Pour la reconstruction des processus d'écriture numériques de Derrida grâce à la computer forensics : reconstruction des données et matérialité numérique historique

Thorsten Ries

Cet article est issu d'une conférence intitulée : *Pour la reconstruction des processus d'écriture numériques de Derrida grâce à la « computer forensics » : reconstruction des données et matérialité numérique historique* qui a eu lieu le 19 octobre 2018 à Paris, pour le 50e anniversaire de l'Institut des textes et manuscrits modernes (ITEM) dans la demi-journée consacrée au volet de la critique génétique appliquée aux supports numériques.

Pour la reconstruction des processus d'écriture numériques de Derrida grâce à la *computer forensics* : reconstruction des données et matérialité numérique historique

Thorsten Ries

Traduit de l'allemand par Yannick Souladié

1 Introduction

Comme vient de le montrer la contribution précédente et comme l'avaient déjà signalé Aurèle Crasson, Jean-Louis Lebrave et Jérémy Pedrazzi dans leur étude « Le "siliscrit" de Jacques Derrida. Exploration d'une archive nativement numérique[1], [2] », l'Institut Mémoires de l'édition contemporaine (IMEC) à Caen a réalisé des images forensiques complètes et pérennes des disques durs de Jacques Derrida, ainsi que des supports de stockage amovibles recelant des traces de la production textuelle du philosophe, qui peuvent être analysées et restaurées avec des moyens d'analyse numérique forensique.

Cet article n'entend pas seulement contribuer aux recherches philologiques sur Derrida, mais également à la codicologie historique et forensique de l'informatique, en présentant les résultats, les méthodes et les aspects forensiques approfondis de la matérialité historique numérique de ces supports de sto-

[1] Cet article et les recherches dont il est issu ont été financés par l'ITEM/CNS, les actions Marie Skłodowska-Curie (MSCA-IF) dans le cadre du programme *Horizon 2020* de la Commission Européenne (Projet : *Digital Forensics in the Historical Humanities (DfitHH) : Hanif Kureishi, Glyn Moody, MOPA*) et par la Fondation pour la Recherche Flammande (Fonds Wetenschappelijk Onderzoek, FWO ; projet : *Hard Drive Philology / Source Code Philology : Tracing the digital writing and coding process in German literature*).

[2] Aurèle Crasson, Jean-Louis Lebrave et Jérémy Pedrazzi, « Le "siliscrit" de Jacques Derrida. Exploration d'une archive nativement numérique », *Genesis*, 49, 2019, p.7-12. URL : https://journals.openedition.org/genesis/4316 (consulté le 02/02/21).

ckage. Nous tenons à souligner ici que cette étude n'aurait pas pu être réalisée sans le concours amical de l'IMEC et de l'héritière de Jacques Derrida, qui nous ont permis d'accéder aux données originales, et de la composante de l'IMEC responsable de la conservation numérique, qui a généré des images forensiques exactes des supports de stockage, destinées à l'archivage à long terme et à l'analyse scientifique (*bitstream-preserving images*).

2 Analyse de fichiers : le logiciel *MacWrite*

La première analyse exploratoire de Crasson, Lebrave et Pedrazzi a déjà montré que le flux de données *binaire* des documents MacWrite et MacWrite Pro créées par Derrida pouvait contenir des variantes supprimées ainsi que différents états génétiques du processus d'écriture[3]. De toute évidence, le nouveau programme de traitement de texte intégré ensuite à *Apple* et ultérieurement développé par *Claris* utilise un procédé de mémorisation incrémental – similaire au *Fast Save Feature (Complex Documents)* utilisé dans les premières versions de *Microsoft Word* –, dans lequel une série de modifications successives du texte est enregistrée comme un segment dans le flux de données du fichier[4]. Ce procédé de mémorisation n'a pas simplement pour conséquence l'accélération de l'enregistrement sur les supports de données lents, il peut également aboutir à la conservation des parties supprimées – ce qui est intéressant dans l'optique de la forensique informatique –. Jérémy Pedrazzi a déjà pu montrer, au moyen de *libmwaw*[5], bibliothèque de conversion programmée par Laurent Alonso pour le traitement de texte open source *LibreOffice*, que l'application *MacWrite* historique sauvegardait ces enregistrements incrémentaux par blocs de taille fixe de 256 caractères, sans forcément enregistrer les données de manière chronologiquement séquentielle (ce qui fait que la capacité de ces blocs de 256 caractères n'est pas toujours utilisée entièrement). L'analyse menée par Pedrazzi indique déjà que l'étude approfondie du processus de sauvegarde de l'installation originale de *MacWrite,* dans la séquence des blocs dans le courant de données, était susceptible de fournir des résultats pour la reconstruction

3 Note 1, paragraphe 22-27.

4 En complément des éditeurs hexadécimaux, l'application open source *Hachoir* est particulièrement adaptée pour le parsing et l'analyse structurelle des documents *Word* binaires (.doc). URL : https://hachoir.readthedocs.io/en/latest/ (consulté le 02/02/21), URL : https://github.com/vstinner/hachoir (consulté le 02/02/21).

5 Code source du site Libmwaw : URL : http://sourceforge.net/p/libmwaw/_list/git (consulté le 02/02/21). Depuis la ligne de commande, *Libmwaw* peut également être utilisé comme analyseur de fichiers binaires pour les fichiers *MacWrite*, en le compilant avec l'option « – enable-full-debug ».

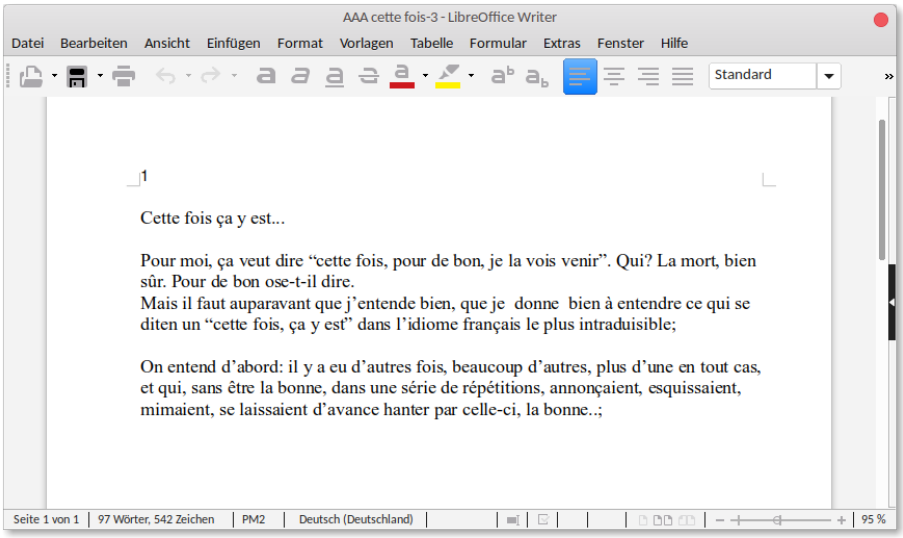

IMAGE 1 – « *AAA cette fois* », aperçu du traitement de texte

génétique du texte[6]. Il apparaît important de noter que *libmwaw* ne parse que les éléments non supprimés des flux de données et met par cela en évidence la structure du document – les parties supprimées du fichier ne sont pas parsées et ne sont visibles qu'à l'aide d'un éditeur hexadécimal, comme par exemple *wxHexEditor*[7].

2.1 AAA cette fois

En nous appuyant sur ces artefacts de données, nous avons réalisé un court spécimen de reconstruction textuelle génétique à partir de trois fichiers. Le fichier s'appelle « *AAA cette fois* ». L'image 1 offre une vue du traitement de texte de la dernière version. Ces trois fichiers *MacWrite Pro 1.5*, présents sur le disque dur (de marque *Conner*) intégré à l'ordinateur *Mac Classic* de Derrida et sur deux supports de données de sauvegarde *Syquest*[8], contiennent non

6 Note 1, paragraphe 25-27. Ceci serait d'autant plus intéressant que le séquençage génétique des artéfacts de *Fastsave* dans *Microsoft Word* repose essentiellement sur une reconstruction philologique approximative, car la séquence du flux de données n'est pas chronologique et n'est plus active et les parties « supprimées » du document ne contiennent plus de métadonnées.

7 Code source du site *WxHexEditor* : URL : https://www.wxhexeditor.org/ (consulté le 02/02/21).

8 Il existe trois versions différentes. Au total, on retrouve quatre doublons de la valeur de hachage de ces trois fichiers sur un disque dur, deux CD-ROM et une disquette *ZIP*. En outre, on trouve des fichiers temporaires avec le préfixe « ._ » sur le disque dur de marque *Conner* et sur les deux disques *Syquest*, qui ne contiennent pas de textes de Derrida, mais qui nous informent du fait que ces fichiers ont potentiellement pu être directement ouverts et modifiés sur ces supports.

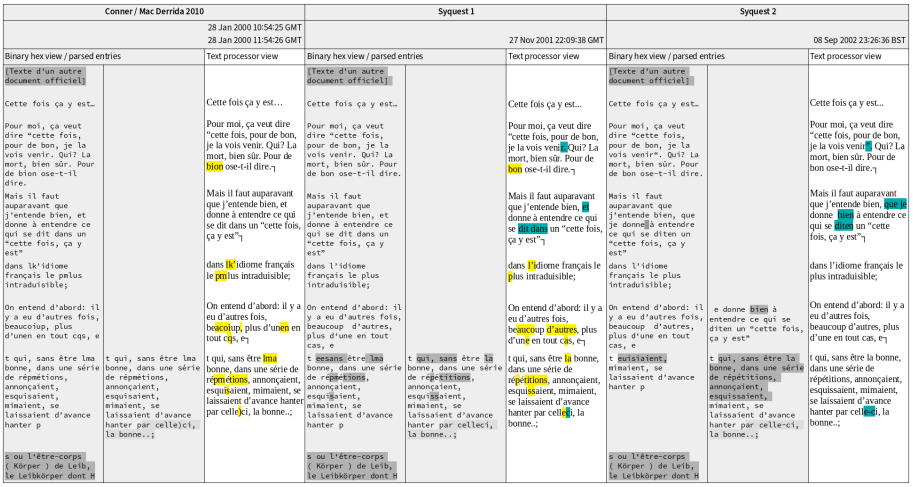

IMAGE 2 – Tableau synoptique des fragments du flux de données et du texte des trois fichiers « *AAA cette fois* » non identiques s'affichant dans le traitement de texte

seulement trois versions différentes de ce texte court, mais aussi, dans le flux de données binaire, des variantes cachées et des passages vraisemblablement supprimés, ainsi que des textes issus d'un autre fichier.

L'image 2 nous offre un tableau synoptique des trois fichiers et des fragments de leurs flux de données, sur lesquels on a probablement travaillé entre janvier 2000 et septembre 2002.[9] Si vous comparez les trois versions du texte affichées dans le traitement de texte normal (colonnes sur fond blanc), vous constaterez qu'elles présentent des variantes rédactionnelles. En outre, on remarque que les deux pans de texte, présents sur les supports de données *Syquest* dans le flux de données, présentent des variantes d'édition qui n'apparaissent plus dans le traitement de texte, parce que Derrida les a effacés dans le document. On peut comparer les fragments de texte dans le flux de données à la version précédente, faisant ainsi apparaître une série génétique de variantes textuelles (voir le tableau synoptique). D'un point de vue technique, nous avons ici affaire à une variante, jusqu'à présent non documentée dans la littérature forensique numérique disponible à l'heure actuelle, de la mémoire incrémentale d'artéfacts de fichiers de *MacWrite Pro*, qui est similaire au mécanisme *Fast Save* des anciennes versions de *Microsoft Word*.

De plus, le flux de données de ces fichiers contient des fragments de texte qui ne semblent pas appartenir au texte du document. D'une part, au début du

9 À condition de supposer, avec toute la prudence nécessaire, que l'heure du système était correctement réglée. Il faudrait le vérifier en procédant à une analyse plus complète du système.

flux de données, on trouve un fragment de texte d'un document administratif, que nous ne reproduisons pas ici. D'autre part, on trouve un pan de texte qui semble thématiquement correspondre au texte *AAA cette fois*. L'allusion aux cours de Husserl et son concept de « *Leibkörper* » (que Derrida traduit par « être-corps ») semblent correspondre à la réflexion sur le paradoxe de la prétendue individualité et du prétendu caractère non reproductible de la mort dans *AAA cette fois* : « s ou l'être-corps (Körper) de Leib, le Leibkörper dont H[10] ». Il semble pour le moins possible que ce deuxième pan de texte ait fait partie du document et ait été supprimé, ou bien, qu'il ait temporairement été stocké dans le presse-papiers ou dans la mémoire vive de l'ordinateur et ait par-là même été enregistré[11]. Dans ce cas, il faudrait considérer que ce fragment de texte apartient au contexte textuel génétique immédiat de *AAA cette fois*.

2.2 S, SE, SEC, SECRE, SECRET, SECRETA …

Les dossiers conservés dans les archives de l'IMEC, qui documentent la progression de la rédaction du séminaire de Derrida *Répondre – du secret* en 1991-1992, suivent le modèle ludique de l'ajout incrémental : « S », « SE », « SEC » … « SECRE », jusqu'à ce que « SECRET » apparaisse dans ces noms de fichiers ; la série s'achevant avec le jeu de mots « SECRETARIAT « [12] . Il s'agit de nombreux fichiers, de versions différentes de fichiers et de doublons de fichiers, disséminés sur plusieurs supports de données. Les fichiers sont enregistrés sous le format *MacWrite 4.5-5.01*, ce qui constitue une difficulté pour l'analyse textuelle génétique et forensique numérique. Contrairement à celui des fichiers de *AAA cette fois*, le flux de données des fichiers de *Répondre – du secret* n'est pas enregistré en texte clair, mais compressé par un mécanisme

Les horaires consignés des dernières modifications de fichiers sont : le 28 janvier 2000 à 11 :54 :26 GMT, le 27 novembre 2001 à 22 :09 :38 GMT et le 8 septembre 2002 à 23 :26 :36 BST.

10 Voir le cours d'Edmund Husserl : « Hauptstücke aus der Phänomenologie und Kritik der Vernunft », deuxième semestre 1907, *in* Edmund Husserl, *Ding und Raum. Vorlesungen 1907*, Hrsg von Karl-Heinz Hahnengress, Smail Rapic, Text nach Husserliana, Bd. XVI, Hamburg, Felix Meiner Verlag 1991. Se référer plus précisément au § 83 : « Sich Bewegen und Bewegtwerden des Leibes. Grenzen der kinästhetischen Konstitution des Leibkörpers », pp. 278-284.

11 De tels artefacts ne sont pas inhabituels dans les applications *Office* de cette époque. Il faudrait cependant vérifier, à l'aide du logiciel originel, si cette version du programme incluait effectivement ces procédures de stockage.

12 Le titre du séminaire est « Répondre – du secret ». Derrida donna ce cours à l'École des Hautes Études en Sciences Sociales (EHESS) à Paris du 13 novembre 1991 au 8 avril 1992. Voir Jacques Derrida, « Responding To/Answering For : The Secret (2nd session, November 27, 1991) », translated, edited by Kevin Newmark, in *French Studies*, 2014, No. 125/126, *Time for Baudelaire (Poetry, Theory, History)*, Yale, pp. 7-29. Se référer ici à la page 7.

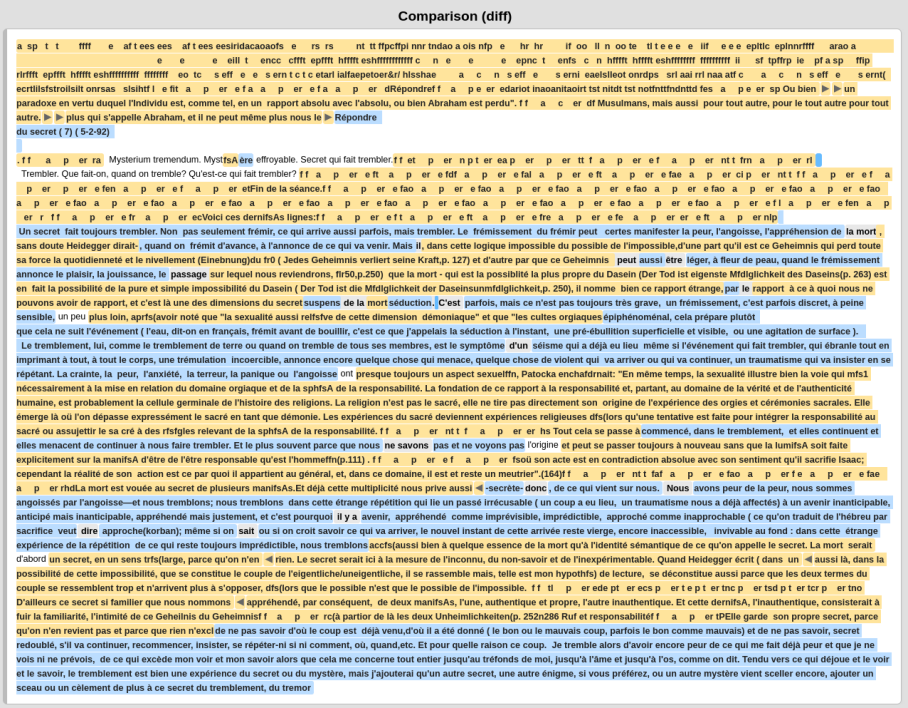

IMAGE 3 – Vue synoptique en surimpression des différences textuelles du passage initial du flux de données décompressé et du texte du fichier SECRETA affiché dans le traitement de texte. Les passages qui ne sont pas en couleur sont identiques, ceux surlignés en jaune ont été rayés (mais sont encore présents dans le flux de données), ceux surlignés en bleu ont été ajoutés plus tard.

propriétaire[13]. Grâce à la bibliothèque *libmwaw*[14], *LibreOffice* peut ouvrir les fichiers sans problème – toutefois, comme la bibliothèque ne parse que le texte qui n'a pas été supprimé, les versions et variantes supprimées demeurent inaccessibles dans le flux de données du document.

Pour rendre ces différentes phases de rédaction et ces variantes visibles, une petite opération de réingénierie était nécessaire. En suivant les indications

13 Voir URL : http://fileformats.archiveteam.org/wiki/MacWrite (consulté le 02/02/21) : « *MacWrite 2.2* a stocké du texte dans un schéma de compression où les caractères (spécifiques à la langue) les plus courants (pour l'anglais, « etnroaisdlhcfp », dans cet ordre, sans oublier l'espace au début) ont tous été stockés comme un nibble (un demi-octet), dans lequel les valeurs 0 à E correspondaient aux caractères de cette liste. La valeur de nibble F signalait que le caractère suivant était différent, ce qui signifie que les caractères qui ne figuraient pas sur la liste occupaient un espace de stockage de trois nibbles (un octet et demi). Il en résultait généralement des économies d'espace, dans la mesure où les caractères les plus courants composent généralement une grande partie du texte. *MacWrite 3.x* et ses versions suivantes utilisaient un système de compression différent et ont été conçus de façon à ce que les fichiers puissent être modifiés directement sur le disque sans avoir à charger le fichier entier dans la mémoire. »

14 Note 5.

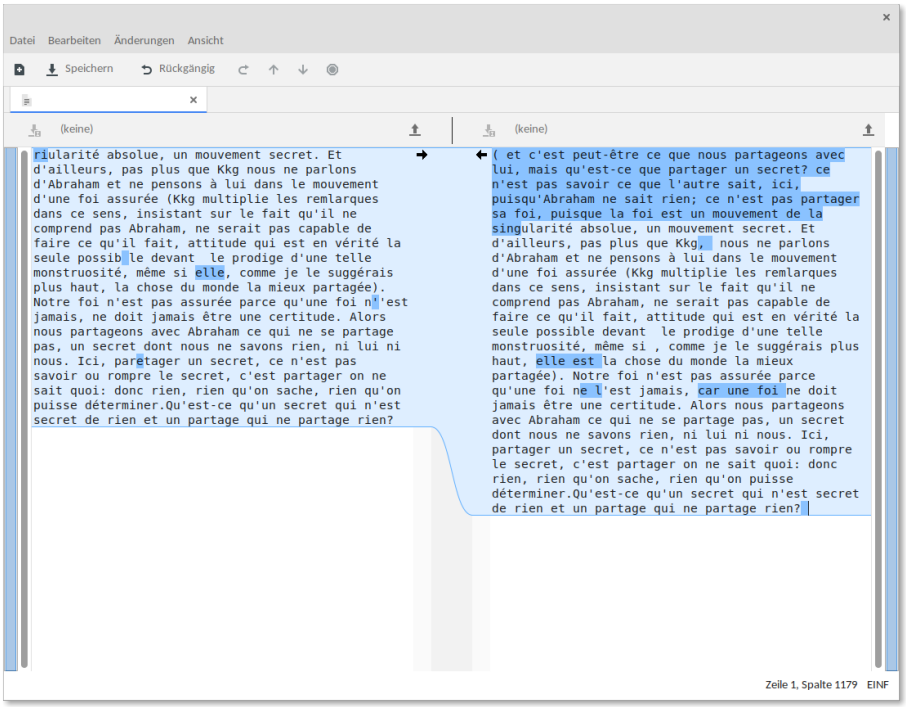

IMAGE 4 – Vue synoptique d'un passage issu d'un fragment dans le flux de données décompressé (à gauche) et du texte affiché dans le traitement de texte (à droite) du fichier « SECRETA ». Les passages surlignés en bleu foncé signalent les modifications (ou les raccords de texte).

de Laurent Alonso sur l'algorithme de compression[15] relativement simple utilisé par *Apple* pour *MacWrite* à cette époque, il a été possible d'écrire un script *Python* simple décompressant le flux de données des fichiers comme un tout (et pas seulement les parties non supprimées). Il est ainsi apparu qu'*Apple* avait déjà utilisé le mécanisme d'enregistrement incrémental dans cette ancienne version du traitement de texte *MacWrite*, ce qui a rendu possible la conservation de variantes et de versions supprimées dans le flux de données. Les images 3 et 4 offrent une vision synoptique, permettant de comparer deux passages du flux de données décompressé avec le texte du document « SECRETA » affiché dans le traitement de texte ; les modifications opérées dans le texte étant surlignées en couleurs.

Grâce à ce script de décompression simple, il a également été possible de décompresser et de restituer sous forme lisible le flux de données de tous les fichiers et fragments de fichiers *MacWrite* supprimés et non écrasés dans

15 Note 13.

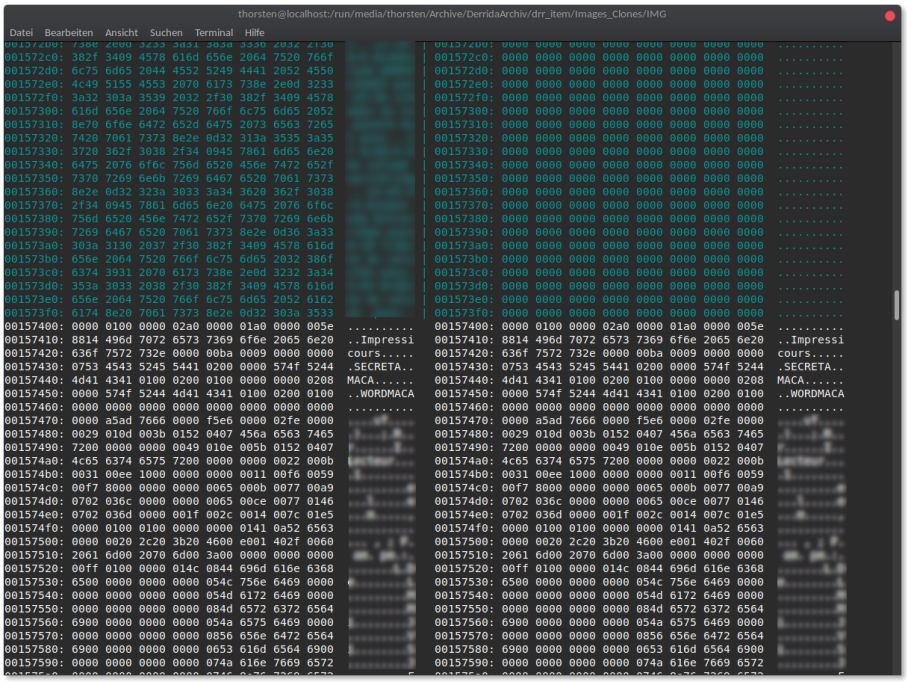

IMAGE 5 – Vue synoptique d'une section de l'image de données forensique avec *Colordiff* : en haut à gauche l'image forensique dans son état originel et en haut à droite l'image avec une zone « découpée ». Dans la partie inférieure de l'image, on peut apercevoir le début d'une instance supprimée du fichier SECRETA.

la zone non allouée de l'image forensique du disque dur. En des termes plus simples, la décompression de tous les flux de données compressés sur l'image du disque dur, effectuée suivant cette méthode appliquée à *MacWrite*, a permis de reconstituer des phases de rédaction issus de fichiers supprimés, que même ce que l'on appelle un *File Carver* n'aurait normalement pas pu reconstituer.

3 Restauration de fichiers et analyse du système de fichiers

Les flux de données des fichiers ne sont cependant pas les seuls à présenter un intérêt dans le cadre d'une analyse numérique forensique et textuelle génétique d'un support de données. La restauration de fichiers et de fragments de fichiers supprimés et n'ayant pas été écrasés – par exemple les versions des documents éliminées par l'auteur ou les fichiers temporaires du traitement

de texte – sont de la plus haute importance du point de vue de la génétique textuelle.

3.1 Restauration de données

Pour pouvoir reconstituer de telles données, il est généralement nécessaire de rechercher, dans la zone dite « non allouée[16] » du support de données, des fichiers et fragments de fichiers supprimés et restaurables, avec, par exemple, un outil comme *TestDisk* ou un *File Carver* comme *Foremost*, *Scalpel* ou *Photorec*[17].

L'analyse des disques durs de Jacques Derrida et de d'autres supports de données pose quelques problèmes supplémentaires qui sont spécifiques à la forensique numérique historique. L'un de ces problèmes réside dans le fait que *TestDisk* gère très bien le nouveau système de fichiers *HFS+* d'*Apple*, mais ne gère pas le système de fichiers *HFS* plus ancien utilisé dans les supports de données de Derrida. À cela s'ajoute le fait que les *File Carver* comme *Foremost*, *Scalpel* et *Photorec*, recherchent ce que l'on appelle des « signatures » de formats de fichiers connus – c'est-à-dire des débuts (en-têtes) et des fins (pieds de page) - typiques. Mais, en règle générale, les formats *MacWrite* obsolètes ne font pas partie du catalogue de signatures des outils. Si l'on a recours à l'astuce consistant à ajouter une signature *MacWrite* au *File Carver Foremost*, on se trouve confronté à une autre curiosité propre à l'histoire des applications informatiques : les fichiers *MacWrite* n'ont pas de pied de page, la taille du flux de données est définie au début du document – mais il n'y a pas de point d'arrêt susceptible de faire office de signature. Il faut par conséquent définir pour *Foremost* qu'il n'y a que l'en-tête et que tous les fichiers, dont l'en-tête aura été identifié, ont la même taille (par exemple 1 Mo). Grâce à cette méthode, de nombreux fichiers *MacWrite* supprimés des disques durs de Derrida ont pu être restaurés intégralement, dans une forme intacte et fonctionnelle, toutefois la longueur de ces fichiers n'est jamais authentique[18].

16 Précisons que la « zone non allouée » d'un support de données désigne les blocs « vides » du système de fichiers qui, du point de vue de ce système, peuvent être utilisés pour écrire des données. Lorsqu'un fichier est supprimé, les blocs dans lesquels il est conservé sont balisés comme réinscriptibles et l'entrée est supprimée ou déplacée dans la « corbeille » dans l'arborescence du système de fichiers. La zone non allouée n'est ainsi pas réellement vide : elle peut contenir des fichiers, des versions de fichiers et des fragments supprimés que l'on peut récupérer. Voir aussi « Drive Slack ».

17 Ces outils forensiques sont des applications open source : *TestDisk* et *Photorec* URL : https://www.cgsecurity.org/, (consulté le 02/02/21), *Scalpel* URL : https://github.com/sleuthkit/scalpel, (consulté le 02/02/21), *Foremost* URL : http://foremost.sourceforge.net/, (consulté le 02/02/21).

18 Sur ce problème, voir Fred Cohen, « Column Putting the Science in Digital Forensics », *Journal of Digital Forensics, Security and Law*, Vol. 6.1, 2011, pp. 7-14.

En outre, il apparaît aussi nécessaire de chercher dans la zone non allouée des supports de données, afin de trouver les fragments de documents et les fichiers temporaires, que le *File Carver* n'aura pas trouvés. C'est le cas notamment lorsque seul l'en-tête du fichier a été écrasée ou lorsque le fichier a été sauvegardé dans des fragments très éloignés les uns des autres. Le processus de décompression de tous les flux de données compressés par *MacWrite* sur ces supports de données (ou dans leur zone non allouée), que nous avons décrit au point 2.2, s'avère particulièrement utile pour cette étape, en ce qu'il convertit aussi de manière automatique en texte clair les flux de données de fichiers fragmentés ne possédant pas d'en-tête.

3.2 Système de fichiers et zone non allouée

Malheureusement, le *File Carver* ne fait pas de distinction entre les fichiers non supprimés et les fichiers supprimés. Il reconstruit toutes les données pour lesquelles il parvient à trouver un en-tête ou un pied de page. Pour cette raison, la reconstruction ou l'analyse de données brutes avec un *File Carver* doit être précédée d'une étape supplémentaire : la séparation de la mémoire non allouée. Cette séparation est généralement effectuée à l'aide de l'outil *BLKLS*, qui regroupe les blocs de mémoire non occupés dans un flux de données et génère ainsi une image exclusive de la zone non allouée[19]. Malheureusement, *BLKLS* ne peut pas travailler avec le système de fichiers *HFS* obsolète de l'ordinateur de Derrida, qui à l'époque n'était exploitable que par l'intermédiaire de *SCSI*, et il n'existe aucun outil capable d'assurer cette tâche pour cet ancien système de fichiers.

Ainsi, pour être certains que seuls les fichiers supprimés seront restaurés lors du processus de restauration de données initié par le *File Carver*, il nous faut emprunter une autre voie qui ne figure pas dans les manuels de forensique numérique : au lieu de séparer la zone non allouée, on peut intégrer dans le système d'analyse une copie de l'image forensique avec des droits d'écriture et utiliser l'outil *Bleachbit* pour écraser l'ensemble des fichiers visibles et les remplacer par des « 0[20] ». Après ce « découpage », seules demeurent, d'une part la zone non allouée et d'autre part les zones remplies de « 0 ».

Cependant, même en utilisant cette solution, on aboutit à un nouveau problème d'ordre forensique numérique : les systèmes de fichiers *HFS* obsolètes ne peuvent être intégrés qu'en lecture seule sur les systèmes *Linux* modernes (*read-only, ro*), en raison de la conception-même des pilotes de

19 *BLKLS*, URL : http://www.sleuthkit.org/sleuthkit/man/blkls.html, (consulté le 02/02/21).
20 *Bleachbit*, URL : https://www.bleachbit.org/, (consulté le 02/02/21).

ces systèmes de fichiers obsolètes. Il existe cependant une distribution *Linux* orientée forensique, unique en son genre, qui permet de monter une copie des images forensiques des systèmes de fichiers *HFS* avec un droit d'écriture : *Parrot Linux*[21]. Grâce à ce procédé, on peut sélectionner, en vue d'une restauration, des données supprimées ciblées sur une image de données forensique « découpée » et les rendre exploitables pour la recherche textuelle génétique des différentes phases de rédaction et des variantes. Cette méthode se présente comme la clef pour restaurer les données de ces systèmes.

4 Conclusion

Dans cet article, nous nous sommes intéressés aux différentes méthodes d'analyse forensiques numériques relatives à la genèse textuelle des disques durs de Jacques Derrida et de d'autres supports de données présentant les mêmes spécificités du point de vue de la matérialité historique numérique. L'analyse *Proof-of-Concept* a montré que l'analyse textuelle génétique des supports de données contenus dans la section dédiée aux archives de Derrida à l'IMEC était possible et offrait de nombreuses de perspectives philologiques. Le débat sur les méthodes et sur les résultats a également montré à quel point les résultats des analyses étaient dépendants de la matérialité numérique historique et avaient rendu nécessaire le développement de nouvelles méthodes forensiques numériques encore inédites.

21 *Parrot Linux*, URL : https://www.parrotsec.org/, (consulté le 02/02/21).