avec des serveurs situés dans toutes les parties du monde. Ces serveurs sont des entités matérielles qui hébergent de façon stable le contenu d’un site web ; ils ont besoin de recevoir une demande pour renvoyer ensuite la page qu’on leur demande d’afficher. Le Web est donc une architecture, une entreprise de recensement et de standardisation plus qu’une entité pleine qui contiendrait tous les sites ; au contraire, ces derniers sont dispersés aux quatre coins de la planète et il faut bien un outil qui puisse faire appel

17. Voir glossaire, “Web/Internet”.

18. Il n’existe pas de réel équivalent du terme en français, mais cela pourrait se traduire par « système de mise en réseau ».

19. L’organisation *World Wide Web Consortium* est la principale institution définissant les standards du Web.

à eux.

C’est pourquoi le rôle du navigateur est fondamental ici ; il est l’interlocuteur qui communique avec « ce qui s’avère être un écosystème fluctuant d’acteurs, d’usages et de dispositifs »<sup>20</sup>. Autrement dit, le navigateur est l’architecture logicielle qui permet d’appeler les pages web. A cela s’ajoute immédiatement deux conditions : l’utilisateur doit pouvoir appeler ces pages sans que cela ne soit trop ardu, donc par le biais d’une interface utilisateur simplifiée, tout en conservant malgré tout l’intrication de commandes complexes nécessaire au fonctionnement d’une telle architecture. Le navigateur est donc la matrice qui permet d’afficher les pages web avec une grande facilité<sup>21</sup>. Pour cela, il fonctionne en deux temps : lorsque l’on tape `www.philosophie.ens.fr` dans la barre de recherche de son interface utilisateur, le navigateur en fait la demande aux serveurs via le protocole HTTP<sup>22</sup> ; puis, une fois la page arrivée, plusieurs enchaînements techniques s’effectuent pour que la page s’affiche de la façon la plus lisible possible. Ces deux étapes cachent une architecture et une ingénierie logicielle complexe, composée de plusieurs éléments principaux. Ce fonctionnement en deux temps, diachronique (de la demande de la page web à son affichage par le navigateur dans le cadre de l’interface utilisateur), peut également être présenté de façon synchronique en deux entités structurellement dis-

20. Voir A. Monnin., *Vers une philosophie du Web* [78]. Nous reviendrons plus loin sur le sous-ensemble qu’est le *networking* et ses relations avec l’écosystème du Web, dans la tentative de parallèle entre le milieu associé simondonien et le web (Voir II.C.2).

21. Ainsi « The web browser acts as a *facilitator* ». D. Odell, *ProJavaScript RIA Techniques*, Chapitre 3, « Understanding how browsers work »[83].

22. Voir glossaire, ”HTTP”.

tinctes<sup>23</sup> :

- L’interface utilisateur [1]<sup>24</sup> : techniquement, il s’agit du cadre qui permet d’afficher la page web demandé ; elle facilite l’interaction de l’utilisateur avec le web et rend sa navigation intuitive. Soulignons que l’interface utilisateur n’est pas la page affichée ; elle est ce qui l’encadre et permet à l’utilisateur d’appeler les pages : elle contient généralement des onglets, des barres de menus déroulants, les flèches avant/après, les marques-pages, le bouton de réactualisation, et les quatre barres principales : barre d’outils, barre d’état, barre d’adresse, barre de recherche. Ainsi, « l’interface utilisateur peut être comprise comme la “peau” qui enveloppe le moteur de rendu »<sup>25</sup>.
- Le moteur de rendu, ou *browser engine* [2]. Il s’agit du dispositif qui fait fonctionner à proprement parler le navigateur<sup>26</sup>. Il est constitué de différents éléments (modules) permettant d’afficher à l’écran et

23. En réalité, ces entités correspondent dans certaines configurations ; pour les besoins de l’explication et la schématisation, il était plus aisé de les différencier (sans que cela ne soit préjudiciable à la compréhension de l’ensemble).

24. Pour le fonctionnement du navigateur illustré, voir FIG – B, C (les numéros dans le corps de cette énumération correspondent aux numéros du schéma C).

25. D. Odell, *Pro JavaScript RIA Techniques*, chapitre 3 (traduction personnelle) [83].

26. Le chapitre 3 de *Pro JavaScript Techniques* explique cette partition avec une analogie intéressante : « Like a car body, a well-designed web browser user interface reflects little of its inner working » : « De même que la carrosserie d’une voiture, une interface utilisateur bien construite est peu représentative du fonctionnement interne du navigateur » (traduction personnelle). Le parallèle avec une voiture montre l’envie de faire correspondre les schèmes techniques, et les analogies possibles.

de mettre en forme la page web demandée, notamment le *HTML parser* et le *CSS parser*<sup>27</sup>.

- Le *browser engine* dialogue enfin avec des informations que lui apportent d’autres composants, qui sont reliés directement à lui : le *networking* [3], qui récupère les pages web en dialoguant avec les serveurs ; le *Java Script interpreter*, qui exécute une partie du code informatique délivrée par le site web, le *UI Backend*, dont le rôle est d’afficher de petites composantes d’interface graphique<sup>28</sup>, et le *data persistence* [4], qui stocke en mémoire certaines données.

L’interface utilisateur simple et intuitive du navigateur web ainsi que son appellation en français (« navigateur ») peuvent induire en erreur. Ces parties émergées de l’iceberg spatialisent, sur une surface homogène (celle de l’écran, de l’interface utilisateur), le logiciel. L’interface est composée des boutons « retour en arrière » ou « en avant », qui induisent une continuité, un déplacement linéaire. Il ne faut pas oublier cependant que les interfaces utilisateurs ont une histoire propre et que, si elles structurent notre expérience du Web, il est primordial de ne pas s’en tenir à elles<sup>29</sup>. Elles induisent une perception du navigateur comme entité uniforme et linéaire. Ce n’est pas le cas : comme tout logiciel, on ne peut le comprendre

27. *Parser* signifie analyseur syntaxique.

28. De type *widget*.

29. Un exercice utile pour la déconstruction du navigateur comme entité spatiale et uniforme est de consulter des captures d’écran du premier navigateur inventé par Tim Berners-Lee (voir II.C.1). Cette confrontation à un modèle ultérieur de navigateur permet de comprendre l’importance des interfaces utilisateur, mais également leurs évolutions.

II.A. DÉFINITION ET PROBLÉMATISATION DE L’OBJET NUMÉRIQUE 83

pleinement qu’en le pensant comme point de correspondance entre des matérialités concrètes (l’ordinateur de l’utilisateur, les serveurs des sites, etc), dont le navigateur est intégralement dépendant en dernier recours, et des abstractions standardisées (le web comme creuset de ressources HTML, les logiciels comme constructions via du code informatique). Les objets numériques sont des objets hybrides, et ne peuvent être compris profondément que comme intersections entre deux réalités d’ordres différents. Ainsi, en anglais, le terme « browser » signifie littéralement « feuilleteur », ou « butineur »; à la navigation sur des eaux homogènes et toujours présentes, se substitue l’idée de la recherche, de la démarche volontaire; ce terme est plus juste, car il permet l’image de la recherche de ressources vers et via des entités matérielles (les câbles d’internet, les serveurs) situées aux quatre coins du monde.

Le navigateur est donc un logiciel permettant d’interpréter ce que lui renvoient les serveurs pour le rendre intelligible à l’utilisateur (et réciproquement : il permet de rendre intelligibles aux serveurs les demandes de l’utilisateur), dans le cadre constitué par les standards du Web. Ce qui apparaît au premier abord comme une simple interface pour interagir avec des pages web sous-tend une technicité très dense.

Ajoutons à notre définition fonctionnelle une caractérisation relevant du mode de production du navigateur<sup>30</sup>. Le logiciel peut être construit selon deux modalités différentes, qui sont en réalité la plupart du temps com-

---

30. Nous rentrerons dans les détails et les implications d’une telle différenciation en troisième partie. Nous la notifions ici pour caractériser notre objet d’étude de façon complète dès le départ.

plémentaires<sup>31</sup> : le logiciel dit « libre » et le logiciel dit « propriétaire »<sup>32</sup>. Ces deux tendances sont en réalité deux extrêmes se situant d’un bout à l’autre d’un curseur qui déploie des nuances vers l’ouverture ou la fermeture. La détermination du mode de production se fait selon l’accès ou non au code source du logiciel. Si le code source n’est pas accessible à tous, alors le mode de production est celui du logiciel fermé ; seuls les salariés de l’entreprise peuvent le connaître et le modifier, et ils sont tenus au secret. Le logiciel fermé est une sorte de boîte noire. On se situe encore, quant au mode de production, dans une configuration presque industrielle. Si l’intégralité du code source est directement accessible en ligne, c’est que le logiciel est libre<sup>33</sup>.

Il est donc primordial de faire rentrer dans la définition de notre objet d’étude la tension inhérente à son mode de production, qui oscille entre l’ouvert et le fermé. De fait, notre objet d’étude n’est pas « le navigateur » dans son entièreté, mais bien un navigateur qui revêt un mode de fonctionnement ouvert, par opposition au fonctionnement fermé - le navigateur libre Mozilla Firefox<sup>34</sup>.

31. Ici encore, formaliser les choses en terme dualistes permet de faciliter l’explication, mais ne reflète pas exactement la réalité : aujourd’hui, la plupart des logiciels sont hybrides, c’est-à-dire à mi-chemin entre l’open source/libre et le logiciel propriétaire. Cela empêche de traiter le sujet de façon radicale si l’on veut être exact, et est au centre de préoccupations de plusieurs penseurs, qu’ils soient sociologues (S. Broca [30] ) ou philosophes (P. Hinamen[63]).

32. Voir glossaire, ”Logiciel (du point de vue de la production)”.

33. Quant au mode de production correspondant au logiciel libre, nous le qualifions dans la suite de l’argumentation de « post-industriel » : voir partie III.

34. Nous avons choisi de ne pas faire ici de présentation circonstanciée de ce navigateur : nous avons énuméré ses deux caractéristiques principales en tant que membre de la communauté des navigateurs

## II.A. DÉFINITION ET PROBLÉMATISATION DE L’OBJET NUMÉRIQUE 85

D’autres caractéristiques sont fondamentales pour comprendre notre objet d’étude : la complexité, la rapidité de son évolution, la versatilité des standards. Nous tenterons d’intégrer ces caractéristiques, liées aux problématiques du numérique en général, dans le cadre de notre comparaison avec la théorie simondonienne<sup>35</sup>. Le logiciel est à la fois une carrière à ciel ouvert selon des modalités d’échange réticulées (en particulier lorsqu’il est libre : des personnes travaillent sans cesse pour l’améliorer) et un objet fini (la version fixe dont l’utilisateur se sert).

### **Le navigateur comme objet numérique nodal**

Jusqu’ici, nous avons tenté de montrer que le navigateur était un objet numérique cohérent : une unité fonctionnelle au mode de production ouvert ou fermé. Il nous faut maintenant appuyer l’importance de cet objet parmi la communauté des objets numériques<sup>36</sup>.

(unité fonctionnelle et mode de production ouvert), ce qui est suffisant pour l’argumentation.

35. Elles seront traitées en III.A sous la problématique commune de la commensurabilité de la technique.

36. Nous avons conscience que notre démarche diffère, ici encore, légèrement de celle de Simondon. En effet, ce dernier ne se justifie quasiment jamais quant à la sélection des exemples techniques. Un tel choix est pourtant mûrement réfléchi à l’aune du critère de représentativité desdits objets. Leur pertinence est garantie par la vaste culture technique de l’auteur, qui se situe par rapport à elle sans la mettre en avant directement. Notre démarche ne diverge donc pas fondamentalement de la démarche simondonienne sur le fond ; elle s’astreint simplement à un travail de justification et de mise en contexte plus précis (ce qui est permis par le fait de travailler sur un objet unique).

Le navigateur dialogue avec une série d’objets à la fois matériels (les ordinateurs, les serveurs) et numériques (le web, les autres logiciels). C’est un logiciel éminemment complexe qui acquiert de plus en plus d’usages selon une logique de compaction<sup>37</sup>. Ce phénomène participe d’un double mouvement : plus le navigateur est capable de supporter de fonctionnalités, plus les usages pour lesquels on le sollicite s’agrandissent. La compaction est donc un mouvement d’extension des fonctionnalités du navigateur accompagné d’un mouvement symétrique d’extension des usages. Plus les possibilités du navigateur grandissent, plus les possibilités d’usages augmentent.<sup>38</sup> Le navigateur est donc un objet numérique en constante expansion.

De plus, le navigateur, on a l’a dit, est l’interface de contact entre l’utilisateur et les objets numériques du Web. En cela, il cristallise bon nombre d’interactions psychosociales ; il est également au centre de grandes batailles économiques ; l’enjeu de pouvoir est devenu le navigateur et non plus le système d’exploitation<sup>39</sup>. Le navigateur est au fait d’une réelle effervescence autour de ses structures techniques depuis son apparition, et évolue de fait à une rapidité extrême.

L’intérêt de l’étude du navigateur tient donc à son fonctionnement technique en synergie avec d’autres objets numériques, qui en fait un objet nodal ; à son exposition particulière aux investissements psychosociaux,

37. François Tainai, entretien, 1er mars 2017.

38. Les objets connectés sont un bon exemple d’un tel phénomène. Avec ces derniers, le navigateur doit s’adapter entre autres aux fonctions de géolocalisation, qui étaient inexistantes auparavant. François Tainai, entretien, 1er mars 2017.

39. Ainsi, ”Sous beaucoup d’aspects, le navigateur est en train de devenir le nouveau système d’exploitation.”, D. Odell, *op. cit.*

II.A. DÉFINITION ET PROBLÉMATISATION DE L’OBJET NUMÉRIQUE 87

et aux intérêts économiques. C’est un objet numérique doublement intéressant : du point de vue de l’objet, intérêt du fonctionnement technique complexe et ramifié ; du point de vue de ses relations avec les usagers, intérêt de la charge des investissements (émotionnels, politiques, économiques) psychosociaux qu’il charrie.

### Précision sur le niveau de l’étude

Simondon place son analyse de l’objet technique à un niveau générique. En d’autres termes, ce dernier prend pour exemple « le tube électronique »<sup>40</sup>, et non le tube électronique d’une marque précise. Il part du principe que le principe de production de l’objet est le même dans toutes les firmes. Ici apparaît la première difficulté pour notre comparaison car, comme on l’a vu, les navigateurs sont des logiciels qui n’ont pas tous le même mode de production. Cela implique un choix de l’objet numérique selon la sous-catégorie du mode de production, ouvert et fermé : nous nous situons donc à un niveau de particularité supplémentaire par rapport au travail de Simondon.

A ce stade de la réflexion, on peut dire du navigateur qu’il est un objet numérique à l’unité fonctionnelle bien définie d’une part, et que cet objet est central dans la communauté des objets numériques en général. Son analyse en tant qu’objet cohérent et important, représentant de façon adéquate une sous-catégorie de la grande classe des logiciels, semble donc justifiée<sup>41</sup>. Pour les besoins de notre étude, nous oscillerons entre l’analyse du

40. MEOT, p.28.

41. Il est par ailleurs notable que très peu d’études aient été menées dans le domaine informatique sur le fonctionnement des navigateurs en général. Nous sommes ici grandement redevables à

logiciel dans sa généralité et l'exemplification par l'étude de certains points techniques du navigateur libre.

## II.B Construire le logiciel à partir de la marge d'indétermination

*« La pensée philosophique ne pourra bien saisir le sens du couplage de la machine et de l'homme que si elle arrive à élucider le véritable rapport qui existe entre forme et information. »*

*– Simondon, Le mode d'existence des objets techniques, p.137.*

Au terme de ce premier parcours, une difficulté majeure survient. La qualification « d'objet technique » est-elle réellement légitime ? Elle l'est selon une certaine définition de l'objet technique : une cohérence, une unité fonctionnelle. Elle ne l'est pas si l'on considère que ce qui définit un objet technique est sa matérialité. Là se situe le point majeur de notre travail : peut-on, sans considérations abusives, penser le logiciel comme un objet technique ? Quel pourrait être son fonctionnement vis-à-vis de la théorie simondonienne ?

Il s'agit maintenant de réussir à trouver, dans les analyses de Simondon, une porte d'entrée permettant de penser un objet immatériel. Nous partirons pour ce faire de la machine à marge d'indétermination. Nous verrons

---

l'article de T. Garsiel, « Understanding how browsers work »[56], ainsi qu'aux nombreuses conversations menées avec des programmeur·euse·s.

qu’une telle machine fournit de solides bases conceptuelles pour penser le logiciel – un objet technique radicalisant la définition fonctionnaliste de l’objet technique.

## II.B.1 La machine-ordinateur et la marge d’indétermination

### L’ordinateur, individu technique

Le point de départ pour la comparaison entre l’objet numérique qu’est le logiciel et la théorie simondonienne se situe dans la machine-ordinateur (la « machine à calculer » de Simondon). On peut affirmer sans trop de difficultés que l’ordinateur est un individu technique, et prendre comme point de départ cette correspondance assurée. Nous en ferons une démonstration rapide (car là n’est pas le sujet de notre travail) afin de pouvoir prolonger notre démonstration<sup>42</sup>. Simondon démontre à plusieurs endroits sa connaissance de la machine-ordinateur telle qu’elle existait à son époque, et semble la considérer comme un individu technique à part entière.

Ainsi, il est possible de faire correspondre point par point les différents critères de la philosophie de la technique de Simondon à l’ordinateur.

---

42. Cette mise en parallèle peut probablement prêter à des objections ; si ces dernières sont mineures (cas particuliers) et ne remettent pas en cause l’ensemble du raisonnement, nous les prendrons en compte sans pour autant modifier l’ensemble ; si ces dernières sont majeures, nous en prendrons acte et modifierons le contenu de ce travail.

Tout d’abord, la structure fondamentale des ordinateurs contemporains reste la même : ils sont tous constitués d’une mémoire et fonctionnent en synergie avec un ensemble de périphériques. On peut considérer, selon cette assertion, que l’objet technique « ordinateur » est bien soumis à une genèse, dont la concrétisation se manifeste en partie par la miniaturisation et la mise progressive en réseau.

L’ordinateur, son aspect matériel, déploie de façon cohérente des niveaux de technicités différenciés.

- Les éléments de l’ordinateur sont les cartes de mémoire vive ainsi que les différents composants qui renferment en eux-mêmes une technicité propre, pouvant assurer de plus en plus de fonctionnalité au fur et à mesure de leur développement.
- L’individu est l’ordinateur en lui-même. Il s’agit d’une machine cohérente qui fait fonctionner les éléments entre eux.
- L’ensemble créé par une communauté d’individus techniques est par exemple, sur un bureau de travail, l’ensemble des périphériques reliés à l’individu ordinateur, qui fonctionnent en synergie avec lui via l’intermédiaire du régulateur humain : l’imprimante, les enceintes, la webcam, etc.
- L’ordinateur contemporain remplit même le quatrième degré de technicité identifié par L. Duhem [46] et mis en valeur par Simondon dans la PST, qui est celui de la réticulation des individus techniques. Avec Internet, l’objet technique ordinateur est relié à ses congénères via un réseau de technicité devenu mondial.

Enfin, l’ordinateur en tant qu’objet technique et matériel entretient des relations avec le système social dans

lequel il s’insère, relations ambivalentes de culte et d’incompréhension de par sa complexité croissante, qui conduisent à des formes d’aliénation. On peut aisément faire correspondre les notions de « halo de technicité » ou de « technophanie » à des attitudes psychosociales envers nos ordinateurs contemporains<sup>43</sup>.

L’ordinateur peut assurément, dans sa matérialité, être reconnu comme un objet technique à part entière, en correspondance avec les critères simondoniens. Mais une telle mise en parallèle serait largement insuffisante, car les ordinateurs ne sont pas uniquement des individus soumis à genèse et comportant plusieurs niveaux de technicité. Ils sont avant tout définis par leur marge d’indétermination, qui est un concept cardinal de la philosophie de la technique de Simondon.

### L’ordinateur, machine à marge d’indétermination

*« Le véritable perfectionnement des machines, celui dont on peut dire qu’il élève le degré de technicité, correspond non pas à un accroissement de l’automatisme, mais au contraire au fait que le fonctionnement d’une machine recèle une certaine marge d’indétermination. »*

*– Simondon, Le mode d’existence des objets techniques, Introduction.*

Le concept de marge d’indétermination apparaît à

---

43. Difficile de ne pas prendre ici l’exemple de l’attraction irrésistible provoquée par les ordinateurs de la marque Apple, avec leur design travaillé, dont le fonctionnement interne reste pourtant inconnu de la plupart des utilisateurs ; l’intérieur des machines est par ailleurs assez peu accessible, ces dernières étant closes.

deux endroits du MEOT. Dès l’introduction, il est la pierre angulaire de l’argumentation ; il est ensuite employé de nouveau dans le cadre d’un commentaire plus large de la cybernétique de Wiener<sup>44</sup> et de sa théorie de l’information.

La marge d’indétermination intervient pour résoudre un problème de mécompréhension ontologique de la machine : lorsque l’on se pose la question de la technique, « c’est la notion de machine qui est déjà faussée<sup>45</sup> » ; on assimile la machine à un automate. On restreint l’idée de machine à un bloc cohérent mais clos, à une « monade »<sup>46</sup>. L’enjeu du concept de marge d’indétermination est d’ouvrir la machine à un champ plus large de possibilités – de « mettre une fenêtre » à la monade. Cela conduit à une redéfinition ontologique profonde de la machine ouverte. Cette redéfinition a des conséquences considérables : techniquement, elle fait office de rupture dans l’ordre machinique ; de façon corrélative, au niveau épistémologique, elle entraîne la nécessité absolue pour la philosophie de prendre acte d’une telle rupture et de repenser les machines à de nouveaux frais, à partir de nouveaux schèmes.

La formation de la marge d’indétermination s’observe dans le dialogue entre l’introduction du MEOT, où elle apparaît comme déterminante et structurante mais peu approfondie, et le chapitre II et III de la seconde partie,

---

44. « La seconde partie [est] en quelque sorte un commentaire dans le domaine de la mécanologie, sur l’étude de Wiener L’Usage humain des êtres humains. », J. Hart, Préface du MEOT.

45. MEOT, p.146.

46. L’automate est la machine qui « possède en elle la formule complète de son activité » (« L’invention et le développement des techniques » [89] ).

où elle est très précisément définie.

La marge d’indétermination est un geste de refonte ontologique. Il s’agit de l’élargissement de la notion de machine, prise en un sens trop restreint – celui de l’automate, aux possibilités déterminées, à un sens plus large – celui de la machine dite « ouverte », aux possibilités plurielles. Le concept est donc défini de façon strictement dialectique : il est en opposition constante avec celui d’automate, tenant injuste du statut de « machine » aux yeux de la culture. Tandis que l’automate est une machine « fermée », la machine à marge d’indétermination est définie comme « ouverte<sup>47</sup> » ; au déterminisme de la première, lié à ses fonctionnalités restreintes, correspond la (relative) indétermination de la seconde, dont les fonctionnalités sont démultipliées. Tandis que l’automate, la machine fermée, est un individu stable, prévu pour un fonctionnement prédéfini et, en ce sens, « parfait », la machine ouverte témoigne d’un équilibre métastable : elle présente un certain nombre de « points critiques<sup>48</sup> » qui nécessitent une détermination extérieure pour l’orienter.

On a donc une ouverture du champ de la machine à de plus larges possibilités : « la perfection » dont on investit trop rapidement les machines n’est pas ce qui fait la perfection de leur technicité : « La machine est enfermée dans cette vision réductrice qui la considère comme achevée en elle-même et parfaite, qui la fait coïncider avec son état actuel, avec ses déterminations matérielles. »<sup>49</sup>

Une fois cette première détermination posée, un problème se pose. L’automate est parfait – en ce sens qu’il

47. MEOT, p.12 : “La plus grande perfection coïncide avec la plus grande ouverture.”

48. MEOT, p.141.

49. MEOT, p.146.

est une monade sans fenêtre ; une fois inventé, il se suffit à lui-même. La machine ouverte est imparfaite : elle ne peut fonctionner sans un agent pour la déterminer de l’extérieur, dans une relation de collaboration ouverte. Ouvrir le concept de machine à un champ de possibilités plus large implique de la penser dans une collaboration avec l’humain – selon le schème de la « transductivité ». Tandis qu’au contact d’automates l’homme est le « surveillant d’une troupe d’esclaves », il est ici un « organisateur permanent » et un « interprète »<sup>50</sup> de la « communauté » des machines ouvertes. Ce champ métaphorique déployé en introduction fait écho à la volonté de collaboration égale entre l’homme et les machines<sup>51</sup>. Simondon convoque l’image du « chef d’orchestre », qui doit tout autant savoir jouer que diriger. L’image sous-tend une relation de codépendance : les musiciens ne produiraient pas cette entité transcendante qu’est par exemple une symphonie sans quelqu’un pour les synchroniser entre eux ; inversement, le chef d’orchestre ne pourrait produire cela seul, sans « matière » musicale. La complémentarité de l’homme et la machine, comme « associé[s] »<sup>52</sup>, crée une perfection transcendante, qui ne peut être attribuée indépendamment à aucune des deux entités ; c’est la collaboration féconde de ces dernières qui transforme une imperfection première en une perfection supérieure. Ainsi, « il y a couplage interindividuel entre l’homme et la machine lorsque les mêmes fonctions auto-régulatrices sont accomplies mieux et plus finement par le couple homme-machine que par l’homme seul ou la

---

50. MEOT, p.11.

51. Cette volonté de collaboration fait écho aux problématiques soulevées en I-a des représentants de la technicité.

52. MEOT, p.125.

machine seule. »<sup>53</sup>

L’homme a ainsi une double fonction : il restreint la marge d’indétermination individuelle de chaque machine et « assure la fonction d’intégration, et prolonge l’auto-régulation en dehors de chaque monade d’automatisme par l’interconnexion et l’intercommutation des « monades ». <sup>54</sup> La machine ouverte, pour intégrer la notion de possibilité, fait entrer le facteur humain comme nécessaire dans son fonctionnement. La marge d’indétermination suppose la « synergie fonctionnelle », couplage qui intervient « lorsqu’une fonctionne unique et complète est remplie par deux êtres »<sup>55</sup>. C’est l’association à parts égales des apports de la machine et de l’homme qui permet de créer une fonction supérieure et éminemment plus perfectible que celle des automates.

Ce couplage fonctionnel est encore repris et précisé dans le cadre de la discussion sous-jacente avec la cybernétique qu’entretient Simondon au chapitre III de la partie II du MEOT. La notion de marge d’indétermination est également ce qui fait l’originalité de la pensée de Simondon quant à la théorie de l’information. La restriction de la marge d’indétermination est comprise comme de l’information – au sens étymologique du terme (in-former, remplir une forme). Cette marge d’indétermination n’est pas infinie (la machine n’est pas ouverte à toutes les possibilités) ; elle est elle-même prédéterminée par des « formes rigides ». C’est également l’homme qui réduit les possibilités du passage de l’information : il a un double rôle d’interprète et d’inventeur ; « L’homme, interprète des machines, est aussi celui qui,

53. MEOT, *ibid.*

54. MEOT, p.126.

55. MEOT, p.124.

à partir de ses schèmes, a fondé les formes rigides qui permettent à la machine de fonctionner. »<sup>56</sup>. Simondon insiste sur cette séparation de la « forme » d’une part, et de « l’information » de l’autre : la forme est ce qui a été cristallisé et déterminé par avance par l’homme, c’est le champ des possibles permis par la marge d’indétermination inscrit dans une structure technique. Elle n’est pas une information mais une « condition d’information ».<sup>57</sup> L’information est ce qui vient remplir, d’une manière ou d’une autre, cette forme.

Suivant cette théorie, la forme d’un ordinateur est la logique binaire implémentée de manière physique, on l’a vu, dans des circuits ouverts ou fermés selon la logique booléenne. La forme de circuits pouvant être ouverts ou fermés selon la présence ou non d’un courant électrique est un « a priori » qui va « accueillir de l’information ». L’information, dans ce cadre, est le code que l’on transmet vers cette logique binaire, traduit en binaire par le compilateur : des 0 (fermeture du circuit) ou des 1 (ouverture du circuit). Le code informatique est donc de l’information, au sens fort du terme : il est ce qui remplit la forme prédéterminée, traduite en schème physiques, de la marge d’indétermination d’une machine ouverte<sup>58</sup>. Le code informatique est la transduction des schèmes mentaux de l’homme (logique binaire, langages de program-

56. MEOT, p.138.

57. MEOT, p.137.

58. La différence avec les exemples analogiques employés par Simondon est qu’on peut supposer que le hasard est plus restreint, puisque l’information avec le numérique peut prendre un nombre de valeurs finies (1 ou 0) et non infinie (comme c’est le cas avec l’analogique et les exemples que traite Simondon). Cette affirmation est pourtant à nuancer, car il est tout de même possible d’opérer une quasi-infinité de combinaisons avec ces 1 et ces 0.

mation) au régime technique.

Ainsi, la forme binaire prédéterminée de l’ordinateur a « une fonction de sélectivité »; elle est ce qui permet à l’information d’être « à mi-chemin entre le hasard pur et la régularité absolue ». <sup>59</sup> On peut alors préciser la notion d’information qui vient restreindre la marge d’indétermination : il ne s’agit pas d’un hasard absolu pouvant prendre toutes les formes possibles (« la pure imprévisibilité de toute variation ») mais bien « l’imprévisibilité d’une variation de forme » <sup>60</sup>.

On le voit, la marge d’indétermination fait l’objet d’un travail de conceptualisation de plus en plus approfondi au fil de l’argumentation de Simondon. Elle participe bien d’une refonte ontologique du concept de machine, en réponse à la restriction de ce dernier au simple automate. Simondon ouvre un champ de possibilités à la machine ouverte – sans que cela ne soit pour autant « hasard pur ». La marge d’indétermination est ce qui est ouvert au possible selon un cadre formel (prédéterminé par des schèmes humains) : la « forme ». Ce cadre est rempli par de l’information. L’homme est celui qui pense le cadre formel d’ouverture des machines, et qui le remplit ensuite ; il est inventeur et informateur. Il doit informer et mettre « relation d’interconnexion » la communauté des machines. Cette relation est réversible : si l’homme collabore à parts égales avec elle, c’est bien que « sa vie est faite du rythme des machines qui l’entourent et qu’il relie les unes aux autres. » <sup>61</sup>

La marge d’indétermination a donc une épaisseur ontologique et psychosociale tout à la fois. Elle introduit

59. MEOT, p.137.

60. MEOT, *ibid.*

61. MEOT p.126.

une rupture dans l'ordre machinique. La machine ouverte, dans sa collaboration avec l'homme, est en mesure de produire des configurations techniques nouvelles et de réconcilier la technique avec la culture.

Sans prendre pour paradigme cette machine à marge d'indétermination<sup>62</sup>, Simondon mesure l'effet de la rupture. Cela se traduit par plusieurs élans réflexifs qui ponctuent les développements sur la théorie de l'information. La prise en compte de cette dernière nécessite selon lui une « attitude d'esprit différente » de celle déployée vis-à-vis des processus thermodynamiques<sup>63</sup>. Il énonce des impératifs : le philosophe doit dépasser les paradigmes établis et chercher à penser autrement pour caractériser les problèmes que posent la marge d'indétermination : « La naissance d'une philosophie technique au niveau des ensembles n'est possible que par l'étude approfondie des régulations, c'est-à-dire de l'information. »<sup>64</sup>.

Il est aujourd'hui possible de mesurer l'ampleur de cette phrase, prononcée en 1958, mais qui fait directement écho à notre époque :

*« Il est donc naturel que l'intervention de l'usage des canaux d'information distincts des canaux énergétiques, dans les machines, ait apporté une modification très profonde de la philosophie des techniques. »*<sup>65</sup>

---

62. Dans le MEOT, Simondon analyse plutôt les machines produites de façon industrielle, dont la marge d'indétermination est faible.

63. MEOT, p.133.

64. MEOT, p.126.

65. MEOT, p.129.

## II.B.2 La complexification du code informatique

Simondon a principalement traité de machines analogiques pour analyser la marge d'indétermination. Il fait cependant mention de machines à fonctionnement binaire et de code : la notion de programmation intervient, bien qu'elle ne soit pas encore très développée.

Comment alors penser le code informatique à l'ère numérique ? La pensée simondonienne est-elle suffisante ? Au vu de ses théories concernant l'information, elle peut déjà donner à la définition du code informatique une épaisseur considérable. Il s'agit de l'information apportée à la forme de la marge d'indétermination. Cette information est une information logique (1 ou 0) traduite en information physique (circuit ouvert ou fermé). Le code informatique est donc de l'information binaire destinée à informer physiquement un schème technique inventé au demeurant par l'homme – celui des circuits ouverts ou fermés (commutation de circuits). C'est un cas exemplaire de transduction d'une réalité (psychique, celle de l'invention) à une autre (technique, celle de la machine). Le code est « condition d'actualisation »<sup>66</sup> des schèmes techniques de l'ordinateur. Il est ce qui restreint sa marge d'indétermination.

On peut affiner cette définition du code informatique. Simondon parle déjà de « programmation » via ce qu'il appelle des « schèmes de décision » : « avant de faire fonctionner la machine, il faut la programmer. »<sup>67</sup> Ce-

---

66. MEOT, p.143.

67. MEOT, p.142.

pendant, les exemples analogiques choisis et les ordinateurs contemporains à Simondon ne lui ont pas permis de penser la complexification de la transmission de l'information en véritable langage. Simondon ne pense en aucun cas le *software* en opposition au *hardware*, car à son époque, la question ne se posait pas en ces termes ; le *hardware* était prédominant et le logiciel n'était pas considéré à part de celui-ci<sup>68</sup>. Aujourd'hui, le code ménagé par la théorie de l'information simondonienne n'est plus seulement ce qui permet de combler la marge d'information. Un degré de complexité a été passé depuis que le codage des machines est devenu l'objet d'une architecture complexe. On parle en effet « d'architecture logicielle »<sup>69</sup>. Ce qui, *in fine*, va être transformé en langage binaire pour informer un cadre formel simple se complexifie au fur et à mesure de la montée en abstraction ; les instructions sont rédigées en langage compréhensible par l'humain et traduites par un programme, le compilateur ou l'interpréteur, pour informer de façon binaire la machine physique. Ces constructions de langage, dirigées vers l'information et la traduction en langage machine, sont appelées langages de programmation. Plusieurs langages de programmation peuvent être employés pour constituer un unique code source. Ce que l'on appelle le « code informatique » a acquis, vis-à-vis des schèmes pensés par Simondon, une réalité propre dont il convient de penser les relations avec le versant machinique qui la fait émerger.

Le vecteur de ce passage à l'abstraction, ce qui permet l'existence d'objets immatériels, est la transforma-

68. La dissociation du *hardware* et du *software* est actée par IBM et signe le début de la privatisation des logiciels (voir III.B.3).

69. Il s'agit d'un domaine de l'informatique à part entière.

tion des machines concrètes qui les supportent : grâce au développement des périphériques de l’ordinateur et aux réflexions sur l’interface utilisateur, un objet s’érige à l’écran. C’est l’arrivée de ce dernier, ainsi que de la souris et le clavier, qui permet de constituer l’objet logiciel en tant que construction architecturale sur laquelle l’homme peut avoir une influence<sup>70</sup>. Le logiciel devient objet grâce à l’interface graphique qui lui donne une consistance visuelle. De tels périphériques étaient très peu répandus à l’époque de la rédaction du MEOT : « Il faut une complication considérable pour obtenir qu’une machine à calculer puisse écrire sur l’écran du tube cathodique les résultats en chiffre directement lisibles.<sup>71</sup> »

Ainsi, le code informatique peut être défini, dans sa généralité, comme une abstraction informatique qui se traduit sous la forme d’un langage structurant et performatif.

Il est une abstraction au sens informatique du terme<sup>72</sup> :

---

70. Une réflexion de fond autour de ces périphériques est initiée par Douglas Engelbart. En 1968, alors chercheur à Standford, il présente à un public d’informaticien·ne·s ses travaux sur l’interface homme-machine. La démonstration qu’il donne pose les bases des éléments aujourd’hui standards en informatique personnelle : fenêtrage, bureau, liens hypertextes. Ces inventions seront reprises et répandues une quinzaine d’années plus tard. [71] Deux sources permettent d’approfondir les avancées techniques de Douglas Engelbart : une source primaire, D. Engelbart “Augmenting human intellect : a conceptual framework” [48], et une source secondaire, T. Bardini, *Bootstrapping : Douglas EngelBart, Coevolution, and the Origins of Personal Computing (Writing Science)* [5].

71. MEOT, p.121.

72. Et non dans son sens courant, « a mental expression that seeks to separate in thought, and capture in some concise manner, the essence of some entity. » (Noam N. et S. Schocken, *op.cit.*)

« a statement of « what the entity does » and ignoring the details of « how it does it »<sup>73</sup>”. Il s’agit d’une définition fonctionnelle : « this fonctionnal description must capture all that needs to be known in order to use the entity’s services, and nothing more »<sup>74</sup>. Une abstraction est ce qui permet de traiter les différents composants informatiques de façon indépendante les uns des autres. Un ordinateur est recouvert de plusieurs couches d’abstractions (*layers*). Par exemple, tandis que le système d’exploitation communique directement avec le matériel informatique, le navigateur fait appel au système d’exploitation et ne correspond pas directement avec la matérialité de l’ordinateur (cela engendrerait des risques considérables pour la sécurité du matériel<sup>75</sup>). Un ordinateur est constitué de couches d’abstractions : « on nomme couche d’abstraction toute couche logicielle cachant au développeur l’implémentation de la couche inférieure, lui évitant de fournir une implémentation différente selon les spécificités de la couche inférieure. »<sup>76</sup> Ainsi, « All the the work, cleverness, information, and drama that went into the entity’s implementation are concealed from the client who is supposed to use it, since they are simply irrelevant »<sup>77</sup>.

Le code est avant tout une abstraction particulière qui fait correspondre à un langage ésotérique un langage mieux adapté à la compréhension humaine ; de même, il parvient à une forme de généralisation pour le détacher au maximum du contexte ; un même code pourra s’appliquer à différents systèmes d’exploitation.

---

73. Ibid.

74. Ibid, p.3.

75. Emmanuel St-James, entretien, 17 mars 2017.

76. Wikipédia France, article « couche d’abstraction » [1].

77. N. Noam et S. Schocken, *op.cit.*

Cette abstraction est un langage. Ici encore, les langages de programmation sont divers, mais présentent des caractéristiques communes. Ils comportent une syntaxe, qui est la description de ce qui est grammaticalement correct dans le langage ; des noms (variables, types, fonctions, paramètres), des types (un certain nombre de valeurs et d’opérations sur ces valeurs), et enfin une sémantique, c’est-à-dire la signification du programme<sup>78</sup>. Il s’agit d’un langage construit à partir d’une grammaire formelle et de règles sémantiques.

Ce langage est structurant, car il fait office d’architecture, de plan d’ensemble : les lignes de code constituant un logiciel permettent d’en montrer le plan de fonctionnement, et de répartir les étapes de sa mise en marche. De même, le code source d’un individu numérique montre à la fois la répartition de ses composants et toutes les démarches que la machine doit exécuter, dans l’ordre, pour parvenir à une tâche définie.

Ainsi, le code informatique a un aspect performatif ; il est de la « parole en actes »<sup>79</sup> ; les « plans », une fois traduits ou interprétés<sup>80</sup>, deviennent de l’information sous la forme de 1 et de 0 puis d’impulsions électriques. Il est un niveau d’abstraction dirigé vers l’implémentation dans la mémoire de l’ordinateur.

Le code informatique est une architecture composée de lignes d’instructions exécutables matériellement.

Il nous faut maintenant questionner le lien entre les constructions langagières complexes du code et la maté-

78. Cette définition provient de M. Grabbrielli et S. Martini, *Programming Languages : Principles and paradigms*. [55]. Voir glossaire, “Code informatique, code source”.

79. Emmanuel St-James, entretien, 17 mars 2017.

80. Voir Annexes, FIG – A.

rialité de la machine « ouverte » qui a permis l’émergence de ces architectures. Cela revient à se positionner vis-à-vis du mode d’existence du logiciel selon Simondon.

### II.B.3 Trois hypothèses concernant le statut du logiciel

Quel est le statut des architectures logicielles ? Nous envisageons trois hypothèses pour comprendre le logiciel à l’aune de Simondon.

- Le paradigme de la machine ouverte à marge d’indétermination est suffisant pour comprendre le logiciel. Dans ce cas, ce dernier peut être défini comme simple vecteur d’information pour la machine ouverte ; alors la machine ouverte théorisée par Simondon n’aurait aucun intérêt à être confrontée au logiciel, puisqu’elle le contient déjà tout entier.
- Le paradigme de la machine ouverte ne suffit pas pour comprendre le logiciel, et ce dernier peut être compris comme entité parfaitement indépendante (avec le terminal de sortie, l’écran), dans un nouveau régime (l’abstraction informatique). Il s’agirait alors de l’apparition d’un nouveau régime de technicité indépendant du premier, avec un mode d’existence différent.
- La machine ouverte est le support du logiciel, et ces deux entités sont liées selon des liens complexes. Nous avons alors affaire à deux régimes différents : celui de la matérialité du *hardware*, qui supporte le

régime d’abstraction informatique nous apparaissant via l’écran, et qui est l’architecture logicielle. Il s’agirait alors d’une nouvelle couche de technicité se superposant à la première, dépendante d’elle, et fonctionnant en lien étroit avec cette dernière. Cela élargirait donc le mode d’existence de la machine « ouverte » théorisée par Simondon dans le MEOT par superposition.

Nous penchons pour la troisième hypothèse, qui permet de considérer de façon indépendante l’architecture du logiciel pour l’analyser, tout en gardant en tête (pour produire une définition entière et cohérente) que ce dernier est constitué d’information, et qu’il fonctionne en symbiose avec l’appareil matériel qui le supporte. Nous sommes donc devant un cas où le codage de la marge d’indétermination s’est à ce point complexifié que le langage utilisé se transforme en ce qui semble être des architectures techniques d’un genre nouveau. Nous nous proposons d’analyser les différents niveaux de notre exemple, le navigateur, afin de voir si notre théorie semble pertinente.

## II.C Les niveaux de technicité logiciels

Récapitulons pour ce faire les différents critères qui caractérisent et définissent un objet technique chez Simondon.

- Un premier postulat fort : les objets techniques sont investis d’un mode d’existence en propre. Ce mode

d’existence se caractérise avant tout par leur technicité, c’est-à-dire par leur fonctionnement.

- Les objets techniques sont sujets à une *genèse* ; en cela, ils descendent d’une lignée phylogénétique ;
- Il y a des phases de technicité (l’élément, l’individu, l’ensemble) qui se déploient selon deux modes différents :
  - Un mode scalaire, synchronique (à l’intérieur de l’objet-même, relations de réticulation)
  - Un mode diachronique (à notre époque, la phase de technicité prépondérante est l’ensemble, selon le MEOT, et l’élément réticulé, selon la PST)

Le logiciel serait donc le produit de l’excroissance de l’information et de sa gestion (grâce à la sortie qu’est l’écran) via le code informatique comme architecture performative. Quel est alors son mode d’être, ses évolutions ? Peut-on dire de lui qu’il s’agit d’un objet technique, d’une nouvelle forme de technicité exclusivement fonctionnelle ?

Les affirmations que nous tenons ici ont encore la forme d’hypothèse, de propositions, mais il serait intéressant de fluidifier les notions d’échelle de technicité au vu de la pensée d’un nouvel objet technique numérique. Ces niveaux de technicité, même s’ils semblent correspondre à ceux de la théorie établie par Simondon, subissent nécessairement des changements causés par le format de l’individu technique. En effet, l’aspect immatériel permet une évolution et une versatilité bien plus rapide que dans le cadre d’un objet matériel. Il serait probablement souhaitable de réfléchir à une méthode – à des critères – légèrement différents pour envisager la technicité du logiciel. L’essai de correspondance proposé

ici n’assume pas tous les aspects du fonctionnement d’un objet technique d’un genre nouveau. Il veut seulement poser des pistes de correspondance et permettre de poursuivre l’argumentation (que nous souhaitons mener un peu plus loin sans trop nous attarder sur ce point, qui par ailleurs mériterait une attention plus grande). Nous nous intéresserons principalement à la genèse du navigateur, et à deux niveaux de technicité de celui-ci : celui de l’élément et celui de l’individu, dans son dialogue avec le milieu associé.

### II.C.1 La genèse du navigateur

Si l’on se place au niveau de généralité propre aux analyses simondoniennes, on peut affirmer, de façon très probable, que le navigateur en tant qu’objet recelant d’une unité fonctionnelle précise est soumis à genèse. Cela est particulièrement visible au niveau de l’évolution de son interface utilisateur ; ce que l’on nomme « navigateur » est en fait le résultat d’une réelle lignée technique, que l’on peut faire remonter jusqu’au premier navigateur établi par Tim Berners-Lee à la fin de l’année 1990, le « WorldWideWeb » ; ce navigateur était le premier éditeur HTML, avant même l’existence du Web. Il permettait aux membres de la communauté de chercheurs d’accéder rapidement à des documents via un systèmes d’hyperliens. Ce dernier était d’abord extrêmement rudimentaire.

Une fonctionnalité représentative de la concrétisation progressive des navigateurs (formant en cela une lignée cohérente) est celle de l’insertion des images dans la même fenêtre de navigation. Le navigateur WorldWideWeb ne pouvait effectuer cette fonctionnalité, et affi-

chait l’image dans une nouvelle fenêtre. En 1993, NCSA-Mosaic est le premier à afficher des images dans la même fenêtre de navigation. De même, si l’on compare les capacités techniques de notre version de Mozilla Firefox et celles du WorldWideWeb de Tim Berners-Lee, le processus de concrétisation est indéniable. Le WorldWideWeb envoie une unique requête HTTP pour afficher une page de texte HTML, dont la présentation est très minimale (elle ne contient que du texte et des hyperliens). Le navigateur Mozilla Firefox, lui, envoie de multiples requêtes HTTP pour présenter une page HTML avec du contenu intégré : images, vidéos, etc. Par exemple, Firefox envoie soixante-dix requêtes pour ouvrir le site [www.philosophie.ens.fr](http://www.philosophie.ens.fr)<sup>81</sup> ; les fichiers affichés à l’écran restent agencés autour de la base du fichier HTML du site, mais s’agrémentent de multiples contenus multimédias qui rendent la page plus lisible<sup>82</sup>.

Par ailleurs, l’interface utilisateur est progressivement devenue plus fonctionnelle : insertion des boutons « avant » et « arrière », de la fonctionnalité « réactualiser », etc. De même, on peut voir dans les différentes « versions » d’un logiciel le processus d’évolution quasi-génétique décrit par Simondon<sup>83</sup>.

---

81. Test effectué le 11.06.17. Le nombre de requêtes envoyé par Mozilla Firefox pour afficher le site de l’ENS a été établi par l’utilisation d’un outil de Mozilla, « Network », qui trace les requêtes effectuées lors de la demande d’une page HTML.

82. Voir glossaire, “HTML/CSS”

83. On se place alors à un niveau de généralité moindre, puisqu’il s’agit là de l’étude d’un objet technique identifié, et non « du » navigateur – par exemple les évolutions de Mozilla Firefox. Mais cela représente tout de même de manière satisfaisante la lignée phylogénétique de ce dernier. Voir Annexes, Fig – D,E.

## II.C.2 L'élément, le milieu associé

### L'élément

L'élément, compris comme le plus petit niveau de technicité, peut pertinemment se trouver au niveau des fonctions. Une fonction est un regroupement insécable d'instructions pour l'ordinateur. Une fois la fonction définie par le-la codeur-euse<sup>84</sup>, elle est considérée comme une unité effectuant un certain nombre de tâches. La fonction peut être appelée à plusieurs reprises dans le code, lorsque les instructions qu'elle réalise permettent de fournir des rouages opportuns pour effectuer des opérations plus vastes, ayant un sens au niveau du logiciel dans son entier. L'exécution du code donne lieu à une mise en cascade d'appels à des fonctions, chacune effectuant des opérations, puis cédant la place à d'autres. Cette réaction en chaîne modifie l'état global du système et fait aboutir ce dernier à un état métastable, correspondant dans le cas du navigateur à l'affichage d'une page HTML.

Une telle mise en cascade ne suit pas toujours le même chemin. Dans le cas du navigateur, cela dépend –entre autres- de l'action entreprise par l'utilisateur. Si ce dernier décide de cliquer sur le bouton « retour » du navigateur, il y a de fortes chances que la page demandée soit restée en mémoire. Il faut donc la récupérer dans la mémoire, plutôt que de renvoyer une nouvelle requête HTTP. Par conséquent, cela justifie la décomposition du code en fonctions : la suite d'exécution du code n'est pas

<sup>84</sup>. Le regroupement initial des instructions qui composent la fonction présente une part d'arbitraire qui correspond à des choix ou à des conventions utilisées par le-la codeur-euse.

toujours identique et demande de la modularité<sup>85</sup>.

Précisons que les fonctions sont écrites dans une syntaxe propre à chaque langage de programmation. Elles sont contenues dans un fichier texte qui sert à les définir (c’est-à-dire à expliciter leur fonctionnement). Ce dernier regroupe généralement plusieurs fonctions en son sein, mais de tels regroupements sont des choix de confort de la part du-de la codeur-euse. Cela appuie notre définition de l’élément au niveau de la fonction et non de l’ensemble du fichier, dont le regroupement s’effectue dans une logique non principalement technique ; la fonction, au contraire, ne peut être subdivisée en sous-ensembles sans être déparée de son fonctionnement. Il s’agit du plus petit dénominateur de technicité numérique.

Un exemple de fonction peut se trouver en étudiant *Necko*, le moteur de réseau de Mozilla Firefox (responsable du *networking*). Ce dernier gère une requête HTTP en la mettant progressivement en place à l’aide de plusieurs fonctions (par exemple *AsyncOpen*, *BeginConnect*, *Connect*, *SetupTransaction*) provenant notamment du fichier intitulé

---

85. Cependant, suite à cette vision de la fonction comme agent atomique, il convient aussi d’ajouter que du point de vue du-de la codeur-euse, l’introduction de fonctions peut être simplement un outil pour rédiger de manière moins laborieuse le code. Elle permet notamment la factorisation de fragments du code : plutôt que d’écrire à plusieurs positions différentes les mêmes instructions, on les écrit à une seule reprise et chaque fois que l’on souhaite rédiger ces mêmes instructions, on a recours à cette fonction. Sans cet aspect de soucis de présentation, nous ne pourrions par exemple pas comprendre l’existence de fonctions auxiliaires, qui sont toujours appelées par la même fonction mère. Plutôt que d’intégrer une fonction auxiliaire dans l’écriture de la fonction mère, le-la codeur-euse choisit de fragmenter ces instructions dans la définition de deux fonctions différentes.

« nsHttpChannel.cpp ». Intéressons-nous à la fonction *SetupTransaction* : cette dernière s’étend de la ligne 933 à la ligne 1135 dans « nsHttpChannel.cpp ». Son but principal est de créer un objet représentant l’ensemble requête-réponse, c’est-à-dire la transaction HTTP. Pour initialiser cet objet, la fonction *SetupTransaction* déploie plusieurs actions. Elle se charge d’abord de mettre en forme la requête à l’aide d’informations fournies par des fonctions en amont. Puis elle spécifie la localisation de la réponse pour les fonctions postérieures, qui se chargeront de l’envoi effectif de la demande et de la réception de la réponse. Ainsi son point culminant réside dans l’initialisation de l’objet représentant la transaction, qui est définie sur les six lignes suivantes<sup>86</sup> :

```

1143     rv = mTransaction->Init(mCaps, mConnectionInfo, &mRequestHead,
1144                           mUploadStream, mReqContentLength,
1145                           mUploadStreamHasHeaders,
1146                           GetCurrentThreadEventTarget(), callbacks, this,
1147                           mTopLevelOuterContentWindowId,
1148                           getter_AddRefs(responseStream));

```

FIGURE II.1 – Capture d’écran des lignes codant l’initialisation d’un objet de la classe nsHttpTransaction dans le fichier nsHttpChannel.cpp

Deux des caractéristiques principales de l’élément pour Simondon sont son invisibilité – qui le préserve des « pollutions » psychosociales, et sa transductivité. L’élément

86. Pour voir les documents contenant les lignes de code en contexte, se référer à : [https://developer.mozilla.org/en-US/docs/Mozilla/Projects/Necko/Necko\\_walkthrough](https://developer.mozilla.org/en-US/docs/Mozilla/Projects/Necko/Necko_walkthrough) et <https://dxr.mozilla.org/mozilla-central/source/network/protocol/http/nsHttpChannel.cpp>

est ce qui contient le plus de technicité car, une fois concrétisé et auto-corrélé, il peut se déplacer d’un individu à un autre – voire engendrer de nouvelles lignées techniques. Les fonctions, considérées comme éléments, sont des « briques »<sup>87</sup> qui peuvent s’employer dans différents individus numériques, et effectuent un petit nombre déterminé de tâches techniques. Enfin, les éléments qui composent l’individu logiciel sont en communication étroite les uns avec les autres : ainsi, la fonction étudiée utilise les informations des fonctions qui se situent en amont dans la suite de tâches et informe celles en aval. Les éléments communiquent les uns avec les autres ; ceci est pour Simondon le signe de la concrétisation d’un objet technique<sup>88</sup>. On l’a vu, on peut considérer que les navigateurs web résultent de différents sous-ensembles<sup>89</sup>. Chacun de ces sous-ensembles jouent chacun un rôle particulier.

Au niveau de Firefox, l’élément se situe au niveau des fonctions qui composent un sous-ensemble plus large : par exemple, les éléments constituant le sous-ensemble plus large du *networking*.

Ainsi, les fonctions informatiques seraient un équivalent immatériel assez pertinent à l’élément théorisé par Simondon. Ils seraient ce qui transmet la technicité aux individus et aux ensembles via le processus de transduction. L’individu serait alors le logiciel dans son entier, compris comme unité fonctionnelle, et agençant les différentes bibliothèques.

87. Elie Michel, entretien, 28 février 2017.

88. Voir I.A.1

89. Voir Annexes, FIG – B, C. Ces sous-ensembles sont eux-mêmes constitués d’éléments plus petits ; Simondon ne pense pas précisément la hiérarchie d’emboîtement entre éléments de tailles différentes.

### L'individu et le milieu associé

Au niveau de l'individu, quel est alors le milieu associé? Pour un logiciel tel que le navigateur, il s'agit avant tout des autres constructions logicielles qui l'entourent – et du web. Le navigateur dialogue et réforme son fonctionnement interne principalement en fonction des interactions avec ce dernier. On peut voir ce phénomène d'adaptation au milieu associé en s'intéressant au fonctionnement du *networking* de Mozilla<sup>90</sup>. On l'a vu, le *networking* est un sous-ensemble dont la fonction est d'appeler et de réceptionner les pages HTML en dialoguant avec des serveurs distants. Le *networking* revêt une fonction de communication et de transfert. Communication d'abord : c'est lui qui établit le dialogue entre le navigateur (appelé « client » dans le processus) et le serveur contenant la page HTML demandée. Le processus se déroule ainsi : lorsqu'un utilisateur veut se rendre sur une page, par exemple `www.philosophie.ens.fr`, il rentre l'adresse HTTP du site web dans la barre de recherche de l'interface utilisateur prévue à cet effet. S'ensuit une réaction technique en cascade du *networking*, qui va ménager un milieu associé entre le navigateur et le web, "réservoir" de pages html contenant la page demandée. Le *networking* envoie une requête HTTP au serveur hébergeant le site de l'ENS. Cette requête HTTP contient la « fiche d'identité » de la demande, comprenant entre autres l'action à effectuer sur la page HTTP<sup>91</sup>, le nom de domaine

90. Voir Annexes, Fig – F.

91. Le plus souvent, il s'agit de la commande « GET », qui sert à afficher simplement la page dans le cadre de l'interface utilisateur du navigateur. Il est possible de modifier la page en elle-même (si l'action est autorisée par le serveur); la requête peut alors conte-

du serveur demandé (`www.philosophie.ens.fr`), des informations quant au navigateur qui envoie la demande (par exemple, un navigateur présentant la version 5.0 de Mozilla et tournant sous Mac), etc. Le serveur reçoit la demande et examine le type de réponse qu’il peut envoyer ; plusieurs codes de statut sont prévus à cet effet (succès, redirection, erreur client ou erreur serveur). S’il peut accéder à la requête, il renvoie un fichier de réponse HTTP contenant la page HTML demandée (c’est-à-dire des lignes de code HTML), ainsi que diverses métainformations le concernant (date et heure de renvoi du document, taille des fichiers envoyés...).

Ainsi, le web et ses protocoles est ce par quoi le navigateur ”se conditionne lui-même dans son fonctionnement”<sup>92</sup>. Le milieu associé formé par le navigateur et le web est un « milieu mixte »<sup>93</sup> entre ces différentes entités. Le navigateur, en communiquant avec les serveurs, applique son fonctionnement à un milieu technique présent d’un bout à l’autre du monde. C’est la création d’un milieu associé qui permet à l’individu navigateur d’être efficace et d’afficher les pages rapidement. Tout comme le moteur de traction de la locomotive rejette dans la ligne qui l’alimente « une réaction qui traduit [la] structure géographique et météorologique du monde »<sup>94</sup>, le networking informe le moteur de rendu de l’état des serveurs web et des pages HTML qu’ils contiennent – cet état influant sur le fonctionnement interne du navigateur.

Le logiciel semble se comporter comme un individu

---

nir, par exemple, la commande « DELETE » (pour supprimer des contenus), ou « PUT » (pour ajouter des contenus).

92. MEOT, p.56-57.

93. MEOT, p.54.

94. MEOT, p.53.

technique : il permet à première vue une analyse diachronique (genèse) et synchronique (niveaux de technicité). Cependant, il faut bien se garder de produire une équivalence stricte. Notre objet numérique est un objet immatériel : s’il déploie une concrétisation similaire aux objets classiques, c’est qu’il y a *transduction* du processus de concrétisation d’un régime d’existence (matériel) à un autre (immatériel) via le code informatique. L’hypothèse que nous émettons ici, et qui est centrale à notre travail, est que le concept de marge d’indétermination permet le développement d’une nouvelle couche de technicité immatérielle, qui se superpose à la première.

Si le logiciel peut ainsi être considéré comme un objet technique, c’est que la thèse fonctionnaliste de Simondon est validée a posteriori comme permettant de comprendre des formes de technicité qui nous contemporaines, ce que la définition aristotélicienne ou heideggérienne de la technique ne permettait pas. Le logiciel radicalise la thèse fonctionnaliste de Simondon *par l’exemple*. Il est un objet technique qui trouve son essence dans son fonctionnement (sa genèse et ses niveaux de technicité), et non premièrement dans la matérialité, puisqu’il est constitué d’information, de langage et de rendu graphique. Le logiciel est une forme d’objet technique épuré à l’extrême.

Cette couche de technicité peut, suite à un raisonnement trop hâtif, donner l’illusion d’une complète indépendance de la couche matérielle ; il n’en est rien. Les objets numériques doivent être pensés comme hybrides, situés au point d’intersection entre la matière et l’abstraction. Autrement dit, bien que le *software*, au mode d’existence immatériel, puisse être abordé fonctionnellement (utilisation, programmation) de façon indépendante, le rôle de la philosophie de la technique est de le

comprendre dans la relation ambivalente qu’il entretient avec le *hardware* et toute la partie matérielle du réseau numérique<sup>95</sup>.

La configuration particulière du logiciel – objet immatériel cohérent supporté par une couche matérielle sous-jacente – a des répercussions sur son analyse psychosociale. Le logiciel ne correspond plus aux cadres d’analyse simondoniens : de par son immatérialité, il ne s’inscrit plus dans un cadre de production industriel. Nous allons tenter de dégager quelques pistes quant aux conséquences pour la pensée simondonienne d’un objet technique post-industriel<sup>96</sup>.

---

95. Nous précisons cela en conclusion.

96. La notion sera définie en III.B.

## Partie III

# Etude psychosociale du logiciel libre

*« On affirme généralement que la production industrielle écrase l’homme parce qu’elle produit des objets qui ne sont pas à sa mesure ; en fait, il serait peut-être plus juste de dire que la production industrielle déroute l’homme parce qu’elle le met en présence d’objets qui ne sont pas immédiatement clairs pour lui ; ils sont tout près de lui en tant qu’objets d’usage, mais ils lui demeurent étrangers, parce qu’ils ne sont pas aisément déchiffrables, et que l’action humaine ne sait plus trouver ses points d’insertion. »*

*– Simondon, Psychosociologie de la technicité, p.65.*

Notre cheminement nous a conduit à faire émerger de la pensée de Simondon une analyse du logiciel informatique. C’est la complexification du codage via la marge d’indétermination présente dans la machine-ordinateur, ainsi que le dialogue permanent entre cette dernière et

l’opérateur humain permis par des périphériques tels que le clavier, la souris et l’écran, qui permet le passage à la limite des objets matériels (*hardware*) aux objets immatériels (*software*). Nous avons établi que le logiciel pouvait être considéré comme un objet technique malgré son immatérialité, puisque la définition de l’objet technique simondonien est avant tout une définition fonctionnelle qui n’engage pas la matérialité de l’objet comme critère premier. Il en résulte un premier moment de définition du logiciel dans une perspective simondonienne : il s’agit d’un objet technique numérique hybride et, ce faisant, complexe.

- Objet technique numérique, car son essence n’est pas matérielle. Le logiciel compris en tant que tel nous apparaît en tant qu’entité graphique composée d’information. C’est là le point qui le distingue de l’objet technique habituel.
- Objet hybride, car ce dernier ne peut toutefois pas fonctionner sans une couche matérielle : contrairement à l’objet technique usuel, qui fonctionne de façon autonome, le logiciel est dépendant d’une couche de technicité première. Du constat d’hybridité du logiciel découle un corrélat important : la couche de technicité logicielle est seconde<sup>1</sup>.

---

1. Cette affirmation n’inféode en aucun cas la couche logicielle à la couche matérielle. Les deux couches qui constituent l’ensemble de l’informatique sont interdépendantes l’une de l’autre : schématiquement, on peut dire que sans couche matérielle, l’information charriée par logiciel ne pourrait exister, s’implémenter quelque part ; inversement, sans couche logicielle, le matériel informatique ne ferait tout simplement rien (sauf si le programme de son fonctionnement était déjà implémenté de façon définitive en lui, mais alors il s’agirait d’automate et non plus des machines ouvertes que sont les ordinateurs). L’expression « technicité seconde » fait allusion non à un

- Objet complexe parce que la complexité est une conséquence de l’hybridité définie ci-dessus (même un petit logiciel avec quelques lignes de code comprend des interactions avec une couche matérielle et un réseau, ce qui en fait un objet complexe). L’objet logiciel, en tant qu’hybride, mélange matérialité (matière physique, tangible) et immatérialité (information binaire) : c’est ce qui fait, en dernier recours, toute la complexité de son essence. Une deuxième couche de complexité se rajoute à cela : celle de l’architecture de certains objets numériques (donc de la couche seconde de technicité). L’exemple fil-rouge utilisé tout au long de notre exposé le montre bien ; le navigateur comporte des centaines de milliers de lignes de code, de fichiers, de ramifications.

De ces premiers jalons de définition émergent une problématique proprement psychosociale, que nous avons posée en dans la partie I. Il s’agit des modalités du couplage objet technique numérique – opérateurs/utilisateurs humains. Analyser les conséquences psychosociales d’un tel objet technique numérique permettra de proposer une définition complète du logiciel selon Simondon, puisque nous avons défini en I sa philosophie de la technique comme étant composée de deux facettes : l’une génétique (que nous avons tenté de prolonger en partie II) et l’autre psychosociale (que nous tenterons de prolonger ici).

Notre analyse portera, dans cette partie, sur les interactions psychosociales dirigées vers la couche de techni-

---

ordre hiérarchique mais à une configuration technique : contrairement aux objets théorisés par Simondon, le logiciel a par essence besoin d’une couche de technicité matérielle pour exister.

cit e seconde : c’est- a-dire sur le logiciel pris comme objet technique en lui-m eme, sans prise en compte de la couche de mat erialit e qui le supporte. Il s’agit de l’appr ehension de l’objet la plus r epandue, et il est important de l’analyser dans un premier temps de fa con ind ependante, pour comprendre les usages qui en sont faits<sup>2</sup>.

Afin de situer le logiciel vis- a-vis des probl ematiques psychosociales propres  a Simondon, il nous faut en d egager les enjeux principaux. Une fois ces enjeux  etablis, nous verrons comment le logiciel libre (et non le logiciel propri etaire) propose une r esolution partielle des probl ematiques psychosociales soulev ees par Simondon.

### III.A Le probl eme de l’objet technique industriel

Les questions d’ echelles sont fondamentales pour comprendre ce qui devient moins un objet technique num erique isol e qu’une r ealit e hybride fonctionnant de fa con r eticulaire. Nous avons choisi de d evelopper le concept de commensurabilit e pour faire r ef erence  a ces probl ematiques. Ce concept n’est jamais mentionn e par Simondon, mais il recouvre une r ealit e d ecrite par l’auteur, et bien pr esente dans la production technique contemporaine.

---

2. Nous reprendrons les fils de l’analyse compl ete du logiciel en tant qu’objet num erique hybride en conclusion.

### III.A.1 La question de la commensurabilité de la technique

La question de la commensurabilité de la technique est au cœur de la philosophie de la technique de Simondon. Sans que cette dernière n’intervienne clairement dans le MEOT, elle y est préparée par la distinction de différents niveaux de technicité<sup>3</sup>. Elle intervient plus directement lorsque ces derniers sont analysés au prisme de leur relation avec l’homme. Quelle échelle de technicité un homme peut-il appréhender seul (et, la compréhension étant le corrélat d’une action potentielle, sur quelle échelle de technicité l’homme peut-il agir ?) Est-ce avec l’élément, l’individu, ou l’ensemble ? Dans la PST, la question de la commensurabilité de la technique et de ses implications pour l’opérateur humain est posée avec force (toujours dans l’optique d’une réconciliation de la technique et de la culture), mais pas de façon frontale. Cette question d’échelle se pose au philosophe car ce dernier prend en compte l’objet technique produit de façon artisanale, et celui produit de façon industrielle. Il constate, entre ces deux modes de production, une différence de commensurabilité des niveaux de technicité : au niveau artisanal, l’objet est compréhensible par l’homme en tant qu’individu ; au niveau industriel, l’individu-objet devient sujet à consommation et à charge psychosociale ; le lien entre le producteur et le consommateur est distendu. L’échelle a changé, puisque la production se fait sur des chaînes où toutes les pièces sont similaires. L’utilisateur de l’objet ne peut suivre le processus de production du début à la fin, ni exprimer des vœux particuliers à

---

3. Voir I.A.1.

l'égard de l'objet. C'est de ce problème de commensurabilité que surgit la dichotomie entre l'objet ouvert et l'objet fermé<sup>4</sup>. Tandis que l'objet artisanal reste ouvert car la production de l'objet est à taille humaine, Simondon constate une double fermeture au niveau de l'objet industriel. Cette fermeture est en partie question d'échelle : « On affirme généralement que la production industrielle écrase l'homme parce qu'elle produit des objets qui ne sont pas à sa mesure »<sup>5</sup>, et en partie (avant tout) question de sens : les objets deviennent « indéchiffrables »<sup>6</sup>.

Ce que nous nommons commensurabilité chez Simondon, c'est donc ce problème d'échelle de la réalité technique, posé avec acuité chez l'auteur dès le MEOT, et approfondi dans les analyses de la PST. La commensurabilité est le fait, pour une réalité technique quelconque, d'être accessible dans son entier par l'homme. L'accessibilité est triple : il s'agit d'une accessibilité à la fois d'échelle (l'individu technique est à la taille de l'homme), entraînant une accessibilité cognitive (compréhension de tous les rouages de la réalité technique en question), permettant elle-même l'action (possibilité d'agir sur chacune des parties de la réalité technique après compréhension des schèmes de fonctionnement). Inversement, il y a incommensurabilité lorsque l'homme, dépassé physiquement par la réalité technique, ne peut saisir dans son entièreté le fonctionnement de cette dernière, et ne peut en conséquence agir sur toutes ses parties.

Les termes du problème sont donc les suivants : à l'époque où écrit Simondon, c'est-à-dire au milieu du XXème siècle, les objets dont le mode de production est

4. Voir I.A.2.

5. PST, p.65.

6. Ibid.

industriel sont dominants. Ces derniers présentent un problème d’incommensurabilité : ils dépassent l’individu humain de par leur système de production qui entraîne leur clôture. L’incommensurabilité inhérente à l’objet fermé sous-tend alors un danger fondamental : celui de l’aliénation.

### III.A.2 La double aliénation des objets techniques industriels

#### La troisième forme d’aliénation

Le concept d’aliénation, pris dans sa généralité, est défini classiquement par Simondon comme le fait d’être autre, de ne plus s’appartenir : ainsi, l’essence de l’esclavage est définie comme « dépendance par rapport à un autre et aux fins d’un autre. »<sup>7</sup> Qui est aliéné est étranger à soi<sup>8</sup>, est dépourvu de capacité d’auto-justification devant soi et autrui. Simondon définit ainsi l’aliénation comme une virtualisation de l’existence de ce qui fait l’objet de l’aliénation : l’existence de l’esclave est « virtualisée », c’est-à-dire qu’il ne continue à exister « qu’autant que son maître l’autorise à exister. »<sup>9</sup> Ajoutons que dans la perspective simondonienne, l’aliénation est un concept essentiellement transitif : une entité est aliénée par rapport à une autre entité (on est aliéné à, et/ou on aliène). L’aliénation est donc à la fois processus sans cesse

7. PST, p.56.

8. Conformément à la racine latine du mot aliénation, « alienus », étranger.

9. PST, ibid.

reproduit (« le processus d’aliénation ») et état de fait, résultat de ce processus à un instant T (« l’aliénation »).

Définir ainsi le processus d’aliénation engendre deux remarques. La première est la distinction établie par Simondon de différents niveaux d’aliénation, plus ou moins premiers. A côté d’une aliénation primaire peuvent exister des aliénations secondaires. Simondon pose le cadre conceptuel de son analyse en se distinguant de Marx et de Feuerbach<sup>10</sup>. Le point commun des deux penseurs consiste pour lui en ce qu’ils situent tous deux l’aliénation basale dans un domaine différent : il voit dans la théorie de Feuerbach que l’aliénation principale réside dans « la séparation qui intervient entre le sacré et l’homme ». Pour Marx, elle réside dans « le jeu de la plus-value dans la relation entre capital et travail »<sup>11</sup>. Ces deux théories défendent une aliénation principale, sous-jacente, de laquelle dépendent les autres.

La seconde remarque porte sur la réversibilité du processus d’aliénation. Le sujet d’aliénation peut être actif ou passif. La théorie marxienne de l’aliénation du travail et par le travail va en ce sens. Pour Marx, le travail procède initialement d’un agent qui exerce librement sa volonté dans la production : « l’homme produit alors même qu’il est libéré du besoin physique. »<sup>12</sup> Dans l’exercice du travail volontaire, l’homme est libre. C’est dans une écono-

---

10. Simondon analyse très brièvement ces deux auteurs. Il ne retient qu’un trait principal de leur pensée ; le but est avant tout de créer un cadre conceptuel pour faire émerger sa propre vision de l’aliénation. Nous ne pouvons pas, faute de temps, entrer dans l’analyse de la discussion détaillée avec Marx et Feuerbach – nous nous contentons donc de faire allusion à ce cadre conceptuel en ce qu’il aide à ressaisir la théorie de l’aliénation propre à Simondon.

11. PST, p.55.

12. MARX, *Le travail aliéné* [76], p.62-63 .

mie fondée sur la propriété privée et le capital que le travail devient aliénant et aliéné : aliénant car l’ouvrier devient étranger au produit de son travail, la plus-value (récupéré par les capitalistes) ; l’ouvrier devient l’esclave de son travail. Travail aliéné, corrélativement, car l’essence libre d’une telle activité se retourne contre elle-même : « l’aliénation de l’objet du travail n’est que le résumé de l’aliénation, de la dépossession, dans l’activité du travail elle-même. »<sup>13</sup> Le travail, libre initialement, est aliéné ; il est alors aliénant pour l’homme qui, à son contact, perd la liberté.

Une telle réversibilité se retrouve chez Simondon. L’originalité de sa théorie réside dans le sujet (actif ou passif) du processus d’aliénation – l’objet technique. En cela, la théorie de l’aliénation simondonienne diffère radicalement des deux théories citées plus haut, qui font toutes deux référence à un individu aliéné par rapport à un système transcendant aliénant (la sacralité ou le capitalisme). Avec Simondon, un passage est opéré de l’individu humain à l’objet technique.

De fait, l’auteur affirme que le processus d’aliénation « primaire », celui qui sous-tend tous les autres, n’est ni l’aliénation par la sacralité, ni celle, économique-sociale, du travail par le capital, mais bien une « troisième forme d’aliénation »<sup>14</sup> : « sous ce rapport juridique et économique de propriété existe un rapport encore plus profond et plus essentiel, celui de la continuité entre l’individu humain et l’individu technique. »<sup>15</sup>

L’objet technique est à la fois aliénant et aliéné : il est dans tous les cas le sujet (passif ou actif) du processus.

13. Ibid, p.58.

14. PST, p.55.

15. MEOT, p.118.

Ainsi, l’objet technique « est ou peut être support et cause d’aliénation ». Précisons que cette dernière se déroule dans un contexte qui est le prolongement de celui dans lequel écrivait Marx (la révolution industrielle). La cause et le support de l’aliénation est donc l’objet technique industriel.

### L’objet aliéné

Le versant le plus contre-intuitif du processus d’aliénation décrit par Simondon est celui de l’aliénation de l’objet<sup>16</sup>. Simondon va jusqu’à affirmer que « l’objet technique industriel est comme un esclave »<sup>17</sup>. Comment comprendre une telle formule ? L’aliénation suppose la confiscation d’une liberté dont l’objet technique, en tant que chose inanimée, n’est pas censé disposer : cela paraît contradictoire. On peut avancer que l’objet technique est la métonymie du travail de son producteur : il est préalablement défini comme « du travail humain concrétisé et détachable du producteur »<sup>18</sup>. C’est donc à la fois le geste d’invention initial et le travail humain concrétisé dans l’objet technique qui est aliéné. Mais l’objet technique n’est pas seulement réduit au travail de l’homme qu’il renferme. C’est également la concrétisation d’une technicité, d’une essence toute particulière. Lorsque Simondon parle d’« objet aliéné », il s’agit de l’aliénation du travail humain mais également, dans un sens plus

16. Cette seconde facette du processus d’aliénation n’intervient que dans la PST, et n’est pas abordée dans le MEOT.

17. PST, p.56

18. PST, p.55

fort, d’une aliénation d’essence, appartenant en propre à l’objet technique, pourtant objet inanimé.

Penchons-nous sur le processus de cette aliénation si particulière. L’objet est aliéné lorsque « la distance augmente entre la production et l’utilisation »<sup>19</sup> - chose qui n’était pas le cas avec la fabrique, où l’artisan était en contact direct avec le client, et produisait en fonction des besoins de ce dernier. Avec le mode de production industriel, l’objet, une fois produit, devient « détachable »<sup>20</sup> : il est autonome mais non pas libre ; c’est la « période intermédiaire »<sup>21</sup> qui est ménagée entre la production de l’objet et l’achat qui l’aliène. Le statut « d’objet technique » nécessite, en plus de l’acte de production, « un second acte de choix qui le reconnaît comme objet jugé digne d’être acheté ». Simondon considère qu’une telle dépendance à l’acheteur aliène l’objet industriel, « virtualise » ce qui constitue son essence, la technicité, en la mettant à la merci de l’appréciation du consommateur. L’aliénation existe lorsque la technicité, « l’historicité » de l’objet, est sanctionnée en dernier recours par des velléités d’achat quasi-contingentes, et souvent déterminées par des facteurs surhistoriques (avec des améliorations de « façade », touchant au paraître mais non à la technicité) La généralisation de la surhistoricité a des conséquences néfastes sur la technicité inhérente aux objets techniques : « L’existence généralisée de la surhistoricité crée chez les constructeurs la nécessité de devenir producteurs de surhistoricité, en créant assez fréquemment des modèles nouveaux, ce qui revient à fractionner volontairement les réformes de structures correspondant à un réel progrès

19. PST, p.55.

20. PST, p.28.

21. PST, p.55.

de technicité, parfois même à les différer. »<sup>22</sup>. Ainsi, « la fonction d’acheteur aliène la fonction de producteur »<sup>23</sup>.

### L’objet aliénant

S’il est support d’aliénation, l’objet technique industriel est également cause de cette dernière. Simondon adopte une définition assez lâche de l’« objet industriel » : il entend par là à la fois l’objet produit (le produit de consommation) et l’objet servant à produire (la machine). Cette imprécision sémantique permet à Simondon d’explorer le processus d’aliénation de l’homme vis-à-vis des deux facettes de l’objet industrielle.

D’abord, l’objet industriel aliénant en tant qu’objet de production (machine) aliène ses producteurs, quelle que soit leur situation hiérarchique dans le système de production. Contrairement à Marx, Simondon déplace le critère premier d’aliénation au sein de l’âge industriel : « l’aliénation de l’homme par rapport à la machine n’a pas seulement un sens économique-social ; elle a aussi un sens psycho-physiologique »<sup>24</sup>. Le processus d’aliénation se trouve au sein des rapports de l’homme (qu’il soit partie prenante du travail ou du capital) avec la machine de la production industrielle : « L’aliénation du capital n’est pas aliénation par rapport au travail (...) mais bien par rapport à l’objet technique. »<sup>25</sup> Il s’agit de rapports d’incommensurabilité fondamentaux : l’ouvrier, qui était auparavant un individu porteur d’outils dans sa fabrique

22. PST, p.58.

23. PST, p.57.

24. MEOT, p.118.

25. MEOT, p.119.

(les outils ne pouvant effectuer le travail seul), se retrouve confronté à un individu-machine autonome qui l'utilise comme outil (l'individu est sous la machine, la nourrissant de gestes mécaniques). Si l'ouvrier est au-dessous de la machine, le patron est au-dessus<sup>26</sup> ; aucun des deux ne connaissent réellement la machine qu'ils emploient, et cette inconnaitance, forme d'incommensurabilité, est à la racine d'un concept d'aliénation plus englobant encore que celui édifié par Marx. Ce n'est pas seulement le travail qui est aliéné ; le capital l'est également. Ainsi, « la relation de propriété par rapport à la machine comporte autant d'aliénation que la relation de non-propriété, bien qu'elle corresponde à un état social très différent. »<sup>27</sup> La rupture du travailleur avec le résultat de sa production est seconde par rapport à une première rupture, non relevée par Marx, qui est celle, généralisée, des opérateurs (qu'ils soient tenants du travail ou du capital) vis-à-vis de l'objet technique sur lequel ils travaillent. Résoudre l'aliénation socio-économique n'empêchera pas l'aliénation par incommensurabilité de subsister : « la collectivisation des moyens de production ne peut opérer que si elle est la condition préalable de l'acquisition par l'individu humain de l'intelligence de l'objet technique individué. »<sup>28</sup>

Ensuite, l'objet industriel en tant que produit de consommation aliène ses consommateurs, car il est clos. La cryptotechnicité<sup>29</sup> de ce dernier empêche l'utilisateur d'en comprendre réellement les rouages, et le rend dépendant de l'objet. L'objet clos est incommensurable :

---

26. L'ouvrier est un « homme des éléments » tandis que le patron est un « homme des ensembles » (MEOT, p.118).

27. MEOT, p.118.

28. MEOT, p.119.

29. Voir partie I.A.2

on ne peut en comprendre les rouages, puisqu’il est une « boîte noire ». L’incompréhension mène à l’impossibilité de l’action sur l’objet.

Pour Simondon, le système de production technique et les objets qui en sont issus deviennent incommensurables à l’homme lors du passage à la production industrielle. Cela a pour conséquence la clôture de l’objet, et l’aliénation de tous les membres de la relation :

- Celle de l’objet technique lui-même, qui se voit départi de ce qui constituait son essence propre, c’est-à-dire de sa technicité, lors de sa mise en vente (virtualisation à la merci des *desirata* d’un marché de consommateurs).
- Celle des hommes par l’objet technique, et plus précisément :
  - Des producteurs par la machine de production
  - Des consommateurs par l’objet de consommation

Le concept d’aliénation développé par Simondon part de l’objet, et a l’objet pour centre. C’est un concept à double face : l’objet est aliéné et aliène, il est à la fois passif et actif. L’objet technique « est ou peut être support et cause d’aliénation, base de processus de causalité cumulative. »<sup>30</sup> L’aliénation est l’enjeu principal de la PST. Résoudre le problème de l’incommensurabilité implique de trouver un moyen de garder les objets ouverts – en dépit de quoi le programme du MEOT -la réconciliation de la technique et de la culture- ne pourrait tout simplement pas être atteint dans le contexte contemporain de Simondon. Deux solutions, à ce stade, s’offrent

---

30. PST, p.55.

logiquement à l’auteur pour résoudre le problème des objets clos en système industriel<sup>31</sup> On peut alors :

- Résoudre le problème de façon externe, c’est-à-dire en tentant de penser un système de production différent de celui de l’industrie,
- Résoudre le problème de façon interne, c’est-à-dire en tentant de trouver une solution qui garde les objets ouverts tout en restant dans les cadres de la production industrielle.

### III.A.3 Sauver l’objet technique en le destinant

Simondon opte pour la seconde option : il tente de sauver le système industriel en faisant changer la technicité d’échelle. La résolution du problème de la commensurabilité à l’homme de l’objet technique individuel se fait, encore une fois, grâce à une analyse de type scalaire : la technicité est déplacée de l’individu au couple élément/-réseau. Tandis que l’objet individuel de consommation, fermé pour les résolutions expliquées plus haut, ne recèle plus qu’une technicité minimale, la technicité « pure » est à trouver dans l’élément (« micro-échelle ») et le réseau (« macro-échelle »).

Ce changement d’échelle induit une profonde modification de la conception de l’objet technique pour Simondon : ce dernier se détache de la notion d’objet technique

---

31. Ces solutions ne sont pas formulées explicitement par Simondon, mais nous tenions à situer les options théoriques de ce dernier dans un cadre plus général. Cela montre aussi « l’optimisme » [35] technologique propre à Simondon : il s’agit de réformer les choses en les prenant telles qu’elles sont, sans renier en bloc le présent).

comme représentant de la technicité, et réhabilite la technicité de l'élément et celle du réseau. Ce n'est donc plus l'objet qui est compris comme paradigme de la technicité. Un tel déplacement est caractéristique de la pensée de Simondon, pour lequel compte moins l'analyse exclusive des objets techniques que celle de la technicité qui les traverse<sup>32</sup>. Le retournement est d'autant plus important que Simondon part d'une analyse première de l'objet technique (comme l'indique le titre du MEOT) pour aboutir à une remise en question de la pertinence de cette échelle via une analyse psychosociale, quelques années plus tard, des liens de l'homme avec les objets, qui l'oblige à déplacer la localisation de la technicité.

C'est au prix de cet effort conceptuel que Simondon arrive à un résultat paradoxal : pour sauver l'objet technique industriel et son ouverture, il le destitue. Ce n'est plus ce dernier qui est le tenant de la technicité, mais les éléments qui le constituent, ainsi que le réseau qui permet la distribution de ces éléments. L'homme accède à un niveau de technicité qui ne se situe pas à son échelle : c'est « précisément ce qui échappe à l'ordre humain de grandeur qui se développe avec la plus faible charge de surhistoricité. » Ainsi, « la technicité vraie est un caractère du réseau d'objets et non de l'objet. »

Simondon cherche à rendre de nouveau ouverts les objets industriels en situant leur technicité non à l'échelle de l'objet, mais à celle de l'élément et du réseau qui relie ces éléments entre eux. Simondon pense avec son époque (qui est encore la nôtre) : le mode de production industriel des objets et son corrélat, la société de

---

32. On l'a dit, philosophie de la technique de Simondon est avant tout une pensée de la technicité et non de l'objet technique, malgré ce que pourrait laisser entendre le titre du MEOT : voir I.A.1.

consommation, y sont dominants. L’idée de « libération de l’élément » est avancée pour sauvegarder l’unité de la culture au niveau industriel.

Mais peut-on considérer que la « libération de l’élément » est menée jusqu’au bout dans le cadre du système industriel ? Permet-elle réellement l’unité de la culture ? Les exemples employés par Simondon, issus du système industriel (pièces détachées de voiture), sont, de l’aveu propre de l’auteur, peu visibles. Ils sont imbriqués dans des individus techniques complexes, peu accessibles aux utilisateurs : ni la chaîne de montage ni la pièce située dans l’automobile ne sont directement accessibles ou modifiables. La solution proposée par Simondon, en renvoyant l’ouverture aux éléments et au réseau, condamne l’échelle de l’objet à la clôture<sup>33</sup>.

Une question se pose alors : qu’en est-il alors de l’idéal des machines à marge d’indétermination du MEOT, permettant une collaboration, un couplage entre l’ensemble technique des machines et l’homme ? Peut-on restituer l’ouverture de l’individu dans le cadre d’un système industriel ? Simondon n’apporte pas de réponse à de telles interrogations. Il émet seulement un vœu ; il faudrait idéalement que le terme médian, l’objet individuel, puisse s’ouvrir de nouveau :

*« La reconstitution de l’unité de la culture demanderait que l’échelon intermédiaire, abandonnant sa charge de surhistoricité, soit lui aussi pénétré de*

---

33. Une clôture s’effectuant via des « opérations irréversibles » : « la soudure, le collage, le rivetage » (PST, p.64). Les propriétés matérielles (voir II.A.1) de l’objet technique empêchent de le rouvrir une fois que ses parties sont soudées.

*technicité, ce qui peut être rendu possible par l'influence des deux autres ordres de grandeur. »<sup>34</sup>*

Une telle impasse existe encore à notre époque au niveau du mode de production industriel des objets techniques. Notre objet d'étude, pourtant, de par sa configuration technique hybride à la croisée de la matérialité et de l'immatérialité, ne répond plus exactement au système de production industriel ; il permettrait alors une ouverture de l'objet plus complète, l'apparition du « terme médian » souhaité par Simondon, que le système industriel, intrinsèquement dépendant de la matérialité de ses objets, ne pouvait produire. Il nous faut maintenant analyser les possibilités d'une telle configuration, ainsi que ses enjeux.

### **III.B La promesse d'ouverture du logiciel en tant qu'objet technique post-industriel**

Notre hypothèse est donc que le logiciel permet d'explorer la première option de renouvellement des objets techniques ouverts : celle de leur ouverture en dehors d'un système de production de type industriel. Cette option n'était pas disponible à l'époque de Simondon - mais puisque nous avons posé l'équivalence entre objet numérique et objet technique, il nous est maintenant possible

---

<sup>34</sup>. PST, p.72.

d’observer comment le logiciel propose de nouvelles solutions d’ouverture.

### III.B.1 Un système complexe

Une objection de taille se pose pourtant à l’affirmation du logiciel en tant qu’objet ouvert, et il nous faut la traiter en premier lieu. Les logiciels apparaissent en effet comme des objets techniques éminemment complexes et ramifiés. C’est le cas de notre exemple, qui n’est pas un objet technique à taille humaine : un·e codeur·euse seul·e, travaillant à l’amélioration du code de Mozilla, ne peut saisir l’ensemble des lignes de code composant le logiciel. « *Mozilla core* », le « noyau » de Mozilla<sup>35</sup>, présente plus d’une dizaine de millions de lignes de code<sup>36</sup>, et leur nombre est exponentiel . Notre navigateur, quant à lui, utilise près de dix millions de lignes de code<sup>37</sup>.

Cette première dimension d’incommensurabilité permet d’assimiler le logiciel à la catégorie des « systèmes complexes » développée par l’ingénierie. Sous cette appellation, il rejoint les objets industriels en tant que « systèmes dont la maîtrise de la conception, de la mainte-

35. « Mozilla Core » est une bibliothèque de code qui regroupe tous le code utilisé pour les applications de Mozilla, parmi lesquelles notre exemple, le navigateur Mozilla Firefox.

36. Mozilla Core contient 15 734 650 lignes de code source, 3 812 280 lignes de commentaires, et 2 812 280 lignes blanches : ce qui donne au total 22 017 612 lignes de code. Source : [https://www.openhub.net/p/mozilla/analyses/latest/languages\\_summary](https://www.openhub.net/p/mozilla/analyses/latest/languages_summary) (consulté le 29.05.2017).

37. 9,7 millions. Source : [https://docs.google.com/spreadsheets/d/1s9u0uprmuJvwr2fkRqxJ4W5Wfomimmk9pwGTK4Dn\\_UI/edit#gid=5](https://docs.google.com/spreadsheets/d/1s9u0uprmuJvwr2fkRqxJ4W5Wfomimmk9pwGTK4Dn_UI/edit#gid=5) (relevé le 29.05.17).

nance et de l'évolution pose des problèmes importants, liés à leur taille et au nombre de technologies utilisées, qui rendent l'ensemble difficile à appréhender. » Leur complexité ne peut être affrontée par « un seul ingénieur talentueux »<sup>38</sup>.

A cette dimension d'incommensurabilité se surajoute deux autres obstacles, que l'on ne trouvait a priori pas dans les objets industriels classiques. Il s'agit de :

- La polyphonie des langages utilisés pour le logiciel<sup>39</sup> : le noyau de Mozilla regroupe l'utilisation de 42 langages de programmation différents () : C++ (33% du code), JavaScript (24,6%), C (14,6%), etc.
- L'extrême rapidité des versionnages () : en un peu plus de dix ans (de 2004 à 2017), cinquante versions différentes de Firefox sont parues.

Considéré comme « système complexe », notre logiciel remplit toutes les caractéristiques de l'incommensurabilité : son échelle n'est pas celle de l'opérateur humain, ce qui entraîne l'impossibilité cognitive de sa compréhension entière par un individu, et, de fait, une impossibilité d'action à l'échelle de l'objet technique entier. Comment peut-on, dans ces conditions, considérer le logiciel comme une alternative significative aux objets industriels analysés par Simondon ? Il semblerait plutôt que nous retom-

---

38. Les deux grands types de systèmes complexes sont les systèmes physiques et les systèmes logiciels. Voir le descriptif de la chaire « Ingénierie des systèmes complexes » : <http://www.enseignement.polytechnique.fr/informatique/chaire-systemes-complexes/-Les-systemes-complexes-.html> (consulté le 29.05.17).

39. Dans un système physique complexe, cela pourrait équivaloir à la complexité des différents modes de fonctionnement.

bions face aux mêmes problématiques que celui-ci : le risque d’une aliénation induite par cette incommensurabilité, ce qui pourrait provoquer une nouvelle fermeture.

### III.B.2 Une configuration post-industrielle

La résolution que notre objet d’étude apporte à ce problème est un peu différente de celle proposée par Simondon, même si elle semble conserver comme effectives certaines des propositions de ce dernier.

Le logiciel permet en effet d’explorer la première voie possible pour éviter l’aliénation ; celle de la sortie du système industriel. Le logiciel est un objet post-industriel<sup>40</sup> en ce qu’il ne participe pas du mode de production industriel classique tel que dominant à l’époque de Simondon. On peut caractériser le mode de production industriel par quelques traits saillants, qui se résument en un seul : le caractère matériel des objets techniques produits industriellement conditionne tout le reste de la production.

— Tout d’abord, la clôture des objets est avant tout une clôture physique, imposée par les propriétés de

---

40. Le qualificatif de « post-industriel » est compris ici en un sens faible, qui ne prend en compte que la configuration technique des moyens de production (passage de la matérialité à l’immatérialité), sans être chargé des enjeux plus généraux relevant de la sociologie post-industrialiste de la seconde moitié du XXème s (A. Touraine, D. Bell), faisant de ces changements techniques un paradigme de société. Il en sera ainsi durant toute la suite de la réflexion.

la matière : si la fermeture est une « opération irréversible »<sup>41</sup> au niveau des objets industriels, c’est que ces derniers sont soumis aux lois des corps physiques, et que les opérations de « soudure, [de] collage, [de] rivetage »<sup>42</sup> ne peuvent être défaits sans rompre l’objet. Par ailleurs, toujours suite à ces propriétés physiques, une telle clôture est nécessaire<sup>43</sup> : pour l’utilisation de l’objet, pour sa cohérence externe en tant qu’objet fini. La clôture physique est nécessaire pour que l’on puisse qualifier un objet technique d’objet, et que l’on puisse l’utiliser en conséquence<sup>44</sup>.

- La matérialité des objets induit aussi le fait, au niveau de la production cette fois-ci, que la reproductibilité des objets engendre un coût de production, comprenant, entre autres, l’alimentation en énergie des machines et des chaînes sur lesquelles l’objet est produit, le salaire des ouvriers le produisant, etc. La production de chaque objet identique demande un peu de travail humain et un peu de matière.
- L’organisation du travail est alors caractéristique :

---

41. PST, p.64.

42. Ibid.

43. C’est pourquoi Simondon fait appel aux pièces détachées pour résoudre le problème de l’objet industriel clos ; l’objet physique a *besoin* d’être clôturé et achevé pour présenter une cohérence externe et être utilisé (c’est tout autant le cas pour l’objet artisanal). Ce qu’il importe, c’est de pouvoir le modifier, le retoucher.

44. Simondon, prenant acte de cet impératif, pense la *réouverture* possible de l’objet (par « retouches successives », PST, p.67, grâce aux pièces détachées en réseau) plutôt que son ouverture complète et continue. Une telle ouverture complète dissoudrait l’objet physique (qu’il soit industriel ou artisanal), en rentrant avec la définition initiale qui fixe dans sa cohérence externe et son utilisation extérieure son statut d’individu.

le processus de production de l’objet est fragmenté, ainsi que Simondon y fait allusion dans la PST. Contrairement à l’objet artisanal, conçu de part et d’autre par la même personne, l’objet industriel est au centre d’un processus de plusieurs étapes ; invention, fabrication, commercialisation.

Dans ce contexte, la solution proposée par Simondon pour conserver l’ouverture est de se résigner à un terme médian (l’objet individuel) fermé, tout en misant sur la manipulation d’éléments ouverts via un réseau de distribution resserré. A l’époque du philosophe, il s’agit de la gestion des pièces détachées. Or, à notre époque, cette notion de pièce détachée tend à être remplacée par une clôture de plus en plus intégrale des objets techniques matériels qui nous sont proches<sup>45</sup>. La solution de Simondon paraît de plus en plus compromise par les effets de clôture des objets produits par les géants industriels, à des fins de capture économique (empêcher le client de réparer son objet technique lui-même). L’aliénation n’est pas complètement évitée avec la solution ménagée par Simondon.

Notre objet numérique, quant à lui, se détache de ces conditions industrielles. Cela se comprend par le fait qu’il s’affranchit de la matérialité. Un objet technique immatériel présente alors des caractéristiques différentes, symétriques à ceux présentés par l’objet technique matériel :

— Tout d’abord, l’objet technique immatériel peut être *ouvert et fermé tout à la fois*. Puisqu’il n’est pas soumis aux lois physiques, il peut se présenter sous

---

45. Les produits de la marque Apple (iPhone ou MacBook) sont un bon exemple d’une telle clôture.

la forme d’une interface utilisateur cohérente et compacte (l’équivalent de l’objet matériel clos) et, simultanément, sous la forme de lignes de codes (l’équivalent de l’objet matériel ouvert). Il est alors possible de modifier en permanence le fonctionnement de l’objet sans que cela ne soit rendu impossible par une nécessaire clôture physique permettant son utilisation.

- De plus, son caractère immatériel permet une reproduction des objets techniques numériques à un coût nul. La reproduction d’un objet numérique n’engage ni de nécessité matérielle, ni de travail supplémentaire.
- Enfin, cela engendre des configurations de travail différentes. Puisque l’objet est immatériel, la chaîne invention/fabrication/commercialisation n’est plus valable. En théorie, puisque le coût de reproduction d’un tel objet est gratuit, et que son immatérialité permet d’abolir les frontières physiques conduisant à la clôture, le logiciel est un objet technique qui permet à la fois l’utilisation et la continuation de l’acte d’invention initial par qui veut. Ainsi, un logiciel devrait pouvoir logiquement être employé comme objet achevé par un utilisateur, et ouvert à tout moment par ce dernier.

Ici, la proposition de Simondon pour pallier à l’incommensurabilité (bien encore valable dans le cas du logiciel, on l’a vu) trouverait son sens plein. Le logiciel permettrait alors une ouverture renouvelée de l’objet technique dans un cadre post-industriel.

### III.B.3 Le logiciel libre, garant de la technicité logicielle

Une dernière précision est ici nécessaire avant d’explorer la configuration d’invention technique singulière que propose le logiciel. Elle est d’ordre terminologique : nous n’avons pas, jusqu’ici, caractérisé le logiciel comme « libre » ou « propriétaire ».

Cela s’explique par le fait que de telles configurations renvoient à deux interprétations distinctes du même objet technique.<sup>46</sup> Ces derniers renvoient à l’échafaudage psychosocial des moyens de gestion et de production de tels objets. Nous voulons montrer ici que l’interprétation du logiciel en tant que logiciel « propriétaire » est un exemple paradigmatique de ce que Simondon nomme un « déphasage »<sup>47</sup>, résultant de l’incompréhension d’une société à son égard. Le déphasage réside en ce que la société interprète l’objet selon des normes passées qui ne correspondent plus à la réalité et aux potentialités de ce dernier<sup>48</sup>.

46. L’être du logiciel reste le même, qu’il soit libre ou propriétaire : un objet hybride à la croisée de la matérialité et de l’immatérialité (voir partie II) ; il nous fallait l’étudier en tant que tel avant de se pencher sur ses différentes interprétations.

47. PST, p.35.

48. PST, p.35 : « Dans les périodes où les techniques se modifient peu, il y a adéquation du contenu culturel et du contenu technique d’une civilisation. Mais, lorsque les techniques se modifient, certains des phénomènes humains constituant une culture se modifient moins vite et moins radicalement que les objets techniques : *les institutions juridiques*, le langage, les coutumes, les rites religieux se modifient moins vite que les objets techniques. » Le logiciel propriétaire est donc, dans des catégories simondoniennes précises, un déphasage de type juridique vis-à-vis de l’évolution des objets techniques logiciels.

Un détour par l’histoire des techniques s’impose ici. Sébastien Broca<sup>49</sup> résume parfaitement l’histoire de l’évolution de la production du logiciel. Au début de l’histoire des ordinateurs, lorsque ces derniers étaient d’énormes machines coûteuses, le transfert de code est monnaie courante : les fabricants de matériel (le plus souvent IBM<sup>50</sup>) fournissent à la fois les machines et les logiciels. Les informaticien·ne·s écrivaient ensuite en interne, souvent avec l’aide du fabricant, les logiciels dont ils avaient plus spécifiquement besoin. Ainsi, les logiciels sont « développés de manière coopérative par le constructeur de matériel dominant et par ses utilisateurs techniquement compétents. »<sup>51</sup> Les logiciels sont une « garniture offerte par les fabricants pour donner plus de saveur à leurs coûteux systèmes informatiques. »<sup>52</sup> IBM intègre donc les modifications des utilisateurs à ses produits pour que cela mette plus en avant ses machines : il s’agit d’une stratégie économique. En des termes simondoniens, le logiciel est encore ouvert ; il est fourni par l’entreprise mais peut être ouvert et modifié par ses utilisateurs.

Deux événements mettent fin à une telle situation de fait. En 1969, IBM se fait poursuivre en justice par le ministère de la Justice des Etats Unis : on lui reproche de maintenir sa position dominante avec une politique de prix anticoncurrentielle liant matériel et logiciel. La firme

49. Ce paragraphe historique reprend la synthèse et les sources de S. Broca dans *Utopie du logiciel libre*, Chapitre I, p.39-41 [30].

50. *International Business Machines*. Entreprise exerçant un quasi-monopole commercial à l’ère des grands ensembles mécanographiques, dans les années cinquante-soixante.

51. E. Moglen, « L’anarchisme triomphant : le logiciel libre et la mort du copyright » [77] (traduction modifiée selon S. BROCA).

52. R. Stallman, S. Williams, C. Masutti, *Richard Stallman et la révolution du logiciel libre. Une biographie autorisée* [102].

cesse alors de fournir ses programmes gratuitement<sup>53</sup>. Un tel événement est renforcé par l’avènement du micro-ordinateur, dont le premier commercialisé est l’Apple II (Altair 8800), conçu en 1975. L’ordinateur pénètre alors dans les classes moyennes<sup>54</sup>.

Ces deux épisodes historiques (le procès contre IBM et l’avènement des micro-ordinateurs) entraîne le développement de la portabilité<sup>55</sup> du logiciel. Ce dernier peut être développé pour différents ordinateurs, ce qui achève de constituer son indépendance. L’expansion des micro-ordinateurs entraîne la multiplication des éditeurs de logiciels commerciaux : le logiciel se constitue comme un objet commercialisable à part entière. Le meilleur exemple de ce tournant est la lettre ouverte adressée en 1976 par Bill Gates (alors jeune programmeur) à la communauté des développeurs. « An Open Letter to Hobbyists » stigmatise les « hobbyists » qui utilisent sans autorisation un logiciel écrit avec Paul Allen pour l’Altair 8800. Il relève que seuls 10% des utilisateurs l’ont payé, et assimile le comportement des 90% restants à du vol, tout en affirmant que le temps passé à développer des logiciels mérite rémunération, comme tout autre activité professionnelle. A partir de cette époque, les programmeur·euse·s sont sommé·e·s de respecter des clauses de confidentialités strictes imposées par leurs employeurs.

L’accès du grand public aux ordinateurs rend le marché du logiciel lucratif. Ainsi, « le monde des informati-

53. Décision qui est connue sous le nom d’ « *unbundling* », (« bundle » signifie « paquet »), voir C. Kelty, *Two bits* [68], p.124.

54. En 1980, 200 000 ordinateurs sont vendus aux Etats-Unis ; en 1985, 10 millions d’ordinateurs sont écoulés. [29], p.211).

55. La portabilité est « la possibilité de transférer un logiciel d’une machine à l’autre, et surtout d’un constructeur à l’autre. » (S. BROCA, *op. cit.*)

ciens passa en quelques années d’une culture professionnelle dominée par des normes universitaires (publicité du savoir, collaboration et jugements par les pairs) à une pratique de la programmation organisée autour d’impératifs commerciaux »<sup>56</sup>. C’est en réaction à ce mouvement de fermeture que s’initie le mouvement libre, en tête duquel Stallman. Il s’agissait de penser de nouvelles pratiques adaptées au format immatériel de l’objet technique, sans se référer à un cadre obsolète ne valant que pour les objets matériels.

L’exemple du *copyright* montre que ces deux interprétations du même objet technique s’adaptent plus ou moins bien à la réalité ontologique de ce dernier.

Le logiciel propriétaire est protégé par un droit d’auteur (*copyright*). Aux Etats-Unis, le droit d’auteur s’applique à « ce qui est exprimé sous une forme tangible ». Les « idées, théories, procédures et méthodes » en sont exclus<sup>57</sup>. Les défenseurs des logiciels propriétaires, s’appuyant sur le fait que le *copyright* donne le droit aux auteurs sur la copie d’un texte donné, affirment que ce dernier est le plus approprié pour régler l’accès au travail logiciel. Mais un tel argument tombe si l’on invoque la complexité ontologique de l’objet logiciel : bien que les algorithmes, en tant qu’idées mathématiques abstraites, ne puissent être soumises au droit d’auteur, les langages de haut-niveau, en tant qu’« expressions textuelles » de ces algorithmes, peuvent l’être. Mais cela est un argument beaucoup trop simpliste, qui ne tient pas compte de la configuration hybride de l’objet-logiciel :

56. S. BROCA, *op. cit.*

57. “The Philosophy of Computer Science”, *Standford encyclopedia* [104].

« On peut objecter à cela qu’une telle affirmation est trop simpliste, et qu’elle ne prend pas en considération l’ontologie propre du logiciel. En effet, les objets informatiques peuvent être examinés à de nombreux niveaux d’abstraction (...) Le problème principal, ici, est de comprendre ce qui, des fonctions, des algorithmes, des programmes, (...) peut tomber sous le droit d’auteur. Par exemple, les algorithmes eux-mêmes peuvent être considérés comme des expressions des fonctions qu’ils implémentent, et, en conséquence, peuvent tomber sous le droit d’auteur. »<sup>58</sup>

Ainsi, « l’ontologie des systèmes logiciels influence le débat sur les droits de propriété des programmes.<sup>59</sup> De plus, la possession d’entités intellectuelles, on l’a vu, n’est pas exclusive, contrairement aux entités matérielles ; de fait, « un objet intellectuel peut être partagé par plusieurs personnes en même temps sans que personne n’en soit lésé. »<sup>60</sup> En ce sens, l’interprétation libre du développement logiciel proposé par GNU/Linux correspond mieux à la configuration technique d’un tel objet : avec le « *copyleft* », les logiciels peuvent être vendus, mais, une fois achetés, les acheteurs peuvent en faire ce qu’ils veulent (le revendre, le distribuer librement, le modifier)<sup>61</sup>, tant que les quatre libertés fondamentales<sup>62</sup> ne sont pas lésées

58. *Ibid.*

59. *Ibid.*

60. *Ibid.*

61. Il s’agit d’une sorte de « copyright inversé », mis au point en 1989 avec l’aide du juriste Eben Moglen pour consolider la licence GPL. Voir glossaire, “Logiciel (du point de vue de sa production)”.

62. Liberté d’exécuter le programme comme l’on veut, d’étudier le fonctionnement du programme, de redistribuer des copies, de dis-

(cela implique donc l’interdiction d’en clôturer le code).

L’interprétation propre au logiciel libre est celle qui retient notre attention car elle est une tentative de compréhension de l’objet technique dans sa singularité. Elle n’est pas déphasée : contrairement au logiciel propriétaire, elle ne tente pas de plaquer des catégories adaptées aux objets physiques (production industrielle, *copyright*) à un objet hybride. L’interprétation « libre » du logiciel<sup>63</sup> permet mieux de comprendre les potentialités d’un tel objet technique, et l’organisation du travail qui peut s’effectuer autour de lui.

### III.C Bricoler l’objet technique numérique

Comment fonctionne l’interprétation libre de la construction logicielle ? Le terme de « bricolage » est apparu comme récurrent dans nos entretiens avec des informaticien-ne-s pour mettre en réflexion la pratique du codage. L’idée de bricolage se retrouve dans les pratiques de « hacking » et lie des valeurs initiales (travail fondé sur la passion et l’intérêt personnel) à des pratiques techniques<sup>64</sup>. Il s’agit ici d’interroger le concept de bricolage,

tribuer des copies des versions modifiées incluant toujours l’accès au code source.

63. Il serait plus approprié de dire « les interprétations » libres : il existe, à côté de la licence GNU et de son *copyleft* que nous avons examiné, beaucoup d’autres licences libres.

64. De telles pratiques sont théorisées par P. Himanen [63] qui identifie le bricolage comme étant une activité au centre de

pris dans sa conception la plus classique, afin d’évaluer dans un premier temps si ce dernier a réellement un impact significatif – s’il désigne un rapport à la technique particulier, ou simplement un mode d’action contingent et anecdotique. Est-ce que le concept de « bricolage » a sa place pour désigner un rapport psychosocial nouveau à la réalité technique ? Nous verrons si un tel concept est à même de prolonger des intuitions simondoniennes, en ouvrant des « points d’insertion » dans les objets techniques au niveau individuel et non plus seulement au niveau de l’élément et du réseau.

Une fois ce premier parcours effectué, nous tenterons d’échafauder une idée de « bricolage informatique », pour répondre à la question des rapports entretenus entre les producteurs et les consommateurs d’un tel objet numérique.

### III.C.1 Enjeux et extension du concept de bricolage

#### Le bricolage comme activité privée

Deux textes complémentaires permettent de délimiter le terme de « bricolage » dans sa version la plus courante : quelques pages de *La pensée sauvage* de Lévi-

---

« l’éthique hacker » et de ses contradictions ; Tristan Nitot parle de « hackability », de « bidouillabilité », désignant ainsi la capacité d’un objet technique « à être détourné de sa fonction première en vue d’essayer de lui trouver de nouveaux usages », pour désigner, précisément, le fonctionnement de Mozilla Firefox et de ses extensions. Voir “A propos de bidouillabilité” [82].

Strauss<sup>65</sup>, et l'extrait d'un entretien avec Y. Deforge autour de la philosophie de Simondon<sup>66</sup>.

Le bricolage est, étymologiquement parlant, ce qui s'applique à éviter un obstacle. Il s'agit d'un « mouvement incident »<sup>67</sup> : le hasard se trouve donc au cœur de l'activité bricoleuse. Par extension, le bricolage est aujourd'hui considéré comme l'œuvre de « celui qui œuvre de ses mains, en utilisant des moyens détournés par comparaison avec ceux de l'homme de l'art. »<sup>68</sup> Le geste de définition du bricolage s'effectue exclusivement via l'action de son agent, le bricoleur. A partir de ce prisme, Lévi-Strauss dépeint quelques traits essentiels du bricoleur et de son activité. L'essence de l'activité bricoleuse est de rassembler des « éléments » déjà existants pour en agencer une « réorganisation » qui aboutira à un résultat « contingent » et incertain. Le propre du bricoleur est donc de composer à partir d'éléments déjà existants, déjà déterminés, en fonction d'un « schème » que le bricoleur a en lui, et qui procèdera d'une « réorganisation » du réel. Ce trait fondamental présuppose une forte caractérisation de « l'univers instrumental » du bricoleur : il s'agit d'un univers « clos ». Les « moyens du bord » correspondent à « un ensemble à chaque instant fini d'outils et de matériaux, hétéroclites au sur-

65. Lévi Strauss, *La pensée sauvage*, Chapitre I « La science du concret », p. 26 à 33 [73].

66. Y. Deforge, MEOT, *Postface*, Question vive n°9, p. 325 à 331.

67. Lévi-Strauss le formule en ces termes : « Dans son sens ancien, le verbe bricoler s'applique au jeu de balle et de billard, à la chasse et à l'équitation, mais toujours pour évoquer un mouvement incident : celui de la balle qui rebondit, du chien qui divague, du cheval qui s'écarte de la ligne droite pour éviter un obstacle. », *Op. cit.*

68. *Ibid.*

plus »<sup>69</sup>. Ces outils ont une histoire particulière, car ils ont initialement été créés pour autre chose – ils sont une fin qui devient moyen. L’univers clos est celui de la prédétermination des pièces utilisées, de leurs limites internes :

*« Ces possibilités demeurent toujours limitées par l’histoire particulière de chaque pièce, et par ce qui subsiste en elle de prédéterminé, dû à l’usage originel pour lequel elle a été conçue ou par les adaptations qu’elle a subies en vue d’autres emplois. »<sup>70</sup>*

La caractéristique première du bricoleur, son jalon de définition essentiel, est donc d’utiliser des éléments qui sont « précontraints » (des « signes »<sup>71</sup>) et de les réorganiser. Le bricolage n’est donc pas la nouveauté radicale : plutôt que d’effectuer une percée dans le réel, d’en découvrir de nouveaux éléments, il est ce qui réorganise le réel avec des éléments déjà connus. Il réalise, via « l’expressivité »<sup>72</sup> de l’individu, de nouveaux ponts entre différents éléments présents via une virtualisation, opérée par l’individu bricoleur, de leurs liens potentiels.

Les caractéristiques secondaires du bricolage, qui découlent de cette définition, sont l’aspect personnel et privé d’une telle activité. Chaque individu évolue dans son cercle instrumental clos dont il dresse l’inventaire

---

69. Ibid.

70. Ibid.

71. Pour Lévi-Strauss, le bricoleur, en manipulant des signes, opère la « réorganisation » de l’ensemble, tandis que l’ingénieur ouvre l’ensemble sur lequel il travaille avec des concepts, en créant de la nouveauté.

72. Y. Deforge, *op. cit.*

avant de s’adonner à l’assemblage hasardeux des éléments, pour le plaisir de faire. Y. Deforge appelle cela « le système de production privé »<sup>73</sup>, et affirme qu’« il n’y a de vrai bricolage qu’en dehors du domaine d’activité professionnelle. » Le bricolage prend alors un tour de jeu : il est « naïveté, ludique, liberté »<sup>74</sup>.

Ainsi, dans la perspective de Lévi-Strauss comme dans celle de Y. Deforge, la fin du bricolage est avant tout l’expression de soi. Y. Deforge l’affirme clairement :

*« (...) la vraie finalité du bricolage est expressive : expression de soi, de son pouvoir créatif, de sa capacité de dialogue avec la matière rébarbative, de son besoin de s’investir dans une œuvre personnelle. »<sup>75</sup>*

En cela, il rejoint la conclusion de l’analyse de Lévi-Strauss : « La poésie du bricolage lui vient aussi, et surtout, de ce qu’il ne se borne pas à accomplir ou exécuter ; il « parle », non seulement avec les choses, comme nous l’avons déjà montré, mais aussi au moyen des choses : racontant, par les choix qu’il opère entre des possibles limités, le caractère et la vie de son auteur. Sans jamais remplir son projet, le bricoleur y met toujours quelque chose de soi. »<sup>76</sup>

Ces analyses refusent au bricolage une portée autre que personnelle. Le bricolage est, par essence, lié au cercle privé, et ne peut faire l’objet que d’une étude descriptive. Il n’est pas question de conceptualiser ce dernier : il

73. Voir *Technologie et génétique de l’objet industriel* [42].

74. Y. Deforge, Postface du MEOT.

75. *Ibid.*

76. Lévi-Strauss, *op. cit.*

restera une activité que l’on ne peut qu’observer structurellement. Les expressions choisies par Lévi-Strauss sont significatives : il s’agit d’utiliser les « moyens du bord », en vertu du principe de « ça peut toujours servir »<sup>77</sup>. La description est présente dans l’usage du discours indirect, qui laisse parler l’acteur du bricolage, comme si une telle activité était si personnelle qu’il était vain et prétentieux de la part du discours réflexif de prétendre réussir à le ceindre entre quelques grands principes. Le bricolage, comme activité intime de travail de réorganisation des éléments, comme « dialogue » avec les éléments, permet l’expression de soi, et ne peut être exprimée en termes généraux sans être détruite dans sa particularité.

### **Le bricolage comme condition d’ouverture de l’objet et comme réappropriation des schèmes inventifs**

On peut cependant noter un mouvement d’élargissement d’une telle pensée du bricolage qui s’amorce dans deux aspects principaux de ces descriptions. En prolongeant ces amorces de réflexions, nous voudrions définir le bricolage comme relevant également d’un enjeu pour la philosophie de la technique.

La première intuition, non aboutie, se trouve dans l’analyse de Y. Deforge. Ce dernier apporte une nouveauté à la pensée de Lévi-Strauss : tout en s’en tenant à la définition structurale initiale (celle de la réorganisation d’éléments préexistants dans un univers instrumental clos), Y. Deforge effectue un pas de côté en ajoutant au commentaire de Lévi-Strauss l’enjeu de l’objet ouvert ou

---

77. Ibid.

fermé, mis au jour par Simondon. On a là une première tentative de problématisation du bricolage, par-delà sa simple description. Le bricolage répond à un problème concret : il est « une réponse pragmatique à la question vive du clos et de l’ouvert »<sup>78</sup> soulevée par Simondon. Ce qui est défini comme « question vive » ici, c’est la problématique psychosociale qui réside au centre de la philosophie de Simondon. Dans un monde industriel où les objets sont résolument clos, comment faire pour retrouver l’ouverture et s’extraire de l’aliénation ?

Cette montée en généralité reste cependant cantonnée au champ individuel – c’est la conclusion sur laquelle retombe Y. Deforge en assignant au bricolage un rôle d’expressivité personnelle. Il nous faut alors radicaliser cette conclusion : croisé aux problématiques simondoniennes, l’enjeu majeur du bricolage n’est pas seulement celui de l’expression personnelle. C’est aussi un mode de relation à l’objet technique qui, à l’ère industrielle, permet la réappropriation de la technicité via le bricolage d’éléments hétéroclites (qui peuvent être au départ des objets fermés) pour créer des objets ouverts. Le bricolage est expressivité personnelle, mais, en des termes simondoniens, il est également voie de réconciliation de la culture et de la technique à l’époque des objets industriels clos. Le bricolage permet à l’homme d’agir « dans les espaces de liberté qui lui restent » et d’« en créer »<sup>79</sup>. Il est ce qui permet de réduire la distance entre la production et l’utilisation, distance créant le problème de l’incommensurabilité du circuit de production des objets fermés à l’ère industrielle. Avec le bricolage, « le bricoleur est à la fois le concepteur, le producteur et le consommateur de

78. Y. Deforge, *op. cit.*

79. *Ibid.*

ses productions. »<sup>80</sup> Simondon l'affirme lui-même, dans une de ses rares allusions au bricolage : « En un certain sens, la condition artisanale de la technicité se réintroduit dans nos mœurs par le biais du travail d'amateur. »<sup>81</sup> Son champ dépasse donc largement celui de la sphère personnelle, même s'il se développe à partir de cette dernière. Il s'agit d'une réponse à une problématique générale (comment se débrouiller face aux objets fermés de l'industrie) qui ne doit pas être sous-estimée, malgré l'aspect a priori privé et individuel de l'activité.

Le bricolage revêt donc également un enjeu psychosocial de taille. Cet enjeu psychosocial (l'ouverture des objets à l'ère industrielle via l'élément) est en fait permis par une caractéristique fondamentale du bricolage, qui touche directement à la partie dite « génétique » de la philosophie de la technique de Simondon : l'invention. Le thème du dialogue, présent chez Lévi-Strauss avec quelque distance (signalée par la mise entre guillemets du verbe « parle », comme on le voit plus bas) et repris plus directement par Y. Deforge, prépare l'enjeu propre à la genèse technique de l'objet sans le développer. Chez Lévi-Strauss, la notion de dialogue est double ; elle est dialogue du bricoleur en son for intérieur et dialogue du bricoleur avec les éléments qu'il assemble : ce dernier « « parle », non seulement avec les choses [...]

---

80. Ibid.

81. La condition artisanale de technicité se définit dans la PST par un rapport de commensurabilité entre l'individu technique, l'artisan et le consommateur, ce qui permet de garder l'objet ouvert et compréhensible. Par opposition, la condition industrielle de technicité se traduit par une incommensurabilité par rapport au processus de production de l'individu technique et une incommensurabilité corrélative au niveau de l'individu technique lui-même.

mais aussi au moyen des choses. »<sup>82</sup> Y. Deforge, quant à lui, décrit la « capacité de dialogue [du bricoleur] avec la matière rébarbative »<sup>83</sup>. Le dialogue avec la matière peut être rapporté au processus de genèse présent en tout objet technique, dont le point de départ est l'invention. L'invention, en des termes simondoniens, est « une prise en charge du système de l'actualité par le système des virtualités »<sup>84</sup> : on retrouve ici la première facette du « dialogue », celui interne à l'individu, qui crée des schèmes techniques potentiels en assemblant une matière ou (dans notre cas) des éléments préexistants. L'invention simondonienne revêt deux facettes : ce jeu de l'imagination qui permet de virtualiser le réel pour l'imaginer configuré de façon différente, puis une concrétisation réelle qui engage l'inventeur dans un dialogue actif avec la matière ou les éléments sur lesquels il avait projeté ses schèmes (seconde facette du « dialogue » avec l'objet). A la lumière de la théorie de Simondon, le bricolage n'est pas une simple activité ; il est souvent ce qui permet de nouvelles inventions via la réutilisation d'éléments déjà chargés de technicité, avant que ces inventions ne soient produites à un niveau industriel. Ainsi, le concept de bricolage se retrouve pour bon nombre d'inventions, qui réutilisent des éléments déjà existants, porteurs de technicité, et contournent peu à peu des obstacles en se confrontant aux réalités et aux difficultés de la matière, qui impose elle aussi certaines de ses lois. C'est en ce sens que l'on peut parler de « dialogue » ; ce n'est pas seulement l'homme qui impose ses schèmes mentaux, mais également les éléments assemblés qui imposent leurs limites

82. Lévi-Strauss, *op. cit.*

83. Y. Deforge, *op. cit.*

84. MEOT, p. 58.

physiques.

Le bricolage est donc, dans sa contingence, ce qui s’effectue dans l’invention simondonienne, sans que Simondon ne le nomme. Il est ce qui permet une prégnation des objets via le paradigme du jeu, c’est-à-dire une compréhension intime de leurs schèmes techniques par un état « d’adualisme primaire »<sup>85</sup>, où le bricoleur, tout absorbé à son ouvrage et aux réponses des éléments qu’il manipule, tente de dialoguer avec eux pour les faire marcher. Chez Simondon, la notion de « jeu » prend alors toute sa profondeur : l’appréhension d’un objet à travers le jeu suscite en l’objet une « puissance archétypale » qui « fait de lui un être, et non pas seulement un objet ». En bricolant, l’homme découvre ou redécouvre des schèmes techniques par un moyen autre que ceux du langage : « La connaissance que donne de l’objet la prégnation n’est pas de nature inductive ni explicitement conceptuelle. Elle est la saisie d’un schème. »<sup>86</sup> Le bricolage

85. Pour Simondon, ce mécanisme instinctif de prégnation (qui est un terme issu des psychologues Lorenz et Tinbergen) connaît une restriction de taille : il ne se trouve presque exclusivement qu’au stade de l’enfance. Après sa définition de la prégnation, il rajoute immédiatement : « [la prégnation] n’est pas possible à n’importe quel moment de la vie ; elle nécessite une particulière attention, un particulier éveil, qui se trouvent surtout possibles dans l’être non-adulte. » (PST, p.43). Nous pensons qu’il est effectivement important de provoquer des situations très tôt car les enfants plus sont malléables, mais qu’il n’est cependant pas impossible d’opérer des processus de prégnation similaires chez l’adulte.

86. On retrouve ici cette idée de sortie du langage pour comprendre d’autres fonctionnements, que nous avons développée par rapport à ce que nous avons nommé la méthode de Simondon en I-B. (faire un renvoi dans cette partie vers le III-B). Nous pousserons plus loin (en conclusion) la comparaison de ces deux modes d’acquisition du savoir (du point de vue de celui qui se veut « philosophe », c’est-à-dire qui veut tenir un discours réflexif, et du point de vue du

permet, via la prégnation de l’objet, un « lien de connaturalité fonctionnelle primitive »<sup>87</sup>, qui n’est pas possible avec l’enseignement théorique qui dicte l’application de principes scientifiques : ce dernier est alors « indirectement et abstraitement connu »<sup>88</sup>.

Le bricolage est donc fondamental, en ce sens qu’il permet à tout individu de transgresser les limites des objets fermés en se réappropriant leur technicité de façon détournée (en les utilisant comme éléments d’un autre dispositif, par exemple, ou en les détournant de leur fonction primaire). Cette réappropriation n’est pas si anodine et contingente qu’il n’y paraît : en bricolant ainsi, l’individu effectue (découvre ou redécouvre), via l’imagination puis l’invention, le processus de genèse des objets techniques. Il s’ouvre aux schèmes de la technicité via une activité qui, si elle est ludique (elle doit l’être), est également fondamentale, car par elle, l’objet peut alors devenir une catégorie culturelle comme une autre, puisqu’appropriée par le bricoleur.

Le bricolage dit « classique », dans sa version ainsi élargie, est une activité humaine qui recèle un double enjeu pour la philosophie de la technique – un enjeu proprement génétique, qu’on peut faire émerger de la notion de « dialogue » entre le bricoleur et la matière, duquel découle un enjeu psychosocial, celui de l’objet ouvert et fermé. Ainsi remodelé, le bricolage est un processus de réappropriation de la réalité technique par l’individu, qui contourne par ce biais le problème du règne des objets clos en système industriel. Grâce au bricolage, l’homme recrée un cheminement technique de compréhension des

---

bricoleur).

87. PST, p.144.

88. Ibid.

objets qui lui permet de se réapproprier les objets techniques via les éléments (les individus clos pouvant devenir des éléments). Le bricolage permet à l’individu de réactiver ses schèmes d’invention et de comprendre intimement la genèse des objets techniques en créant des réalités qui fonctionnent, et non en utilisant simplement des objets. Il est une voie pratique vers l’éducation de la culture aux réalités techniques.

### III.C.2 Le bricolage informatique

#### Echanger les éléments en réseau : définition du bricolage informatique

Qu’est-ce que l’édification d’objet logiciel apporte à la notion de bricolage ? Au premier abord, les codeur·euse·s qui affirment bricoler semblent tenir une position paradoxale. Tandis que le bricolage repose sur une définition implicite de l’objet comme strictement matériel (le bricoleur « œuvre de ses mains »), le bricolage logiciel se confronte à une technicité immatérielle : des lignes d’instruction rédigées en différents langages, une architecture déterminée. Une telle configuration remet le mode de production lié au bricolage en cause. Bricoler des lignes de code pour tenter de faire tourner un programme nécessite de disposer d’un matériel préalable conséquent (un ordinateur, le plus souvent une connexion internet) et de capacités non strictement intuitives (maîtrise d’un langage de programmation, de sa syntaxe, de ses règles). Le bricolage informatique peut rester une activité privée, mais elle s’exerce dans un cadre matériel et intellectuel prédéterminé et nécessaire. De fait, ces pratiques de

« bricolage » peuvent se retrouver chez des codeur·euse·s employé·e·s – ce qui exclue complètement, dans ce cas, l’aspect ludique et contingent d’une telle activité.

Dans sa première confrontation au bricolage « classique », le bricolage informatique apparaît ainsi comme une activité nécessitant un nombre plus élevé de présupposés techniques (matériels et savoirs). Il n’est pas une confrontation directe et instinctive à la matière.

Le dénominateur commun entre bricolage « classique » et bricolage « informatique » est donc à chercher ailleurs. L’équivalence entre les deux peut être opérée en mimant le geste théorique de Lévi-Strauss dans la *Pensée Sauvage*, qui dresse un parallèle entre le mythe et le bricolage. Les points communs des deux activités permettent de faire émerger les caractéristiques structurales de l’activité bricoleuse – puisque le dénominateur commun est bien le bricolage : le mythe est défini comme « bricolage intellectuel », comme une déclinaison, dans la pensée et l’écriture, de ce qui peut se faire sur le plan pratique. Ainsi :

*« Le propre de la pensée mythique, comme du bricolage sur le plan pratique, est d’élaborer des ensembles structurés, non pas directement avec d’autres ensembles structurés, mais en utilisant des résidus et des débris d’événement. »<sup>89</sup>*

Une telle comparaison a beaucoup de valeur pour notre analyse du bricolage informatique. Elle permet d’identi-

---

89. Lévi-Strauss, *op. cit.*

fier le terme commun des deux activités : la réorganisation d’éléments préexistants. En dehors de la matérialité de l’objet (puisque le mythe n’agit pas directement sur de la matière concrète), ces deux activités manipulent des éléments existant dans un cadre prédéfini, et les agencent d’une manière différente en fonction d’un schème inventif.<sup>90</sup>

Les caractéristiques ci-dessus, qui auraient pu faire office d’objection, apparaissent alors comme secondaires et tombent devant l’identité structurale entre le bricolage et le bricolage informatique. Le bricolage informatique peut être appelé bricolage parce qu’il est réutilisation d’éléments préexistants à d’autres fins ; comme le mythe, il n’est cependant pas assimilable totalement au bricolage classique (par l’absence de matérialité, par l’éclatement de l’aspect strictement privé). Il est la déclinaison d’un dénominateur commun (faire avec « les moyens du bord »). Le bricolage informatique se situe d’ailleurs à la croisée des chemins entre le bricolage classique et le mythe, et emprunte à chacun des caractéristiques. Comme le premier, le bricolage informatique voit son action confrontée à des règles de fonctionnement : règles de la matière pour l’un, règles de la mémoire de l’ordinateur et du langage pour l’autre. Comme le mythe, et contrairement au bricolage classique, le bricolage informatique « édifie des ensembles structurés au moyen d’un ensemble structuré, qui est le langage »<sup>91</sup> (on pourrait dire ici : les langages). La notion de langage informatique, langage fonctionnel et performatif<sup>92</sup>, contient en

90. Ibid.

91. Ibid.

92. Voir II.B.2.

elle-même l’hybridation entre l’aspect pratique du bricolage classique et l’aspect sémantique de la mythologie. Contrairement au mythe, qui ne s’empare pas « de la structure du langage » mais qui « bâtit ses palais idéologiques avec les gravats d’un discours social ancien »<sup>93</sup>, le bricolage informatique est plus large, et peut fonctionner selon trois modalités :

- Soit ce dernier s’empare des structures-même du langage pour tenter d’en faire un « brouillon de programme » capable de « tourner » sur la machine, même s’il n’est pas complètement « propre » : il s’agit d’une sorte de premier jet fonctionnel<sup>94</sup>. Dans ce cas, le bricolage informatique est une appropriation des schèmes techniques par le tâtonnement, par l’essai. Les éléments disponibles sont alors puisés dans le langage-même, dans sa structure. Ce langage prédéfini va réorganiser un univers instrumental concret, et clos : celui de la mémoire de la machine.
- Soit il réutilise certaines briques de construction d’un programme pour en construire un autre (on se rapproche ici de la compréhension du mythe chez Lévi-Strauss, et de la conception du hacking en général).
- Soit il surajoute des éléments à un tout fonctionnant de façon autonome.

Les bibliothèques logicielles sont représentatives de telles pratiques : incontournables pour les codeur·euse·s, elles rassemblent des morceaux de programme déjà codés (les routines) qui peuvent s’imbriquer dans un pro-

93. Ibid.

94. Elie Michel, entretien, 28 février 2017.

gramme en cours d’écriture, pour éviter la reproduction lente et fastidieuse de ce qui a déjà été fait. Cela se retrouve à des niveaux plus vastes : si nous plongeons dans le sous-ensemble constitué par le moteur de rendu du navigateur, on se rend compte que ce dernier présente des éléments interchangeables. Les deux principaux moteurs de rendu sont Gecko (produit et utilisé par Mozilla) et Webkit. Il est notable que Safari (navigateur d’Apple) et Chrome (navigateur de Google), pourtant concurrents, utilisent tous les deux webkit, le même moteur de rendu : la même entité technique sert donc à deux individus techniques distincts.

Il s’agit donc bien du détournement d’éléments pré-existants à d’autres fins que celles auxquelles ils avaient été destinés de façon initiale. Il est rare d’écrire un programme de zéro : tou-te-s les programmeur-euse-s s’appuient sur des briques préexistantes, dont l’échange est permis et grandement facilité par la structure immatérielle des objets : à la fois pour leur reproductibilité et pour leur transfert via le réseau qu’est internet. L’objet individuel est ouvert malgré sa complexité : le code qui constitue le programme de Mozilla Firefox est consultable en ligne ou téléchargeable à des fins de modifications personnelles ou communautaires.<sup>95</sup> L’objet individuel peut être repris, travaillé à tout moment.

Un dernier exemple de bricolage informatique est celui des extensions de Mozilla Firefox. Le principe des extensions est simple : il s’agit de micro-logiciels rajoutant des fonctionnalités à l’individu navigateur. Pour ce faire, le-la

---

95. Une plateforme, « Mozilla Developer Network », est prévue à cet effet, selon les besoins des utilisateurs : <https://developer.mozilla.org/en-US/> (consultée les six premiers mois de l’année 2017.)

codeur-euse de l'extension n'a pas besoin de connaître tout le système technique sur lequel il la greffe : il lui suffit de produire un élément compatible qui se surajoutera au fonctionnement de l'ensemble. Les extensions fourmillent dans la communauté Mozilla, et certaines sont très répandues car hautement utiles : ainsi, les bloqueurs de publicité<sup>96</sup> sont des extensions permettant d'empêcher l'affichage des publicités contenues dans les pages HTML que le navigateur affiche.

Un moyen utile de comprendre une telle fonctionnalité est de se pencher sur une autre, plus simple, mais relevant du même fonctionnement : *Beastify*<sup>97</sup>. Cette dernière se répartit en quatre entités distinctes : un ensemble contenant un répertoire d'images, le *script*<sup>98</sup> permettant d'exécuter le programme, un troisième ensemble gérant les icônes de l'extension dans l'interface utilisateur du navigateur, et enfin un quatrième programmant le menu déroulant de l'extension.

*Beastify* est une extension qui programme l'ouverture d'une fenêtre intrusive<sup>99</sup> : elle s'empare du fichier HTML de la page visitée par le navigateur et en supprime le contenu. L'action de suppression est bien visible dans les lignes de code suivantes :

96. Dont *AdBlock*, une des plus utilisées, fait partie.

97. Dont voici la page du tutoriel : [https://developer.mozilla.org/en-US/Add-ons/WebExtensions/Your\\_second\\_WebExtension](https://developer.mozilla.org/en-US/Add-ons/WebExtensions/Your_second_WebExtension). Cette extension, malgré ses airs absurdes, est un exemple pédagogique mis en avant par les développeurs de Mozilla : il est analysé dans le cadre d'un tutoriel permettant de se familiariser avec le système des extensions.

98. En français, « script » renvoie à un petit programme.

99. Les fenêtres intrusives sont l'équivalent, en français, des fenêtres *pop-up*.

```

1 function beastify(request, sender, sendResponse) {
2   removeEverything(); //supprime le contenu
3   insertBeast(request.beastURL); //insertion de l'
   image
4   browser.runtime.onMessage.removeListener(
   beastify); //clôture l'action du programme
5 }

```

Le contenu supprimé est remplacé par l’image de la bête sélectionnée au préalable dans le menu déroulant. Dans notre cas, une grenouille a remplacé le contenu de la page HTML du site <http://www.philosophie.ens.fr><sup>100</sup>.

On peut modifier le code de *Beastify* pour que l’image animalière ne remplace pas l’entièreté de la page HTML mais une image prédéfinie dans cette dernière. Dans le cas de notre exemple, la grenouille remplace la case de *L’Ecole d’Athènes* qui ne comporte pas de personnage<sup>101</sup>.

Cette version plus complexe de l’extension illustre bien un des modes de fonctionnement des bloqueurs de publicité. Ces derniers ciblent ce qu’ils définissent par avance comme publicités, et empêchent leur affichage. Le *networking* ne prend pas en charge lui-même une telle fonctionnalité; ce sont les lignes de code des bloqueurs qui spécifient à ce dernier de ne pas afficher ce qui est considéré comme une publicité. Les bloqueurs de publicité ajoutent donc un filtre à l’affichage des pages HTML. Ces exemples présentent bien des cas plus ou moins complexes de bricolage informatique.

100. Voir Annexes, FIG – H.

101. Voir Annexes, FIG – J.

## Faire face à l'incommensurabilité : organisation du bricolage informatique

L'incommensurabilité d'objets techniques tels que Mozilla Firefox induit une très grande réflexivité de la communauté quant à la gestion de la production d'un tel individu technique. Ainsi, une organisation bien précise préside au bricolage du code. La plateforme de gestion de bugs « *Bugzilla* »<sup>102</sup> en est le meilleur exemple<sup>103</sup>. Il s'agit d'une plateforme permettant la gestion de toutes les erreurs logicielles présentes dans le code source de Mozilla Firefox – et par extension, le développement de tout nouveau projet concernant le navigateur. Lorsqu'un bug apparaît, il est « déposé » sur le site de *Bugzilla*, et classé par un numéro (ID) et une urgence de traitement (comprenant sept niveaux). Un utilisateur peut alors récupérer le fichier du bug pour le faire tourner sur sa machine et en proposer un « *patch* », c'est-à-dire une réparation ; ce patch est soumis à vérification (« *peer review* ») par une personne responsable du fichier dans lequel le bug s'est déroulé. Après cela, le code « débogué » est implanté (« *land* ») dans le code source de Firefox ; si une régression<sup>104</sup> s'opère, il est retiré (« *deland* ») du code source par des modérateurs (les « *sherifs* ») et l'on sélectionne une autre proposition de réparation<sup>105</sup>.

Le logiciel libre présente des communautés organi-

102. Voir <https://bugzilla.mozilla.org/> (consulté en mai-juin 2017), et Annexes, Fig – L,M.

103. Nous devons l'explication du fonctionnement de la plateforme à Stéphanie Ouillon, entretien, 21 avril 2017.

104. Une régression est le fait de faire « régresser » le code en y implantant quelque chose qui l'empêche de marcher correctement.

105. Voir Annexes, Fig – L.

sées et réflexives qui gèrent l’évolution d’un individu technique extrêmement complexe, et pallient à son incommensurabilité via une répartition des tâches précise : ainsi, on parle de « développeur·euse·s logiciels » plus que de « programmeur·euse·s » pour désigner l’ensemble de la communauté travaillant au développement d’un logiciel. Cette dernière ne contient pas que des technicien·ne·s ; elle se compose également de traducteurs, de rédacteurs de manuels d’utilisation, etc.

L’objet technique numérique logiciel libre suscite la réflexion. Sa configuration post-industrielle permet des pratiques de bricolage informatique fondées sur l’échange d’éléments informationnels en réseau ; le logiciel libre préfigure une ouverture renouvelée de l’objet technique.

### **Renouer avec l’objet technique : enjeux du bricolage informatique**

Le bricolage classique, le mythe (« bricolage intellectuel ») ou le bricolage informatique ont ceci en commun qu’ils évoluent tous dans un univers instrumental clos, dont ils réemploient les éléments. Ce sont diverses déclinaisons de ce même dénominateur commun structural. Ainsi, la notion de bricolage peut se développer dans différents domaines. Le bricolage du code permet de se réappropriier les schèmes de technicité de la machine, via la prégnation et les tests répétés. le·la codeur·euse manipule un langage qui est lui-même conditionné par ses interactions avec la machine ; selon le niveau langage utilisé (plus ou moins « haut » ou « bas »), il comprend, en codant, les structures logiques et le fonctionnement de la

machine sur laquelle il travaille. Il s’agit d’un apprentissage par la construction, par l’essai, le contournement des obstacles. Nous avons assisté à un atelier de programmation en langage Python mené avec de jeunes élèves<sup>106</sup> : ces dernières ont manifesté, au terme de l’initiation, leur surprise quant au fait de pouvoir « faire faire » quelque chose à l’ordinateur en écrivant des lignes de code. Apprendre à coder revient à reprendre le contrôle sur une machine où l’on croyait que tout était prédéterminé.

Le codage, et le bricolage qu’une telle activité implique, est une pédagogie pratique de familiarisation et de dialogue avec la machine qui permet de créer de nouveaux liens entre la technique et la culture. Apprendre à coder, c’est apprendre à dialoguer avec la machine, à lui dire ce qu’elle doit faire : c’est se placer sur un pied d’égalité avec elle. Cela est d’autant plus facilité par le fait que l’erreur n’engendre pas de risque physique (contrairement à une erreur lors du bricolage effectif), et qu’il est possible de recommencer indéfiniment<sup>107</sup>. Le regard des programmeur·euse·s saisit l’objet logiciel comme « une matière réformable, prolongeable », à l’instar du regard de l’artisan.

C’est pourquoi la configuration du logiciel résout, en un sens, les problématiques posées par Simondon vis-à-

106. Stéphanie Ouillon, entretien, 21 avril 2017.

107. Précisons qu’une telle situation est inhérente à notre période contemporaine : auparavant, les informaticien·ne·s devaient écrire leur code, le mettre dans la machine, et attendre 24h pour voir si leur code avait « planté » ou non (François Taiani, entretien, 1er mars 2017). Une telle configuration d’immédiateté, qui nous est contemporaine, est bénéfique dans le sens où elle permet d’essayer plusieurs options rapidement ; il faut toutefois prendre garde à savoir également pratiquer le « codage propre », pour ne pas faire un code à la va-vite qui ne soit pas “solide” (c’est-à-dire qui finisse par présenter des erreurs au cours du fonctionnement).

vis du travail de l’élément en réseau et du statut desreprésentants de la technicité. Le logiciel libre, de par sa configuration post-industrielle, permet la libération des problématiques inhérentes à l’industrie. Il est un objet technique inédit qui échappe à la double aliénation relevée par Simondon, provoquée par la clôture de l’objet technique produit industriellement. Il n’est plus aliéné, car il n’est plus seulement produit de consommation, « population d’objets » pouvant être « vendue, achetée, échangée »<sup>108</sup> : il est surtout un produit directement accessible et modifiable par tout utilisateur, donc toujours potentiellement ouvert (chacun peut avoir accès au code source, au fonctionnement de Mozilla Firefox). En théorie, il n’est pas virtualisé par le fait de dépendre ou non de l’achat d’un utilisateur<sup>109</sup>. De même, et réciproquement, il n’est plus aliénant : le logiciel libre est ouvert et chacun peut s’y initier. Il y a dans chaque logiciel libre la possibilité d’ouverture et de modification par l’utilisateur. Cette possibilité n’est pas toujours exploitée, mais elle est sous-jacente et fondamentale. L’utilisateur comme le codeur connaît (ou peut connaître) la structure sur laquelle il travaille/ qu’il utilise.

Plutôt que de travailler au niveau de l’objet (système complexe dont tout un chacun ne peut comprendre tous les détails), le travail se fait sur l’élément technique via un réseau qui l’appuie. La configuration du logiciel comme objet technique numérique corrobore les analyses de Si-

108. PST, p.155.

109. Cet argument ne tient quasiment que sur le plan théorique : Mozilla Firefox en est le meilleur contre-exemple. Le secteur des navigateurs est extrêmement concurrentiel et la course est faite au nombre d’utilisateurs. Nous considérons toutefois que l’objet technique logiciel libre a le grand mérite de ne plus être aliénant, et c’est le point suivant.

mondon, qui situe son époque à l’âge des ensembles dans le MEOT, et à l’âge des éléments reliés via les réseaux dans la PST. Il permet un bricolage collectif d’éléments en réseaux, et semble corroborer la présence de la technicité à ce niveau.

De plus, les codeur·euse·s, en manipulant le code informatique, sont au fait d’un dialogue entre l’homme et la machine – le dialogue souhaité par Simondon lorsqu’il théorise les machines à marge d’indétermination dans l’introduction du MEOT. En codant (et en apprenant à coder), ils donnent des instructions à la machine : ils sont réellement chefs d’orchestre de cette dernière. Le codage permet de renouer avec les machines qui nous entourent au quotidien, pour les comprendre et les utiliser de façon libre et intelligente. C’est en cela que le mouvement du logiciel libre est fondamental à l’histoire des techniques : il revendique explicitement un rapport de collaboration avec les objets numériques que nous utilisons au quotidien.

## Conclusion

« *Ce qui réside dans les machines, c’est de la réalité humaine, du geste humain fixé et cristallisé en des structures qui fonctionnent.* »

– Simondon, *Le mode d’existence des objets techniques*, Introduction.

La philosophie de la technique de Simondon permet de construire le logiciel en tant qu’objet technique – et le logiciel, en retour, éclaire les schèmes techniques simondoniens.

L’approche fonctionnaliste de Simondon, dans la confrontation à ce nouvel objet, se radicalise. On l’a vu, le logiciel se comporte comme un objet technique : il est soumis à une genèse, présente différents niveaux de technicité, et évolue dans un milieu associé. Le logiciel *fonctionne*. Rien n’interdit, à partir de ce constat, de dire qu’il s’agit d’un objet technique comme les autres, et que l’absence de matérialité ne change rien à l’affaire. Le logiciel est un individu technique manipulable par des agents humains, composé d’information et d’interfaces graphiques ; il se matérialise et existe au monde via

une image sur un écran. Les interactions avec ce dernier sont rendues possibles par les périphériques que sont le clavier et la souris. Penser l’objet technique au prisme de son fonctionnement permet de dépasser ses déterminations de forme et de matière (Aristote) ou son ustensilité (Heidegger).

Ainsi, le dialogue entre Simondon et le logiciel adresse une question ouverte à la philosophie de la technique contemporaine : qu’est-ce qu’un objet technique contemporain ? Comment penser le logiciel dans les cadres de la philosophie de la technique classique, avec quelle définition de l’objet ? La question se doit d’être soulevée ; et elle donne raison par-delà les années à la philosophie de la technique simondonienne.

Cette dernière s’en trouve renouvelée. Le logiciel, dans son mode de production libre, apporte une solution à la problématique posée par l’introduction du MEOT : celle de l’amointrissement du fossé entre la technique et la culture. Il est ce qui émerge de la marge d’indétermination : le résultat complexe, traduit en langage de programmation, de restrictions successives de cette dernière. Compris en ce sens, le logiciel est, dans sa version la plus pure, du « geste humain fixé et cristallisé dans des structures qui fonctionnent. »<sup>110</sup> La machine à marge d’indétermination qu’est l’ordinateur présente des possibilités de fonctionnement infinies, et c’est l’homme, en la codant, qui la restreint. La puissance technique de l’informatique émerge d’une collaboration active entre l’homme et la machine ; le·la codeur·euse est le·la chef d’orchestre de la communauté des machines. Il·elle n’existe ni au-dessus d’elles (comme l’ingénieur) ni en dessous (comme le travailleur), mais bien ”au même

---

110. MEOT, Introduction.

niveau qu’elles”<sup>111</sup>. Cette collaboration intime, qui réalise les espoirs de collaboration émis par le MEOT, est permise par la configuration post-industrielle de l’objet logiciel. La technicité logicielle, détachée de la matérialité, se construit à partir de l’échange d’éléments numériques en réseau ; le bricolage informatique qu’effectuent les acteur·trice·s de l’informatique en est la traduction pratique. De même, un tel contexte permet l’avènement de représentants qui dialoguent réellement avec les machines sur lesquelles ils travaillent via le codage puisque ce dernier est, dans sa définition la plus simple, restriction de la marge d’indétermination. Le logiciel est le résultat d’une relation ouverte entre la machine à calculer et le·la technicien·ne (relation que le·la technicien·ne restreint temporairement). Le codage informatique consacre l’avènement de la synergie fonctionnelle tant souhaitée par Simondon, où « une fonction unique et complète est remplie par deux êtres.<sup>112</sup> »

S’interroger sur le statut du logiciel revient à questionner la définition des objets techniques contemporains en général. Le cadre de pensée fonctionnaliste de Simondon permet d’appréhender de façon adéquate un tel objet, en acceptant sa technicité malgré son absence de matérialité. Il s’agit d’un apport important pour la philosophie de la technique. La pensée d’un objet technique fonctionnant sans matière vient, en retour, modifier les possibles rapports des agents humains aux machines, dans le sens d’une synergie plus grande.

La feuille blanche d’un logiciel de traitement de texte est donc un objet technique numérique avant d’être une

111. MEOT,p.125.

112. MEOT, p.173.

feuille blanche. D’après notre analyse, elle est une couche de technicité secondaire et immatérielle qui émerge à partir d’une couche de technicité primaire – celle de la machine à indétermination qu’est l’ordinateur. Cette couche de technicité secondaire peut parfaitement se comprendre de façon indépendante quant à la couche de technicité primaire. Si le logiciel qui fait apparaître la feuille blanche est un logiciel propriétaire, il s’agit d’un objet technique clos ; impossible d’en comprendre les mécanismes internes ou d’y avoir accès. Il nous est incommensurable et la seule façon d’y avoir de nouveau accès sera de payer à nouveau le prix de la licence. Si le logiciel est un logiciel libre, l’objet technique sera ouvert. Il n’est pas exclu que nous ayons à payer une licence dans ce cas également ; en revanche, nous pourrions comprendre son fonctionnement interne grâce à l’accès ouvert au code source, et nous le réapproprier en le modifiant si nous le désirons.

Le fait de pouvoir traiter notre feuille blanche comme un objet technique indépendant ne doit toutefois pas faire oublier son aspect hybride. La feuille blanche et le logiciel qui la sous-tend sont une couche de technicité secondaire : ils dépendent de la matérialité de l’objet dans laquelle le code source du logiciel de traitement de texte s’exécute. Un code source non implémenté équivaut au plan architectural d’un immeuble sans la construction effective de ce même immeuble.

La nature hybride du logiciel fait écho à une autre expérience-limite expérimentée par l’universitaire malchanceux, plus radicale encore que la première. Ce dernier, ayant compris la leçon, écrit maintenant sur un logiciel de traitement de texte libre, qui ne retiendra pas en otage son travail. Pourtant, un matin, son ordinateur ne s’allume plus. Il lui est impossible de l’ouvrir et de

rechercher la source de la panne par lui-même : il s’agit d’un ordinateur portable totalement clos, très difficilement démontable.

Que se passe-t-il ici ? L’accès au logiciel libre hébergeant la feuille blanche de notre universitaire est compromis par la défaillance du support matériel. L’ordinateur, en tant que couche de technicité première, est ce qui conditionne en derniers recours l’existence de l’objet technique numérique. Or il s’avère que ce dernier est, la plupart du temps, clos. Dans un cadre d’analyse simondonien, on aboutit alors à une situation profondément paradoxale. Tandis que la couche de technicité première, soumise à des logiques matérielles et industrielles, reste la plupart du temps close, la technicité secondaire, qui déploie des logiques immatérielles et post-industrielles, fait montre d’ouverture. Le mode d’existence hybride du logiciel déplace, en réalité, l’aliénation psychosociale. L’ouverture des objets techniques immatériels tend à faire oublier la fermeture de leurs supports matériels – desquels ils sont pourtant dépendants en dernier recours. Le rapport de bricolage, garant de l’ouverture de l’objet, s’est beaucoup moins développé au niveau de la matérialité de la machine<sup>113</sup>.

Nous sommes ainsi devant un second paradoxe déployé par le numérique, qui précise celui que nous avons relevé en introduction. Nous avons affirmé que le numérique sous-tendait une lourde technicité qui était obli-

---

113. Il se développe tout de même, mais de façon beaucoup moins visible, dans le courant du « *open hardware* », qui consiste à entretenir un rapport d’ouverture et de bricolage avec l’objet-ordinateur lui-même. Ce mouvement est une piste qui devrait être impérativement approfondie pour toute étude de la réalité logicielle dans sa généralité – et pour tout examen à visée politique des possibilités de son ouverture complète.

rée par les relations d’usage que nous entretenons avec ce dernier. Précisons maintenant : le numérique est ouvert en certains endroits (ceux des logiciels libres, compris comme objets numériques immatériels), mais de plus en plus clos en d’autres. L’inconscient numérique se situe de plus en plus au niveau de sa matérialité. Au niveau psychosocial, l’aliénation réside en ce que la complexité du logiciel tend à faire oublier son aspect matériel au profit de son aspect immatériel libre.

Au cours de cette étude, nous avons qualifié le logiciel d’objet hybride. Une telle notion mériterait d’être approfondie plus encore au prisme de la philosophie de l’informatique anglo-saxonne<sup>114</sup>, afin de définir quelles sont les modes de relations entre le *hardware* et le *software*. Le logiciel y est considéré comme faisant partie de la classe des « objets informatiques » (« *computational artifacts* »), et présentant de fait une nature duale<sup>115</sup>. Le courant de la philosophie de l’informatique tente de dresser une ontologie du programme informatique en définissant la nature du lien entre le *hardware* et le *software*, et en insistant sur le fait qu’il est absolument nécessaire de prendre en compte la dimension matérielle du logiciel :

*« It is important to remember that computer programs can be understood on the physical level as well as the symbolic level. The programming of early digital computer was commonly done by plugging in wires and throwing switches. Some analogue computers are still programmed in this way. The resulting*

114. Dont l’article « The Philosophy of Computer Science » dans l’encyclopédie de Standford dresse un compte-rendu [104].

115. Ibid.

*programs are clearly as physical and as much part of the computer system as any other part. Today digital machines usually store a program internally to speed up the execution of the program. A program in such a form is certainly physical and part of the computer system.”<sup>116</sup>*

La philosophie de l’informatique s’attache à prendre en compte l’aspect matériel du logiciel pour en dresser une ontologie complète et cohérente. Les penseurs du logiciel ont fourni plusieurs réponses à la question de la relation entre matérialité et immatérialité dans le logiciel. T. Colburn définit le logiciel comme une « abstraction concrète<sup>117</sup> » contenant un médium de description (le texte) et un médium d’exécution (l’implémentation concrète dans des circuits électriques). J. Fetzer, quant à lui, voit le programme comme la théorie de son implémentation physique : le programme est alors considéré comme un modèle causal ; il est une description textuelle de sa réalisation physique<sup>118</sup>.

Les cadres conceptuels proposés par la philosophie de l’informatique permettraient de renforcer ce dont nous avons jeté les bases à partir d’une analyse simondonienne – à savoir la définition ontologique du logiciel dans son entier. Une telle définition réside dans la résolution d’une problématique majeure : celle de la *relation* entre l’aspect matériel du logiciel d’une part, et son aspect immatériel de l’autre. Cette première problématique serait efficacement complétée par une seconde : l’étude de l’impensé

116. J. Moor, “Three Myths of Computer Science” [79].

117. T. Colburn, “Software, Abstraction, and Ontology” [37].

118. J. Fetzer, “Program Verification : The Very Idea” [51].

matériel qui se niche au cœur de nos relations avec le numérique. Approfondir les intuitions développées dans notre étude impliquerait alors de creuser du côté d’un courant intellectuel récent consacré à l’archéologie des infrastructures. Ce dernier vise à mettre au jour la matérialité de notre environnement numérique (des câbles sous-marins aux data centers jusqu’aux déchets produits par ce dernier). Il soulève des problématiques telles que l’impact écologique désastreux<sup>119</sup> de ce dernier, ou les dynamiques géopolitiques qui sous-tendent sa répartition géographique. Un tel courant met au jour un réel « inconscient numérique »<sup>120</sup> qui empêche tant les usagers que les penseurs de la technique d’appréhender cette dernière au prisme de ce qui la conditionne en dernier recours : une matérialité tentaculaire, parcourue de rapports de domination violents (domination de la nature, domination des utilisateurs).

Selon nous, l’approfondissement de l’ontologie du logiciel ne peut se faire seule. Elle est inséparable de la prise en compte de l’oubli total de la matérialité numérique, dont notre époque déploie le paradoxe gigantesque. Il est impossible de traiter du numérique en n’accordant crédit qu’à la réalité immatérielle à laquelle ce dernier renvoie, car les dynamiques d’aliénation les plus puissantes et les plus primaires se situent au niveau des réseaux matériels en eux-mêmes. Cette aliénation, trop souvent oubliée, est pourtant celle qui conditionne notre emploi quotidien des

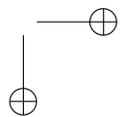
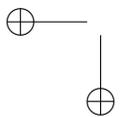
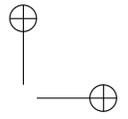
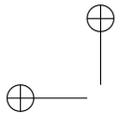
---

119. Prise dans son ensemble, « l’empreinte carbone totale des serveurs disséminés sur la planète dépasse déjà celle de tout le transport aérien, et va encore probablement tripler d’ici 2020. » D’après B. Bratton, *The Stack. On Software and sovereignty* [28], p.92-93 (cité par Y. Citton dans la *Revue du Crieur* [36]).

120. Y. Citton, *op. cit.*.

logiciels en dernier recours. Une telle étude de caractérisation du numérique (sur le versant ontologique et sur le versant politique) s’effectuerait en maintenant le geste d’analyse simondonien effectué dans ce mémoire. Des exemples caractéristiques devraient être analysés finement afin d’informer la réflexion théorique. Il serait alors intéressant de sélectionner deux pratiques renouvelant le bricolage informatique immatériel au niveau matériel, afin de d’ouvrir des perspectives d’action politique quant à notre impuissance actuelle vis-à-vis de la matérialité du numérique. L’étude de cas du mouvement « open hardware », ainsi que celle des fournisseurs d’accès coopératifs, seraient opportunes ici - en ce qu’elles déploient des pratiques de bricolage qui visent cette fois l’ouverture du support matériel de la réalité logicielle.

L’objet technique immatériel n’est pas si « libre » du point de vue de sa définition ontologique complète. Ce paradoxe dessine un programme de recherche qui se doit d’essayer de tenir ensemble les deux dimensions ontologiques du logiciel, afin de comprendre précisément quels sont les enjeux d’aliénation qui le sous-tendent. Mener une telle étude revient à questionner ce qui semble être la plus grande question de notre époque, et qui était déjà sous-tendue dans les analyses de Simondon : celle de l’incommensurabilité de nos objets techniques contemporains.



# Annexes

## Liste des entretiens :

Elie Michel, le 28 février 2017.

François Taiani et Davide Frey, le 1er mars 2017.

Mathieu et Lunar, le 1er mars 2017.

Rémi Hubscher, le 1er mars 2017.

Emmanuel St-James, le 17 mars 2017.

Irlande Saurin, le 28 mars 2017.

Hellekin, le 7 avril 2017.

Stéphanie Ouillon, le 21 avril 2017.

Jean-Yves Château, le 6 juin 2017.

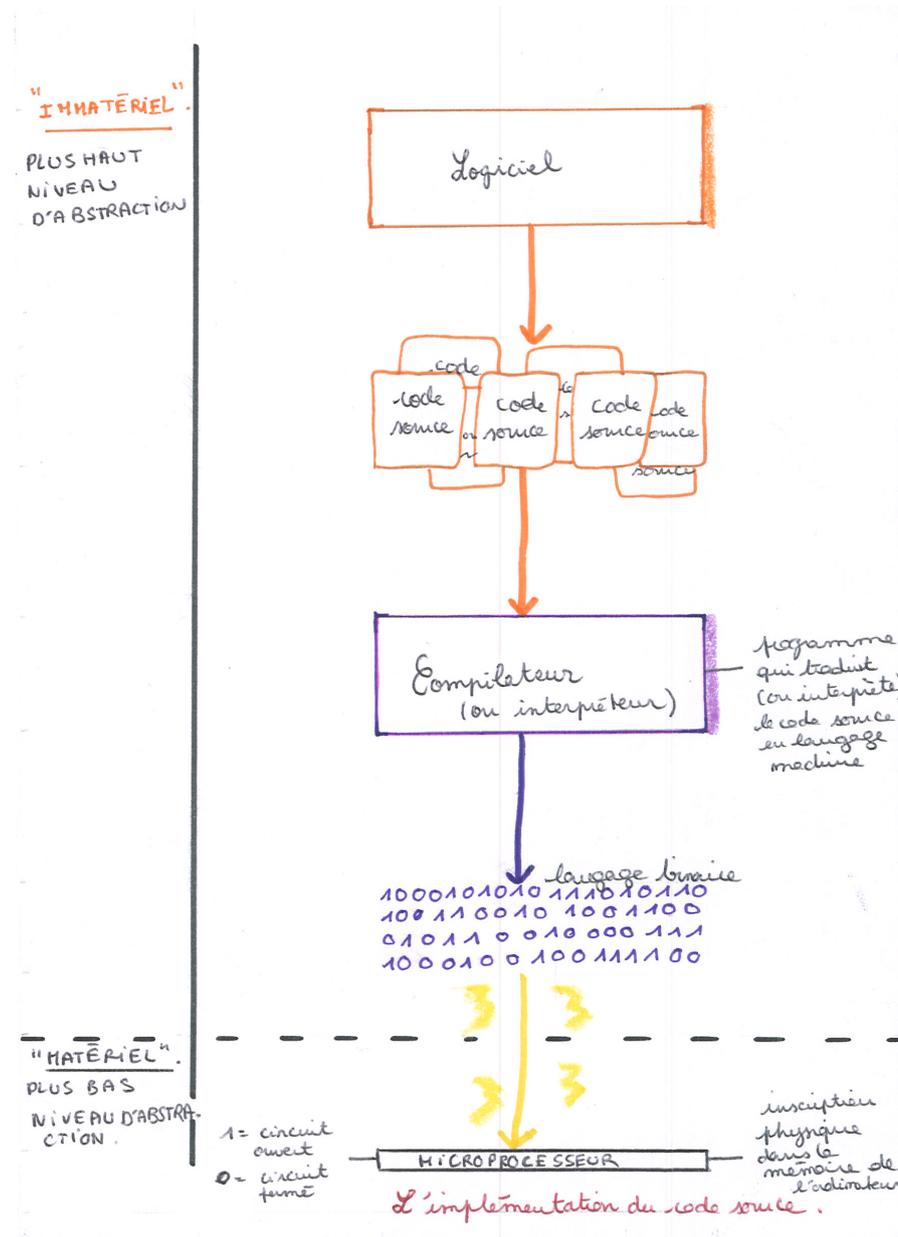
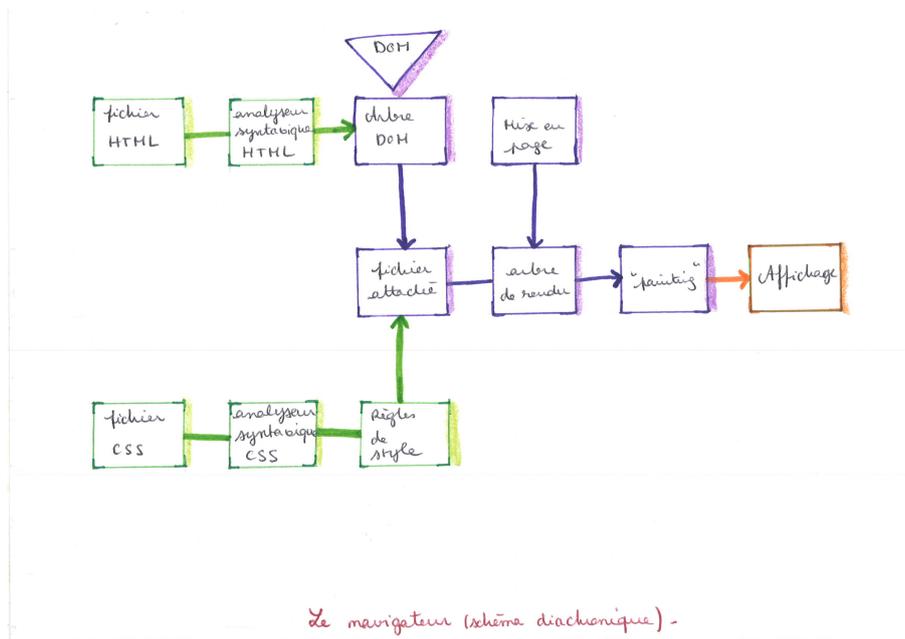


FIG. A – L'implémentation du code source.



*Le navigateur (schéma diachronique) -*

FIG. B – Schéma diachronique du moteur de rendu.

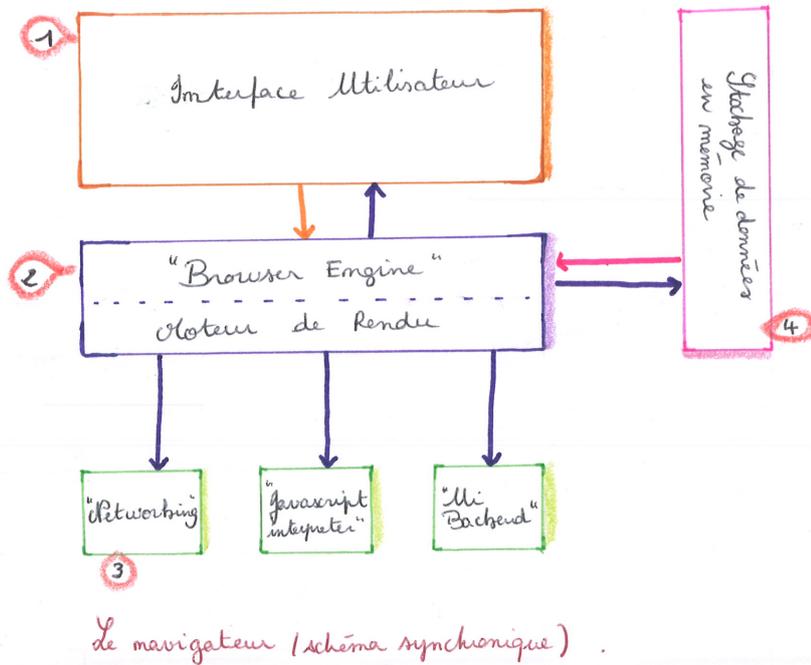


FIG. C – Schéma synchrone du navigateur.

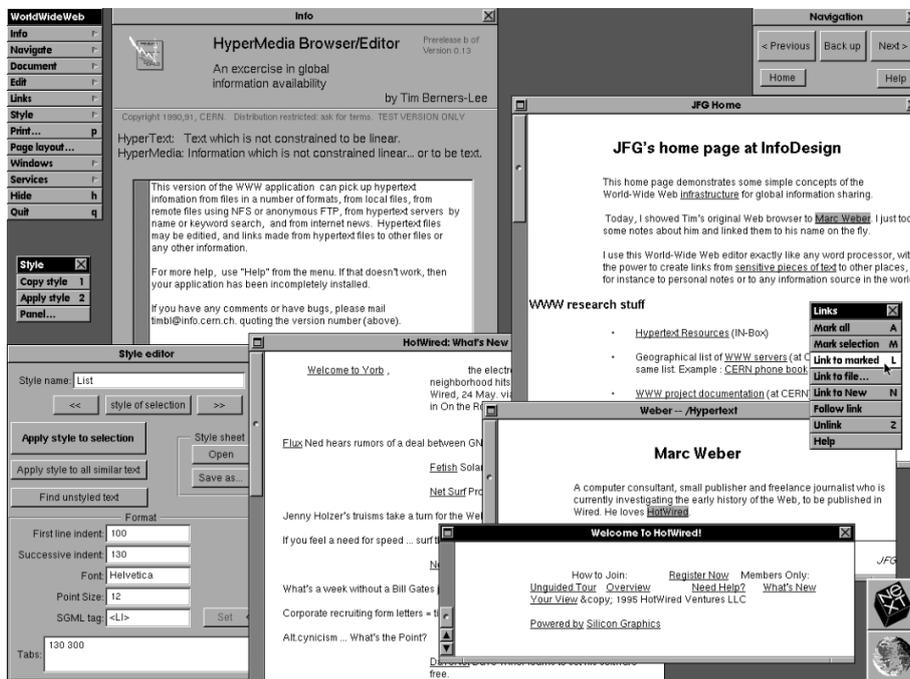


FIG. D – Aperçu du navigateur WorldWideWeb (1990).

Source :

[www.w3.org/People/Berners-Lee/WorldWideWeb.html](http://www.w3.org/People/Berners-Lee/WorldWideWeb.html)

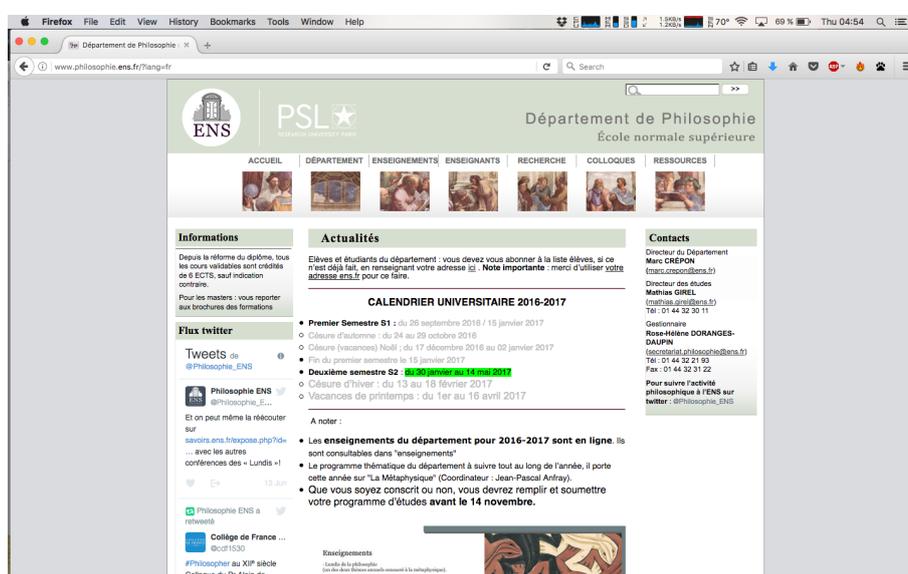


FIG. E – Aperçu du navigateur Mozilla Firefox (2017).

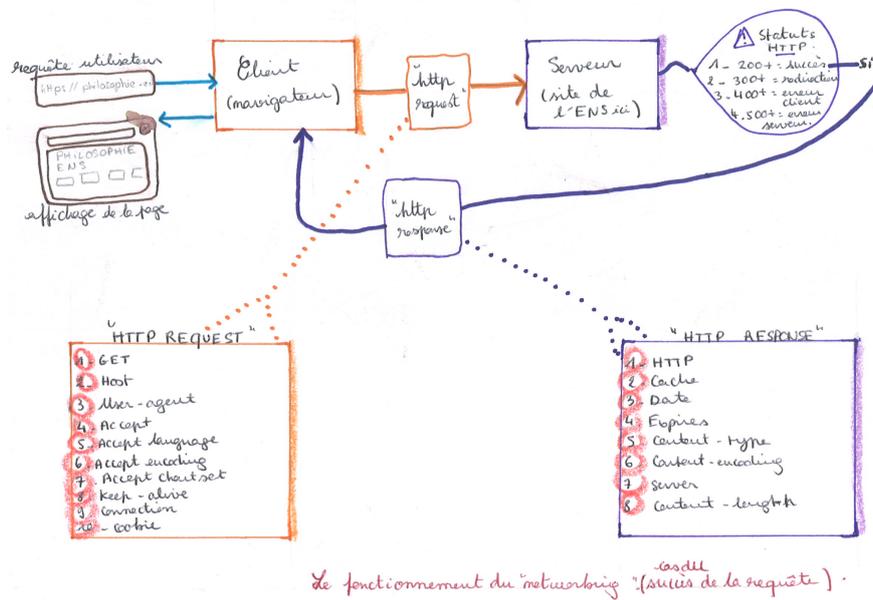


FIG. F – Fonctionnement du *networking* et illustration du protocole HTTP.

### **Requête HTTP**

1. GET .. etc : action engagée sur le serveur (récupérer les informations, les modifier si cela est autorisé).
2. Host : nom du domaine (ex. www.philosophie.ens.fr).
3. User-Agent : informations sur le client (navigateur utilisé, système d’exploitation).
4. Accept : extensions que le navigateur accepte.
5. Accept-language : langue du navigateur.
6. Accept-Encoding : encodages que le navigateur accepte.
7. Accept-Charset : caractères d’écriture que le navigateur accepte.
8. Keep-Alive : temps d’ouverture de la connexion entre le client et le serveur.
9. Connection : spécifie le type de connexion.
10. Cookie : retient des informations de la navigation, stockées sur le serveur.

### **Réponse HTTP**

1. HTTP : version du protocole utilisée pour le message.
2. Cache-control : cache retenant des informations sur le client (pour envoyer des pages personnalisées, etc).
3. Date : date et heure de renvoi de la réponse.
4. Expires : moment où la page doit être réactualisée.
5. Content-type : décrit le type de contenu envoyé.
6. Content-Encoding : réponse à ce que le client supporte.
7. Serveur : nom du logiciel supportant le serveur.
8. Content-Length : taille (en bytes) des fichiers envoyés.

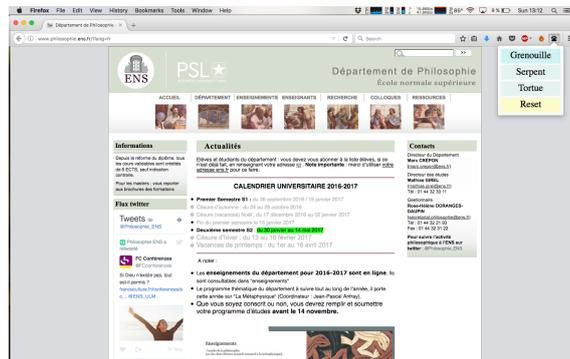


FIG. G – Avant *Beastify* 1.0.

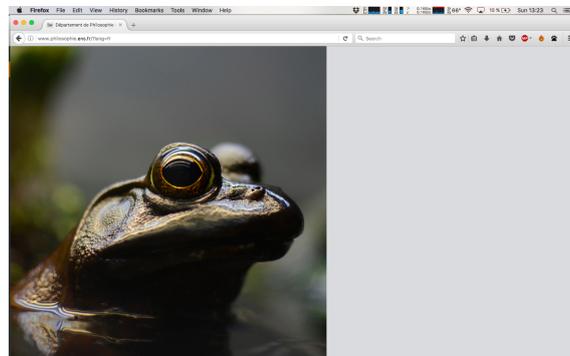


FIG. H – Après *Beastify* 1.0.

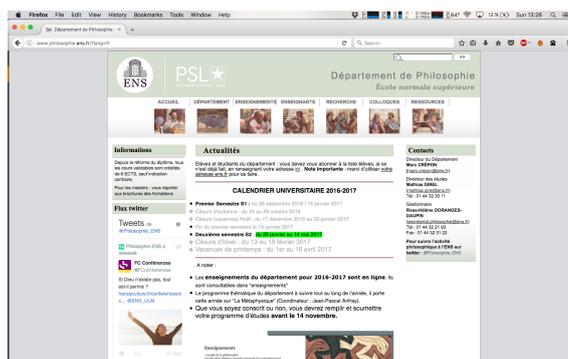


FIG. I – Avant *Beastify* 2.0.

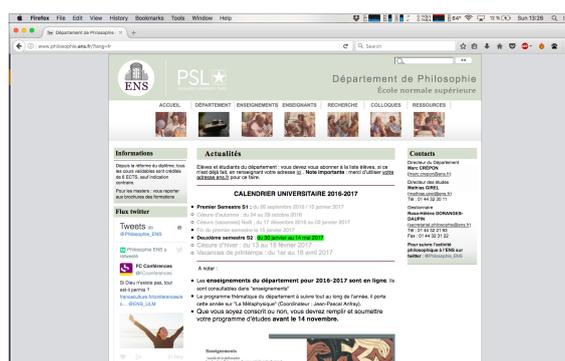


FIG. J – Après *Beastify* 2.0.

Illustration du fonctionnement de *Beastify* sur le site <http://www.philosophie.ens.fr/>. La version 1.0 correspond à la version du tutoriel de MDN : [https://developer.mozilla.org/fr/Add-ons/WebExtensions/Your\\_second\\_WebExtension](https://developer.mozilla.org/fr/Add-ons/WebExtensions/Your_second_WebExtension). La version 2.0 correspond aux modifications que nous y avons ajouté.

```
1 beastify/  
2  
3   beasts/ //répertoire avec les images  
4     frog.jpg  
5     snake.jpg  
6     turtle.jpg  
7  
8   content_scripts/  
9     beastify.js //programme transformant le  
10      fichier HTML  
11  
12  icons/ // icônes pour la barre d'outils  
13     beasts-32.png  
14     beasts-48.png  
15  
16  popup/  
17     choose_beast.html //fichier HTML codant le  
18      contenu de la fenêtre intrusive  
19     choose_beast.css //feuille de style de la  
20      fenêtre intrusive  
21     choose_beast.js //traite le choix et lance  
22      beastify.js  
23  
24  manifest.json //fichier d'information pour  
25      Mozilla Firefox
```

FIG. K – Architecture du code de l’extension *Beastify*

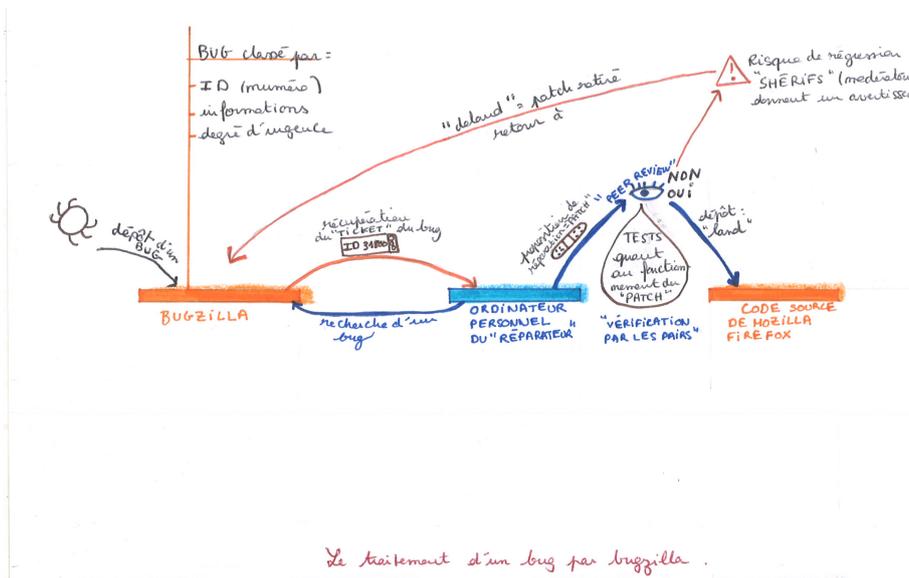
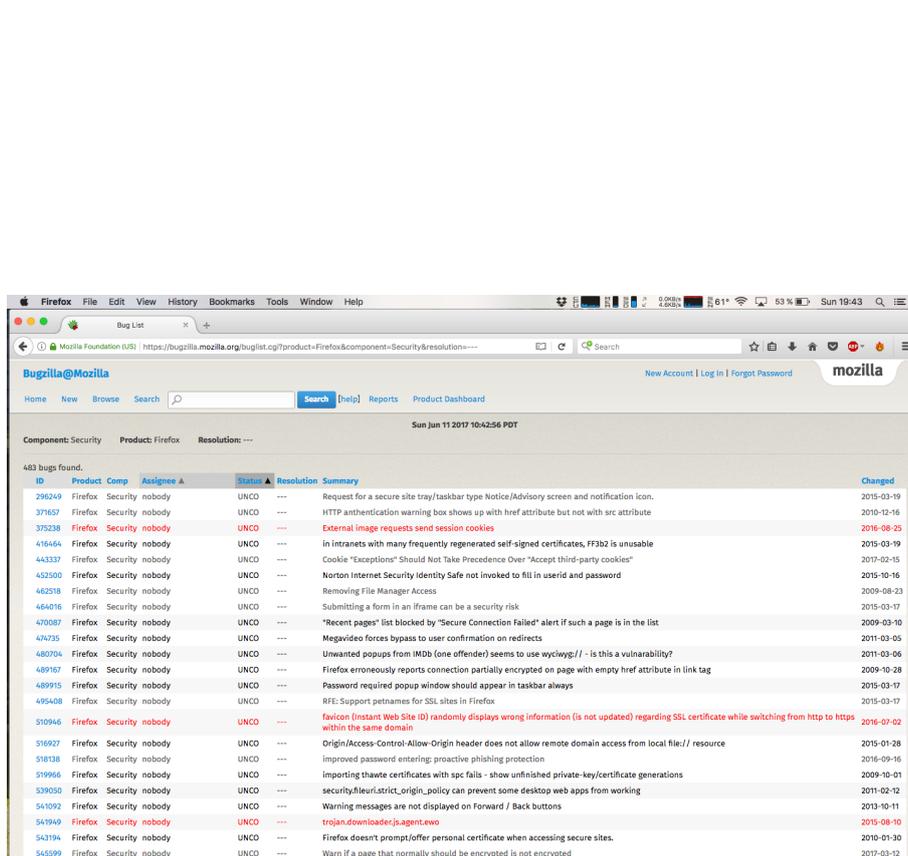


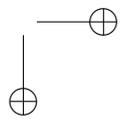
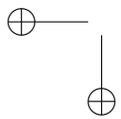
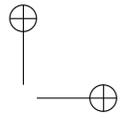
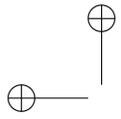
FIG. L – Le traitement d’un bug par Bugzilla.



The screenshot displays the Bugzilla web interface in a Firefox browser window. The browser's address bar shows the URL: <https://bugzilla.mozilla.org/buglist.cgi?product=Firefox&component=Security&resolution=--->. The page header includes the Mozilla logo and navigation links like 'Home', 'New', 'Browse', and 'Search'. Below the header, the search filters are set to 'Component: Security', 'Product: Firefox', and 'Resolution: ---'. The main content area shows a table of 483 bugs found. The table has columns for ID, Product, Comp, Assignee, Status, Resolution, Summary, and Changed. The first few rows of the table are as follows:

ID	Product	Comp	Assignee	Status	Resolution	Summary	Changed
296249	Firefox	Security	nobody	UNCO	---	Request for a secure site tray/taskbar type Notice/Advisory screen and notification icon.	2015-03-19
371657	Firefox	Security	nobody	UNCO	---	HTTP authentication warning box shows up with href attribute but not with src attribute	2010-12-16
375238	Firefox	Security	nobody	UNCO	---	External image requests send session cookies	2016-08-25
416464	Firefox	Security	nobody	UNCO	---	in intranets with many frequently regenerated self-signed certificates, FF3b2 is unusable	2015-03-19
443337	Firefox	Security	nobody	UNCO	---	Cookie "Exceptions" Should Not Take Precedence Over "Accept third-party cookies"	2017-02-15
432500	Firefox	Security	nobody	UNCO	---	Norton Internet Security Identity Safe not invoked to fill in useraid and password	2015-10-16
462518	Firefox	Security	nobody	UNCO	---	Removing File Manager Access	2009-08-23
464016	Firefox	Security	nobody	UNCO	---	Submitting a form in an iframe can be a security risk	2015-03-17
470087	Firefox	Security	nobody	UNCO	---	"Recent pages" list blocked by "Secure Connection Failed" alert if such a page is in the list	2009-03-10
474735	Firefox	Security	nobody	UNCO	---	Megavideo forces bypass to user confirmation on redirects	2011-03-05
480704	Firefox	Security	nobody	UNCO	---	Unwanted popups from IMDb (one offender) seems to use wycwylg// - is this a vulnerability?	2011-03-06
489167	Firefox	Security	nobody	UNCO	---	Firefox erroneously reports connection partially encrypted on page with empty href attribute in link tag	2009-10-28
489915	Firefox	Security	nobody	UNCO	---	Password required popup window should appear in taskbar always	2015-03-17
495408	Firefox	Security	nobody	UNCO	---	RFE: Support petnames for SSL sites in Firefox	2015-03-17
510946	Firefox	Security	nobody	UNCO	---	favicon (Instant Web Site ID) randomly displays wrong information (is not updated) regarding SSL certificate while switching from http to https within the same domain	2016-07-02
516927	Firefox	Security	nobody	UNCO	---	Origin/Access-Control-Allow-Origin header does not allow remote domain access from local file:// resource	2015-01-28
518138	Firefox	Security	nobody	UNCO	---	Improved password entering: proactive phishing protection	2016-09-16
519966	Firefox	Security	nobody	UNCO	---	Importing thaute certificates with spc fails - show unfinished private-key/certificate generations	2009-10-01
539050	Firefox	Security	nobody	UNCO	---	security#leurlstrict_origin_policy can prevent some desktop web apps from working	2011-03-12
541092	Firefox	Security	nobody	UNCO	---	Warning messages are not displayed on Forward / Back buttons	2013-10-11
541949	Firefox	Security	nobody	UNCO	---	trojan.downloader.js.agent.exe	2015-08-10
543194	Firefox	Security	nobody	UNCO	---	Firefox doesn't prompt offer personal certificate when accessing secure sites.	2010-01-30
545599	Firefox	Security	nobody	UNCO	---	Warn if a page that normally should be encrypted is not encrypted	2017-03-12

FIG. M – Aperçu de l’interface de *Bugzilla*.



## Glossaire

**Code informatique, code source** : Le code informatique, pris dans sa généralité, est une abstraction informatique qui se traduit sous la forme de langages structurants et performatifs. Il existe différents langages de programmation (Python, C++, Pearl, JavaScript...). Ces derniers sont un code commun qui permettent à l’opérateur humain de communiquer avec la machine. D’un côté, l’opérateur humain peut manier facilement les langages informatiques en ce qu’ils se rapprochent des langages usuels (syntaxe, noms, types, sémantique). De l’autre, la machine comprend de tels langages via un logiciel le traduisant en langage binaire (le compilateur et l’interpréteur), et in fine en électricité.

Le code source est l’ensemble des lignes de code, écrites en différents langages de programmation, qui composent un logiciel donné. Il est le plan architectural de l’objet numérique logiciel. Ce plan est inséparable de son implémentation, c’est-à-dire de sa compilation/interprétation dans la machine. Lorsque le code source se transforme en courant électrique, on dit qu’il est exécuté.

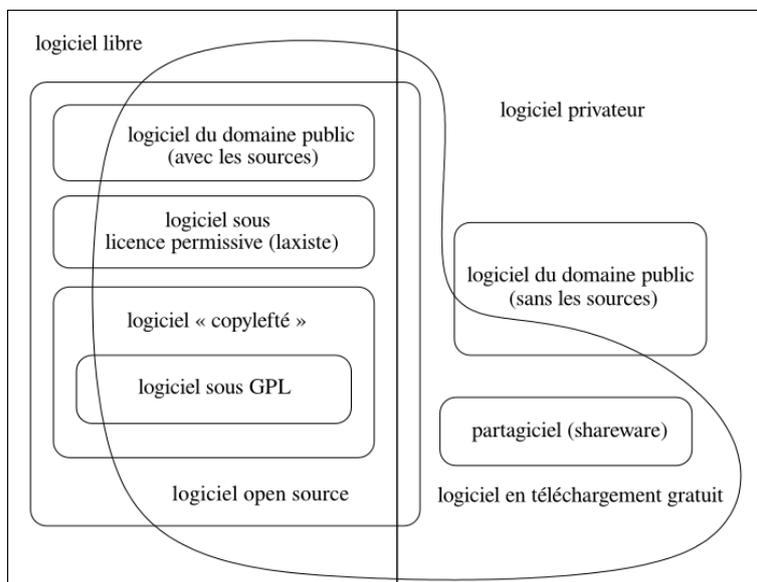
**HTML/ CSS** : *HyperText Markup Language* (HTML) est un langage informatique décrivant les contenus d’un document. C’est de telles descriptions de documents que

demande un navigateur auprès d’un serveur web et qu’il affiche dans un second temps. Les feuilles de style en cascade, ou *Cascading Style Sheets* (CSS), sont un langage informatique utilisé pour décrire la forme du rendu d’un document HTML et accompagnent presque systématiquement les documents HTML, afin d’indiquer au navigateur comment rendre le document (couleurs, tailles de police, etc...).

**HTTP** : *HyperText Transfer Protocol* (HTTP) est un protocole de communication entre terminaux mis en réseau pour former le web : c’est par cette forme de communication qu’un navigateur accède notamment à des contenus sur des serveurs distants.

**Logiciel (du point de vue fonctionnel)** : Le logiciel est une unité fonctionnelle présentant des séquences d’instructions (rédigées dans des langages de programmation) interprétables par une machine. Il est d’abord désigné sous le nom de *software*, par opposition avec le *hardware*, qui est la machine concrète sur laquelle il est implémenté (l’ordinateur par exemple). L’étymologie du mot français est notable : ce dernier est formé en 1969 à partir des mots « logique » et « matériel ». On voit déjà se dessiner dans le croisement de ces termes le statut hybride du logiciel, à la croisée du régime matériel et immatériel.

Le logiciel permettant les interactions de base avec la machine est le système d’exploitation (GNU/Linux, Apple, Windows). Le navigateur, exemple fil-rouge de notre travail, est un logiciel nodal pour les interactions de l’utilisateur avec le web.



Source : « Catégories de logiciels libres et non libres »,  
 Système d’exploitation GNU,  
<https://www.gnu.org/philosophy/categories.fr.html>  
 ( consulté le 10/05/17).

**Logiciel (du point de vue de la production) :** On oppose généralement le logiciel propriétaire au logiciel libre. Il s’agit de deux interprétations différentes du même objet technique numérique (la configuration technique fondamentale du logiciel n’est pas changée selon son mode de production). Le logiciel propriétaire et le logiciel libre sont les deux extrêmes de ce qui forme en réalité un continuum autour d’un critère : l’accès au code source. Ainsi, un logiciel propriétaire interdit l’accès à son code source, tandis qu’un logiciel libre permettra de le consulter, de le modifier, de le diffuser.

Le logiciel libre est une configuration de production

technique spécifique qui s’organise autour de solides bases théoriques et juridiques. Un des grands représentants en est Richard Stallman. Les quatre libertés définies par ce dernier (regroupant les libertés d’utilisation, de modification et de distribution) s’appuient sur un cadre juridique : celui des licences. La licence GNU, par exemple, utilise le « copyleft » : avec ce dernier, l’ouverture du code source est garantie grâce à une tournure restrictive ; les utilisateurs bénéficient de toutes les libertés, sauf de celle de fermer le code. De nombreuses communautés contribuent à produire des objets techniques ouverts et en constante évolution : elles sont représentées par des associations, comme la *Free Software Foundation* ou *April* en France.

**Navigateur** : Le navigateur est un logiciel permettant de visualiser des fichiers distants. Il est composé d’une interface utilisateur et d’un moteur de rendu. C’est une interface privilégiée entre l’utilisateur et le web.

**Numérique/informatique** : Historiquement, le numérique est un assemblage de logique et de technique. Pris dans ce sens très large, on peut faire remonter son apparition en 1642, avec l’invention de la machine à calculer de Pascal. Il comprend alors toutes les machines à calculer et sauts mathématiques (notamment l’algèbre binaire de Boole en 1847) jusqu’aux ordinateurs que nous connaissons, dont les pères fondateurs sont Turing et Von Neumann. Dans une perspective historique longue, le numérique se comprend comme la mutation progressive des machines à calculer en système technique total. Pourtant, le numérique ne renvoie pas qu’aux ordinateurs :

il existe des télévisions numériques, des appareils photographiques numériques. Dans son sens le plus large, le numérique est ce qui s’oppose au fonctionnement analogique.

L’informatique, en tant que champ d’étude des ordinateurs, est donc une sous-catégorie du numérique. L’informatique s’est constitué très progressivement, à partir de la seconde moitié du XX<sup>ème</sup> siècle, comme un champ d’étude cohérent – au fur de l’évolution de son objet d’étude. En effet, ce que nous nommons aujourd’hui « ordinateur » n’est désigné comme tel qu’en 1955 ; auparavant, l’informatique renvoyait à d’immenses machines à calculer occupant des pièces entières. Sous l’impulsion du complexe militaro-industriel lié à la recherche universitaire, les machines ont rapidement évolué jusqu’à occuper d’autres fonctions que celles de calculatrices géantes, et à s’installer dans les foyers (après avoir rétréci).

**Opposition *hardware/software* :** L’opposition entre *hardware* et *software* est utile pour des besoins de pédagogie, car elle clarifie les différentes entités composant l’ordinateur. Elle revêt également un sens au niveau de l’utilisation des machines, les métiers sur le *hardware* et ceux sur le *software* évoluant aujourd’hui de façon complètement autonome.

Malgré tout, la philosophie de la technique doit prendre acte du fait que cette dernière n’est pas si marquée qu’il n’y paraît. L’ordinateur procède de couches d’abstraction (*layers*) se rapprochant de plus en plus des circuits électroniques, et il est difficile de les différencier. Par ailleurs, une entité ne peut fonctionner pleinement qu’en rapport avec l’autre ; elles ne forment un tout fonctionnel

que lorsqu’elles sont prises ensembles.

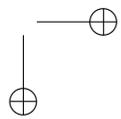
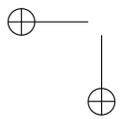
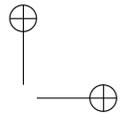
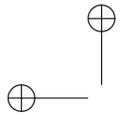
**Théorie de l’information** : La théorie de l’information est au départ une théorie mathématique, portée par Shannon dans les années 1940-1950. Elle est reprise par Wiener, qui entend généraliser son utilisation aux sciences sociales dans leur entier. Comprise en son sens le plus large, la théorie de l’information étudie les interactions entre individus au sein d’un système plus ou moins complexe. L’information peut alors se comprendre comme une mesure de l’organisation du système. L’échange d’information, ou de manière plus générale toute forme de communication (échanges de ressources), conduit les individus à réagir en modifiant leur comportement.

Simondon s’oppose à la conception de l’information de Wiener, car il considère que cette dernière tend à dresser une analogie entre le fonctionnement des êtres vivants et celui des machines. Il lui oppose sa théorie des machines à marge d’indétermination, qui consacre une collaboration (plutôt qu’une identité d’essence) entre l’homme et la machine.

**Web/ Internet** : Le web n’est pas un réservoir de pages web inerte. Il est un système d’étiquetage de pages accessibles sur des sites via le protocole HTTP et est soumis à des logiques tant politiques que techniques. Le web est tout ce qui est accessible par le protocole de communication HTTP. Les navigateurs permettent d’accéder à ces ressources, et les moteurs de recherche de les indexer (avec plus ou moins d’objectivité).

Le web n’est, de fait, qu’une application d’Internet. Comme son nom l’indique, Internet (« *net* », réseau en

anglais, et « inter ») est un méta-réseau englobant plusieurs réseaux différents. Il associe des ressources de télécommunication sur lesquels échangent des ordinateurs serveurs et clients, ainsi que d'autres périphériques (objets connectés). Cet échange est permis par l'utilisation d'un protocole commun qui permet l'acheminement de messages découpés en paquets indépendants : il s'agit du protocole TCP/IP. La gestion d'internet est (en principe) décentralisée en réseaux interconnectés.



## Bibliographie

- [1] « Couche d’abstraction matérielle ». Wikipédia : [https://fr.wikipedia.org/wiki/Couche\\_d%27abstraction\\_mat%C3%A9rielle](https://fr.wikipedia.org/wiki/Couche_d%27abstraction_mat%C3%A9rielle) (05/06/2017).
- [2] ARISTOTE (TRAD. H. CARTERON) : *Physique (I-VIII)*. 1973.
- [3] ARISTOTE (TRAD. J. TRICOT) : *Métaphysique*, volume 1 : livre A-Z. Vrin, 2000.
- [4] Bruno BACHIMONT : *Arts et sciences du numérique : ingénierie des connaissances et critique de la raison computationnelle*. Mémoire d’Habilitation à Diriger les Recherches, Université de Technologie de Compiègne, 2004.
- [5] Thierry BARDINI : *Bootstrapping : Douglas Engelbart, Coevolution, and the Origins of Personal Computing (Writing Science)*. Broché, 2000.
- [6] Jean-Hugues BARTHÉLÉMY : *Penser la connaissance et la technique après Simondon*, volume 2. Editions L’Harmattan, 2005.
- [7] Jean-Hugues BARTHÉLÉMY : « Deux points d’actualité de Simondon ». *Revue philosophique de la France et de l’étranger*, 131(3) :299–310, 2006.
- [8] Jean-Hugues BARTHÉLÉMY : *Technique, Communication et Société : A la recherche d’un modèle de gou-*

*vernance : Les technologies de l'information et de la communication et les limites du paradigme de la raison communicationnelle*, chapitre « Les significations philosophiques de l'ordinateur ou les enjeux épistémologiques du lien technique/société ». Presses universitaires de Namur, 2007.

- [9] Jean-Hugues BARTHÉLÉMY : « Du mort qui saisit le vif. Sur l'actualité de l'ontologie simondonienne ». *Appareil*, (2), 2008.
- [10] Jean-Hugues BARTHÉLÉMY : *Simondon ou l'encyclopédisme génétique*. Puf, 2008.
- [11] Jean-Hugues BARTHÉLÉMY : « Simondon et la question des âges de la technique ». *Appareil*, 2008.
- [12] Jean-Hugues BARTHÉLÉMY : *Cahiers Simondon*, n°1. Broché, 2009.
- [13] Jean-Hugues BARTHÉLÉMY : *Cahiers Simondon*, n°2. 2010.
- [14] Jean-Hugues BARTHÉLÉMY : *Cahiers Simondon*, n°3. 2011.
- [15] Jean-Hugues BARTHÉLÉMY : « Quel mode d'unité pour l'oeuvre de Simondon? ». In *Cahiers Simondon*, n° 3. Broché, 2011.
- [16] Jean-Hugues BARTHÉLÉMY : *Cahiers Simondon*, n°4. 2012.
- [17] Jean-Hugues BARTHÉLÉMY : « Sur l'architectonique du Mode d'existence des objets techniques ». In *Cahiers Simondon*, n°4. Broché, 2012.
- [18] Jean-Hugues BARTHÉLÉMY : *Simondon*. Les Belles Lettres, 2014.
- [19] Jean-Hugues BARTHÉLÉMY : « Glossaire Simondon : les 50 grandes entrées dans l'oeuvre »

- (05/06/2017). <https://appareil.revues.org/2253>, 2015.
- [20] Jean-Hugues BARTHÉLÉMY et Jean-Claude BEAUNE : *Penser l'individuation. Simondon et la philosophie de la nature*, volume 1. Editions L'Harmattan, 2005.
- [21] (dir.) Vincent BONTEMS : *Gilbert Simondon ou l'invention du futur*. 2016.
- [22] Vincent BONTEMS : « Encyclopédisme et crise de la culture ». *Revue philosophique de la France et de l'étranger*, 131(3), 2006.
- [23] Vincent BONTEMS : « Compte-rendu des activités de l'atelier Simondon ». In *Cahiers Simondon*, n° 1. Broché, 2009.
- [24] Vincent BONTEMS : « Du mode d'inexistence des mathématiques. L'ontologie du virtuel chez Simondon ». *Appareil*, (16), 2015.
- [25] Vincent BONTEMS et Vincent MINIER : « Simondon et l'innovation dans les machines spatiales de l'astronomie ». In *Gilbert Simondon ou l'invention du futur*, éditeur : Editions Klincksieck, 2016.
- [26] Dominique BOULLIER : *Sociologie du numérique*. Armand Colin, 2016.
- [27] Sébastien BOURBONNAIS : « Penser l'architecture numérique avec Simondon. Images et technologie ». In *Gilbert Simondon ou l'invention du futur*, éditeur : Editions Klincksieck, 2016.
- [28] Benjamin H. BRATTON : *The Stack. On Software and sovereignty*. MIT Press, 2016.
- [29] Philippe BRETON : *Une histoire de l'informatique*. La Découverte, 1990.

- [30] Sébastien BROCA : *Utopie du logiciel libre : du bricolage informatique à la réinvention sociale*. Le Passager clandestin, 2013.
- [31] Dominique CARDON : *A quoi rêvent les algorithmes : Nos vies à l'heure des big data*. Seuil, 2015.
- [32] Giovanni CARROZZINI : « Gilbert Simondon et Jacques Lafitte : les deux discours de la « culture technique » ». In *Cahiers Simondon*, n° 1. Broché, 2009.
- [33] Pascal CHABOT : *La philosophie de Simondon*. Vrin, 2003.
- [34] Valérie CHAROLLES : *Philosophie de l'écran : dans le monde de la caverne ? Essais*. Fayard., 2013.
- [35] Jean-Yves CHATEAU : « Technophobie et optimisme technologique modernes et contemporains ». In *Gilbert Simondon, une pensée de l'individuation et de la technique*. Albin Michel, 1994.
- [36] Yves CITTON : « Notre inconscient numérique ». *La revue du Crieur*, 2016.
- [37] Timothy R. COLBURN : « Software, Abstraction, and Ontology ». *The Monist*, 1999.
- [38] Muriel COMBES : *Individu et collectivité*. Puf, 1999.
- [39] Muriel COMBES : « L'acte fou ». *Multitudes*, 18, 2004.
- [40] François DAGOGNET : *Rematérialiser*. Vrin, 1989.
- [41] François DAGOGNET : *L'invention de notre monde : l'industrie, pourquoi et comment ?* Encre Marine. Les Belles Lettres., 1995.
- [42] Yves DEFORGE : *Technologie et génétique de l'objet industriel*. Maloine, 1985.

- [43] Milad DOUEIHI : *Pour un humanisme numérique*. Seuil, 2011.
- [44] Ludovic DUHEM : « Penser le numérique avec Simondon ». [http://www.academia.edu/9024613/Penser\\_le\\_num%C3%A9rique\\_avec\\_Simondon\\_Thinking\\_the\\_digital\\_with\\_Simondon\\_\(05/06/2017\)](http://www.academia.edu/9024613/Penser_le_num%C3%A9rique_avec_Simondon_Thinking_the_digital_with_Simondon_(05/06/2017)).
- [45] Ludovic DUHEM : « Simondon et le langage ». *Appareil*, (16), 2015.
- [46] Ludovic DUHEM : « La réticulation du monde. Simondon penseur des réseaux ». In *Gilbert Simondon ou l'invention du futur*, éditeur : *Editions Klincksieck*, 2016.
- [47] William H DUTTON : *The Oxford handbook of internet studies*. Oxford University Press, 2013.
- [48] Douglas ENGELBART : « Augmenting Human Intellect : A Conceptual Framework ». Summary Report AFOSR-3223 under Contract AF 49 (638)-1024, SRI Project 3578 for Air Force Office of Scientific Research. *Stanford Research Institute*, 1962.
- [49] Andrew FEENBERG : *(Re)penser la technique. Vers une technologie démocratique*. La Découverte, 2004.
- [50] Andrew FEENBERG : *Pour une théorie critique de la technique*. Lux, Montréal, 2014.
- [51] James H. FETZER : « Program Verification : The Very Idea ». *Communication of the ACM*, 1988.
- [52] Michel FOUCAULT : *L'archéologie du savoir*. Gallimard, Paris, 1969.
- [53] Michel FOUCAULT : *L'ordre du discours*. Gallimard, Paris, 1971.

- [54] Georges FRIEDMANN : *Sept études sur l’homme et la technique*. Gonthier, 1966.
- [55] Maurizio GABBRIELLI et Simone MARTINI : *Programming Languages : Principles and Paradigms*. Springer, 2010.
- [56] Tali GARSIEL et Paul IRISH : « How Browsers Work : Behind the scenes of modern web browsers ». <https://www.html5rocks.com/en/tutorials/internals/howbrowserswork/> (05/06/2017), 2011.
- [57] Thierry GAUDIN : « Un regard d’ingénieur sur Simondon ». In *Gilbert Simondon ou l’invention du futur*, éditeur : *Editions Klincksieck*, 2016.
- [58] Jérémy GROSMAN : « Simondon et l’informatique II ». In *Gilbert Simondon ou l’invention du futur*, éditeur : *Editions Klincksieck*, 2016.
- [59] Xavier GUCHET : « Simondon, la technologie et les sciences sociales ». In *Cahiers Simondon, n° 1*. Broché, 2009.
- [60] Eric GUICHARD : *Comprendre les usages de l’Internet*. Editions rue d’Ulm, Paris, 2001.
- [61] Eric GUICHARD : « Culture numérique, culture de l’écrit ». *Interfaces numériques*, 4, 2015.
- [62] Martin HEIDEGGER : *Essais et conférences. La question de la technique*. Gallimard, 1958.
- [63] Pekka HIMANEN : *L’éthique hacker*. Paris : Exils, 2001.
- [64] Gilbert HOTTOIS : *Simondon et la philosophie de la « culture technique »*. De Boeck-Wesmael Université, 1994.
- [65] Daniel INCE : *The Computer : A Very Short Introduction*. Oxford, 2011.

- [66] Dominique JANICAUD : *La puissance du rationnel*. Broché, 1985.
- [67] KANT : *Critique de la faculté de juger*. GF, 2015.
- [68] Christopher M KELTY : *Two bits : The cultural significance of free software*. Duke University Press, 2008.
- [69] Michael KURTOV : « Simondon et l’informatique III. L’évolution des langages de programmation à la lumière de l’allagmatique ». In *Gilbert Simondon ou l’invention du futur*, éditeur : *Editions Klincksieck*, 2016.
- [70] Bruno LATOUR : « Prendre le pli des techniques ». *Réseaux*, 163, 2010.
- [71] Emmanuel LAZARD et Pierre MOUNIER-KUHN : *Histoire illustrée de l’informatique*. EDP Sciences, 2016.
- [72] Ronan LE ROUX : « Simondon technologue, une individuation inachevée - Dialogues (et silences) avec les théories des machines dans la France de l’après-guerre ». In *Cahiers Simondon, n 3*. Broché, 2011.
- [73] Claude LÉVI-STRAUSS : *La pensée sauvage*. Plon, 1962.
- [74] Herbert MARCUSE : *L’homme unidimensionnel*. Minuit, Paris, 1968.
- [75] Karl MARX : *Oeuvres de Karl Marx*, chapitre Le Capital. Gallimard, 1968.
- [76] Karl MARX : *Oeuvres de Karl Marx*, chapitre Le travail aliéné, pages 62–63. Numéro II. 1968.
- [77] Eben MOGLEN : « L’anarchisme triomphant : le logiciel libre et la mort du copyright ». *Multitudes*, (5), 2001.

- [78] Alexandre MONNIN : *Vers une philosophie du Web : le Web comme devenir-artefact de la philosophie (entre URIs, tags, ontologie (s) et ressources)*. Thèse de doctorat, Université Panthéon-Sorbonne-Paris I, 2013.
- [79] James MOOR : « Three Myths of Computer Science ». *The British Journal for the Philosophy of Science*, 1978.
- [80] Serge MOSCOVICI : *Psychosociologie*. Puf, 1984.
- [81] Noam NISAN et Shimon SCHOCKEN : *The Elements of Computing Systems - Building a Modern Computer from First Principles*. The MIT Press, 2008, [2005].
- [82] Tristan NITOT : « A propos de bidouillabilité ». <http://standblog.org/blog/post/2009/09/16/A-propos-de-bidouillabilit%C3%A9> (05/06/2017), 2009.
- [83] Den ODELL : *Pro JavaScript RIA Techniques*. Apress, 2009.
- [84] Frédéric PASCAL : « Gilbert Simondon et l’informatique I ». In *Gilbert Simondon ou l’invention du futur*, éditeur : Editions Klincksieck, 2016.
- [85] Eric S RAYMOND : *La cathédrale et le bazar*. 1998.
- [86] Pablo Esteban RODRIGUEZ : « L’information entre Foucault, Deleuze et Simondon ». In *Gilbert Simondon ou l’invention du futur*, éditeur : Editions Klincksieck, 2016.
- [87] László ROPOLYI : *Philosophy of the Internet : A Discourse on the Nature of the Internet*. Thèse de doctorat, 2013.
- [88] Jean-Pierre SERIS : *La technique*. Puf, 2013.
- [89] Gilbert SIMONDON : *L’invention dans les techniques*. Seuil, 2005.

- [90] Gilbert SIMONDON : *Du mode d’existence des objets techniques*. Aubier, 2012 [1958].
- [91] Gilbert SIMONDON : *L’individuation à la lumière des notions de forme et d’information*. Millon, 2013 (2005).
- [92] Gilbert SIMONDON : *Imagination et invention*. Puf, 2014.
- [93] Gilbert SIMONDON : *Sur la technique*. Puf, 2014.
- [94] Gilbert SIMONDON : *Sur la technique*, chapitre « L’effet de halo en matière technique : vers une stratégie de la publicité (1960) ». Puf, 2014.
- [95] Gilbert SIMONDON : *Sur la technique*, chapitre « Psychosociologie de la technicité ». 2014, 2014.
- [96] Gilbert SIMONDON : *Sur la technique*, chapitre « Psychosociologie du cinéma ». Puf, 2014.
- [97] Gilbert SIMONDON : *Communication et information*. Puf, 2015.
- [98] Gilbert SIMONDON : *Sur la psychologie*. Puf, 2015.
- [99] Gilbert SIMONDON : *Sur la philosophie (1950-1980)*. Puf, 2016.
- [100] Etienne SOURIAU : *Les différents modes d’existence, suivi Du mode d’existence de l’oeuvre à faire*. Puf, 2009 [1934].
- [101] Richard STALLMAN : *Free software, free society : Selected essays of Richard M. Stallman*. GNU Press, 2002.
- [102] Richard STALLMAN, Sam WILLIAMS et Christophe MASUTTI : *Richard Stallman et la révolution du logiciel libre : Une biographie autorisée*. Eyrolles, 2013.
- [103] Fred TURNER : *Aux sources de l’utopie numérique : De la contre-culture à la cyberculture*. C&F éditions, 2012.

- [104] Raymond TURNER et Nicola ANGIUS : « The Philosophy of Computer Science ». *The Stanford Encyclopedia of Philosophy*, <https://plato.stanford.edu/entries/computer-science/> (Printemps 2017).
- [105] Stéphane VIAL : *La structure de la révolution numérique : philosophie de la technologie*. Thèse de doctorat, Université René Descartes-Paris V, 2012.
- [106] Norbert WIENER : *Cybernetics : or Controle and Communication in the Animal and the Machine*. MIT Press, 2nde édition, 1965.