

Introduction

Faire avec le numérique ?

- 1 Manipuler, ouvrir et lire un livre sont des actes qui engendrent des sensations et des expériences multiples. L'une d'entre elles, probablement la plus inconsciente, concerne la matérialité du livre, qu'il soit imprimé ou numérique : le codex ou l'ebook renferme l'aboutissement de plusieurs dizaines de siècles de recherche et de développement. Ouvrir un livre est donc l'occasion de prendre la mesure de cette prouesse technologique que ce sont quelques cahiers reliés ou quelques fichiers compressés. Le livre est un moyen d'interroger le rapport que nous entretenons avec la technique, la façon de concevoir et de fabriquer une publication est révélatrice de notre positionnement avec les machines et les programmes. L'informatisation des métiers du livre et l'arrivée du livre numérique et des nouveaux modes de diffusion de l'écrit modifient la manière d'élaborer des livres ou plus globalement des publications. Conception, fabrication, production, diffusion : l'influence du *numérique* sur le livre nous amène à devoir nous questionner sur la chaîne de publication et sur ce qui la compose. D'un modèle linéaire nous pouvons envisager un système modu-

»m-

ARTICLE, DÉBAT, MÉTHODE

INTEROPÉRABLE, MODULAIRE ET MULTIFORME : L'ÂGE D'OR DE L'ÉDITION NUMÉRIQUE ? DÉMONSTRATION PAR UNE DÉMARCHE PERFORMATIVE

27/03/2019 ANTOINE FAUCHIÉ LAISSER UN COMMENTAIRE

Antoine Fauchié est enseignant au département Information-communication de l'IUT2 de Grenoble, intervenant dans différentes formations universitaires et spécialiste indépendant du livre numérique. Ses publications récentes sont : « Markdown Comme Condition d'une Norme de l'écriture Numérique » dans le sixième numéro de la revue Réel – Virtuel, et « Repenser les chaînes de publication par l'intégration des pratiques du développement logiciel » dans le huitième numéro de la revue Sciences du Design co-écrit avec Thomas Parisot. Une liste plus complète de ses publications et interventions est disponible sur son site : <https://www.quaternum.net/publications/>.

TÉLÉCHARGER LE PDF

Le mémoire d'[Antoine Fauchié](#), *Vers un système de publication modulaire : éditer avec le numérique*, est consultable en ligne : <https://memoire.quaternum.net>. Les sources du mémoire sont également disponibles en ligne : <https://gitlab.com/antoinentl/systeme-modulaire-de-publication/>

LIRE LE MÉMOIRE EN LIGNE

TÉLÉCHARGER LE MÉMOIRE AU FORMAT PDF

Vers un système de publication modulaire : éditer avec le numérique est un travail de recherche réalisé dans le cadre d'un Master. Outre la spécificité du parcours universitaire^[1] dans lequel il s'inscrit, l'originalité de ce mémoire est d'appliquer les principes qui y sont exposés : interopérabilité, modularité et multiformité.

Plutôt que de résumer ici ce texte, nous nous proposons d'évoquer un article co-écrit avec [Thomas Parisot](#) qui est un condensé d'une partie du mémoire [Fauchié et Parisot, 2018]. Ainsi nous notons que les univers du livre et le domaine du développement logiciel ont des points de rencontre, aussi surprenant que cela puisse paraître.

Plusieurs méthodes et outils issus du développement web sont à l'œuvre dans divers projets de livres imprimés ou numériques – plus longuement analysés dans le mémoire –, proposant de nouvelles approches et de nouvelles pratiques d'édition et de design.

Le numérique est utilisé comme mode de conception et de production, et non plus uniquement comme un outil monolithique dont le fonctionnement est caché ou dont les façons de faire sont imposées. Les étapes du processus d'édition doivent être analysées à nouveau, c'est ce que nous nous proposons de réaliser avec Thomas Parisot :

« Quelles sont les influences des méthodes et des outils du développement web sur les chaînes de publication des livres ? »

Cet article fait partie du huitième numéro de la revue [Sciences du design](#) dont la thématique est « Éditions numériques » : notre panorama, certes bref, est particulièrement utile pour comprendre comment l'édition est bouleversée dans ses modes de fonctionnement, et à quel point l'influence du développement web peut être une formidable opportunité.

Au-delà de la réponse théorique formulée avec Thomas Parisot, voici une réponse pratique. Comment écrire un mémoire universitaire sur le sujet des chaînes de publication numérique en bouleversant les méthodes classiques et les outils habituellement utilisés ? Comment appliquer justement ces nouveaux principes pour écrire et publier dans le cadre d'un travail universitaire ?

Il s'agit ici d'une tentative de mise en abyme qui, nous l'espérons, vous donnera de nouvelles clés pour appréhender d'autres pratiques d'écriture, d'édition et de diffusion du savoir académique. Mais avant d'exposer les différentes formes produites en fonction des usages et des finalités de lecture, et questionner les dispositifs d'interaction mis en place, analysons d'abord quelles sont les phases d'écriture et les différentes pratiques possibles en la matière.

1. Écrire

Le point de départ semble isolé au premier abord, il s'agit d'une recherche de confort et d'efficacité : comment créer un espace d'écriture efficace et confortable ? Habitué depuis plusieurs années aux langages de balisage léger [Fauchié, 2018], le recours à un traitement de texte pour l'écriture – y compris les phases préparatoires – n'était tout simplement pas concevable.

Un éditeur de texte est un logiciel simple qui réalise ce pour quoi il a été conçu : inscrire du texte. En ayant recours à un langage de balisage léger comme Markdown pour structurer les contenus – niveaux de titre, citations, passages en emphase – nous obtenons un très bon outil d'écriture.

Ce choix quelque peu radical pour l'écriture est également une question d'anticipation, car plusieurs contraintes devaient être prises en compte pour les phases suivantes : la lecture et l'interaction. La plupart de ces contraintes sont inhérentes à la problématique du mémoire : l'utilisation de logiciels ou programmes qui soient libres ou open source, et la reproductibilité du système d'écriture.

Hors de question d'utiliser Microsoft Word ou Adobe InDesign – pour citer un traitement de texte et un logiciel de publication assistée par ordinateur bien connus. Le caractère ouvert des programmes qui constituent le système de publication permet à quiconque de pouvoir se réapproprier cette proposition pratique – pour la dupliquer et la modifier. Ensuite les formats produits sont envisagés avant le début de l'écriture : ils doivent pouvoir être aussi divers que possible, mais la priorité est mise sur une version web et une version PDF imprimable, une version livre numérique peut être également envisagée.

Enfin la question du mode d'échange avec les deux directeurs de mémoire^[2] exige la mise en place de protocoles de commentaires et de validation spécifiques évoquée en dernière partie du mémoire.

Après le balisage des contraintes, la liste des étapes d'écriture peut être explicitée. **L'inscription et la structuration des contenus constituent la première phase.** Il s'agit d'« écrire », donc de consigner des idées via un dispositif informatique, mais également de « structurer », c'est-à-dire qualifier les informations. Niveaux de titres, passages en exergues, éléments en emphase, listes, citations, autant de moyens de donner une sémantique au texte, et non uniquement du texte au kilomètre.

La structuration est une phase essentielle de l'écriture, qui ne doit pas se limiter au rendu graphique, mais aussi au sens que nous donnons aux idées. Pour cela, comme nous l'avons mentionné, nous pouvons utiliser un langage de balisage léger, qui permet de dissocier le contenu et sa mise en forme. Les traitements de texte ont trop souvent tendance à confondre ces deux dimensions, les interfaces ne favorisant pas une distinction claire entre le texte et sa valeur sémantique. Markdown est constitué de quelques balises, allant du # pour les niveaux de titres, jusqu'aux ** ou _ pour les passages mis en évidence – communément appelés gras ou italique^[3]. Par la suite un processeur permet de transformer ces balisages en un format diffusable comme le HTML, puis comme le PDF.

L'enregistrement et la validation constituent la seconde étape. Les environnements informatiques classiques proposent un outil comme le traitement de texte qui agit de façon monolithique – nous entendons par là une multitude de fonctions au sein d'un même dispositif sans vision visible de son fonctionnement.

Nous nous proposons de différencier l'écriture de l'enregistrement puis de la validation. En nous inspirant des pratiques du développement web, nous nous tournons vers un système de gestion de versions qui apportera de nombreuses fonctionnalités utiles dans des pratiques d'écriture ou d'édition. **Git [Chacon & Straub, 2018] a été pensé pour faciliter la visibilité des modifications successives sur un ou plusieurs fichiers, d'abord pour le code, mais son application peut être beaucoup plus large.**

Chaque session de modifications donne lieu à un enregistrement, et il est possible de revenir à n'importe lequel. Git est une façon puissante de **tracer la vie d'un projet**, que ce soit seul ou à plusieurs. C'est sur ce dernier point que nous souhaitons insister : **Git crée un environnement propice à la suggestion et à la validation par le biais de mécanismes qui reprennent des phases d'édition classiques – relectures, propositions de corrections, validation d'un texte.**

Des plateformes en ligne simplifient quelque peu l'usage de Git qui peut s'avérer très technique sur certaines manipulations, ouvrant des perspectives d'interaction plus importantes dans le cas de validations par exemple. Pour approfondir l'usage de ce système de gestion de versions dans une activité d'édition nous vous renvoyons à la future publication des actes du [colloque ÉCRIDIL 2018](#) [Fauchié, 2018].

La publication et la diffusion peuvent être considérées comme la troisième étape. Publication au sens de « rendre public », c'est-à-dire de produire une version du texte que nous pouvons mettre à disposition. Dans notre cas la publication consiste en la génération de formes lisibles par des humains – et non plus par des programmes comme dans le cas de Markdown.

Le format web est celui qui nous intéresse en premier lieu – nous revenons sur ce point précis dans la partie suivante –, mais nous devons aussi prévoir des versions imprimables ou lisibles sur des dispositifs de lecture numérique. Pour transformer le format Markdown en HTML, une pléthore de solutions existe, l'une d'entre elles a retenu notre attention et est utilisée par exemple par Getty Publications [Gardner, 2017] : un générateur de site statique.

Un générateur de site statique crée un ensemble de pages web à partir de fichiers Markdown et d'autres fichiers de configuration. Ainsi nous pouvons inscrire et structurer un texte via un langage de balisage léger, puis générer un ensemble de page HTML qui donneront lieu à un site web ou à d'autres versions numériques. Une diffusion est alors possible, en numérique mais aussi en imprimé puisque nous pouvons aussi transformer des pages HTML en un fichier PDF.

Nous distinguons ici trois phases principales dans une activité de publication ou d'édition, et pour chacune d'entre elles nous sollicitons des outils différents qui peuvent s'interconnecter :

- un langage de balisage léger pour inscrire et structurer
- un système de gestion de versions pour enregistrer et valider
- un générateur de site statique pour publier puis diffuser

La question de la diffusion est centrale : quelles formes allons-nous rendre disponibles et dans quels buts ?

2. Lire

La lecture doit être prise en compte au moment de l'écriture : nous devons envisager comment notre texte sera reçu et perçu. C'est l'objectif intrinsèque de la structuration – ou de la sémantique – et de la mise en forme. L'écriture du mémoire s'est donc engagée dans un processus conjoint de production de plusieurs formes. Tout d'abord pour les directeurs qui pouvaient valider certains fragments, puis pour tout lecteur souhaitant suivre la progression, et enfin pour donner à voir une recherche en cours.

Le premier chantier a été de proposer un choix de disposition de lecture numérique aux directeurs – c'est-à-dire créer une forme de confort et non uniquement un contenu lisible –, accompagnée d'une plateforme incluant des systèmes de commentaire et de validation. La première forme lisible – en dehors de la source du texte au format Markdown – a été permise par la plateforme GitLab, proposant une

interface minimaliste de lecture, mais comportant nombre de fonctionnalités.

Dès les premiers travaux préparatoires – commentaires de texte et analyses de cas – les directeurs ont pu commenter voir valider les textes via cette plateforme. Nous n’abordons pas en détail le fonctionnement, mais la figure 1 illustre ce premier accès.



Figure 1 : capture d’écran de la plateforme GitLab

Dans un deuxième temps un site web a été produit à partir des fichiers Markdown et depuis le dépôt^[4], s’inspirant fortement de la chaîne de publication mise en place par Getty Publications. Chaque intervention sur les fichiers – et versionnés sur le dépôt GitLab – engendrait la génération des fichiers constituant le site web.

Plus lisible et accessible que la plateforme GitLab, ce site web est composé de plusieurs pages web – une page par sous-partie, les sous-parties formant une partie. Un menu permet une vue d’ensemble et une navigation dans le mémoire. Chaque paragraphe est numéroté pour pouvoir facilement préciser sur quelle partie précise porte une remarque. **La question de la citabilité d’un fragment numérique – qui par définition peut évoluer – est délicate**, d’autant plus lorsque l’objectif est d’ajouter la dimension de versionnement^[5].



Figure 2 : capture d’écran du site web <https://memoire.quaternum.net>

Cette version a été mise en place pour faciliter la lecture du mémoire, [chaque page comporte un lien vers sa source sur le dépôt](#), les lecteurs ayant ainsi la possibilité d'intervenir sur les fichiers. Il ne s'agit donc pas que d'une version parmi d'autres, mais bien plutôt du [point d'entrée principal vers d'autres versions ou vers des collaborations](#).

Enfin une version imprimable était nécessaire pour ne pas dépendre que du numérique, et parce que certains lecteurs ne souhaitent pas effectuer une lecture longue sur écran. [Pour produire un fichier PDF deux composants ont été requis : le générateur de site statique](#), pour générer une seule page web comportant tous les contenus associés à une feuille de style spécifique ; [et paged.js \[Taquet, 2018\]](#), un script qui facilite la conversion d'une page HTML en fichier PDF via la fonction impression d'un navigateur web.

Pour fabriquer cette version, la source des textes – rappelons-le, au format Markdown – reste donc la même, ce qui est un avantage conséquent : [il n'est pas nécessaire de modifier la source en fonction des formes produites](#). Concernant la version imprimée nous devons toutefois retenir un point particulier : que se passe-t-il si la version web connaît des évolutions successives ? [Comment suivre le fil de modifications numériques postérieures à une version imprimée](#) – physique ? Autant de questions qu'il faudrait prolonger ailleurs.

[En terme de processus de publication, chaque élément étant modulaire, aucune possibilité n'est écartée, comme par exemple un troisième format comme l'EPUB – le format du livre numérique –, ou une version XML](#). Enfin, nous aurions pu mettre en ligne les textes de façon plus progressive avec un système d'abonnement et la génération automatique d'une lettre d'information (ou de notifications sur des réseaux sociaux) pour informer des évolutions.

La question de la publication d'un texte en brouillon reste entière, d'autant plus dans le cas d'un travail universitaire de deuxième cycle – rappelons-le, il s'agit d'un mémoire de Master 2. Une première restriction a été mise en place : une absence de communication sur le dépôt et le site web avant les premiers retours de mes directeurs. La réalisation de ce mémoire appliquant les principes qui y sont exposés s'imposait, en partant sur l'adoption d'une démarche performative^[6].

Il y a une volonté, parmi la communauté scientifique, de publier tôt, notamment par le biais des *preprints* – ces textes bruts mis à disposition avant une publication dans une revue ou un ouvrage. Rendre disponible un brouillon est encore un autre cas, et peut représenter un intérêt dans des situations bien spécifiques – et en dehors d'une démarche performative comme ici.

Au-delà de ces formes diverses comme une matérialisation des différents objectifs de ce mémoire, qu'en est-il de la question de

l'interaction avec le texte ? Si la lecture est possible, qu'en est-il des commentaires, validations ou propositions de modifications ?

3. Interagir

Le mémoire *Vers un système modulaire de publication : éditer avec le numérique* a une **forte dimension expérimentale**, autant pour les dispositions de lecture proposées que pour les tentatives d'interactions entre le texte et plusieurs acteurs de ce travail universitaire. « Tentatives », car nous devons révéler les intentions et les efforts tout en soulignant les éléments qui n'ont pas abouti. Nous gardons un regard critique sur ce qui constitue une expérimentation, même si elle peut avoir un caractère reproductible.

Deux fonctionnalités étaient nécessaires pendant l'écriture du mémoire et dans le cadre des échanges avec les directeurs : commenter le texte et valider les versions. **Commenter le texte peut être divisé en plusieurs actions : ajouter un commentaire à un texte dans son ensemble, ou commenter un passage précis.** L'usage de Git permet ces deux actions, notamment via l'utilisation des *merge requests*; nous devons nous attarder un moment sur cette spécificité.

« Une demande de fusion », ainsi qu'on peut le traduire, correspond à un moment précis dans les échanges autour d'un projet versionné avec Git : **après que plusieurs intervenants aient modifié des parties du projet, ils proposent de les fusionner avec la version de départ – celle d'avant les modifications.**

Au moment d'une *merge request* les autres acteurs du projet peuvent ajouter des commentaires généraux ou modifier directement des fichiers. À ce niveau il est également possible de commenter le texte ligne par ligne, ce qui est plus précis. Par convention lorsque nous écrivons avec un langage de balisage léger, nous sautons une ligne à chaque phrase : cela permet justement de rendre plus atomique le contenu sans impacter le rendu final qui ne fera pas apparaître ces sauts de ligne.

À chaque nouvelle étape du mémoire, que ce soit la fin de la rédaction d'un commentaire de texte ou celle d'une partie, une *merge request* est créée et les directeurs sont parfois sollicités pour commenter ou valider. Car c'est l'autre fonction de la demande de fusion : pouvoir être validée, et ce assez facilement par le biais de plateforme comme GitLab et des outils mis à disposition.

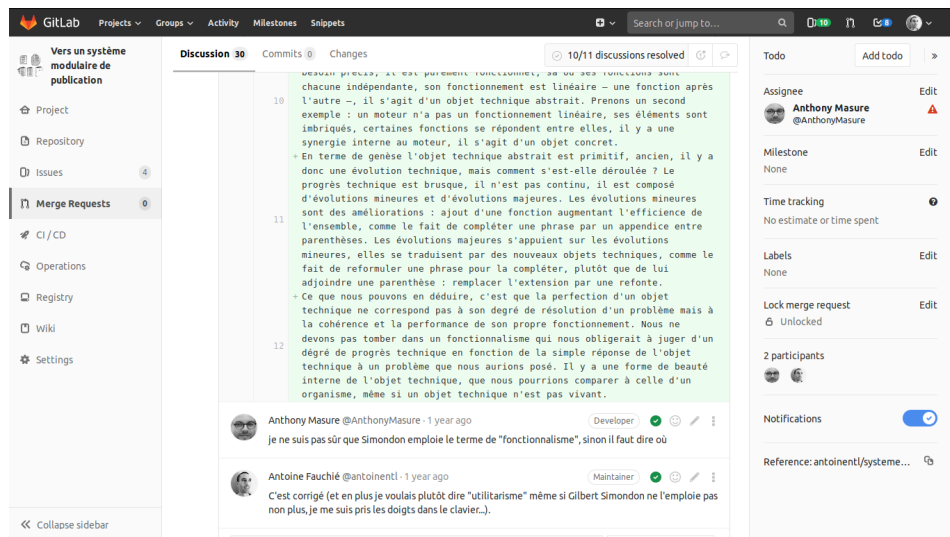


Figure 3 : capture d'écran d'un commentaire sur la plateforme GitLab

Deux nuances doivent être formulées : nous avons envisagé l'usage de l'outil d'annotation Hypothes.is^[Z] qui a été assez vite écarté, peut-être à tort ; interagir via Git ou la plateforme GitLab ne s'est pas avéré si efficient ou évident, malgré la forte implication des deux directeurs de mémoire.

L'utilisation de Git pour un tel projet doit être questionnée, il ne s'agit pas d'une solution parfaite, et il faut parfois tordre ce logiciel pour arriver à nos fins. Plutôt que d'imaginer un *fork* de Git adapté pour l'édition, il faut plutôt mettre en place des règles ou des conventions, en voici quelques exemples :

- segmenter les textes trop longs en différents fichiers ;
- effectuer un saut de ligne à chaque phrase pour identifier les modifications et faciliter les commentaires ;
- créer des *commits* atomiques : un *commit* est un enregistrement commenté après une session de modifications, un *commit* atomique concerne une série de modifications cohérentes entre elles. Ce point concerne par ailleurs tout projet géré avec Git, que ce soit du code ou du texte ;
- etc.

Git demande des manipulations trop longues pour des actions simples, cela correspond bien à l'environnement de la programmation – surtout lorsque les contributeurs sont nombreux –, moins à une activité de publication ou d'édition qui réunit peu de personnes.

Plus globalement cette entreprise a été singulièrement influencée par un désir de simplicité : ne pas dépendre d'un nombre trop important de développements spécifiques ; ajuster les adaptations du développement web vers la publication numérique pour éviter de fabriquer une chimère logicielle ; montrer qu'il est possible de construire un « système » de publication opérant et opérable.

Ce dépouillement avait un autre objectif plus essentiel, celui d'engendrer un système reproductible, que d'autres pourraient

s'emparer. Et cela a été en partie le cas puisqu'un bibliothécaire (Michael Ravedoni) a créé un modèle à partir des différents outils rassemblés, permettant à quiconque de se lancer facilement dans une aventure de publication numérique^[8]. C'est en quelque sorte un aboutissement imprévu et bienvenu.

Conclusion

L'une des critiques que nous pouvons formuler à l'égard des points abordés dans les trois parties de ce texte, est celle des possibilités réelles d'application et des contraintes techniques. La compréhension et la manipulation des technologies invoquées nécessitent une culture numérique, voir des connaissances en programmation.

Si l'auteur du mémoire n'a pas un profil de développeur, il faut toutefois noter la difficulté que peuvent rencontrer des personnes qui n'aurait pas un bagage minimum dans la maîtrise de certaines briques techniques – que ce soit Markdown, Git ou l'usage de certains programmes. Il faut considérer cette contrainte, ce que nous avons appelé « de nouvelles dépendances » dans la conclusion du mémoire, comme une opportunité de mieux comprendre les machines et les programmes, de mieux nous situer par rapport à eux [Simondon, 2012], et de nous réapproprier les outils d'écriture – ici dans le champ scientifique [Vitali-Rosati, 2018].

Si cela nécessite un effort certain, il peut prendre plusieurs formes : la formation des individus souhaitant acquérir ces connaissances, ou l'ajout d'un profil technique au sein d'une structure de publication.

Ce mémoire est toujours en cours de développement, à la façon d'un programme il reçoit des ajouts, des remarques, et une quatrième partie est en préparation. Nous avons abordé des questions techniques d'édition et de publication, mais il ne faut pas passer à côté de ce qui a été mis en place pour l'écriture.

La modularité du mémoire ne se limite pas à ses composants en terme de production de formes, elle concerne aussi l'organisation des textes. Nous nous sommes inspirés de pratiques relativement courantes, qui consistent à intégrer tous les matériaux d'un travail de recherche dans le texte, plutôt que de les réunir en annexes. Ce n'est pas là un détail, car ainsi le mémoire est constitué de fragments, potentiellement indépendants, qu'il faut ensuite agencer pour créer une cohérence.

Un commentaire de texte, une analyse de cas, un entretien, autant de travaux parfois jugés préparatoires que nous avons souhaité intégrer « dans » le mémoire, à la fois pour enrichir la réflexion, pour les mettre avant, mais aussi pour proposer une lecture non linéaire.

Plutôt que de proposer une nouvelle fois le schéma classique d'un texte central suivi d'annexes, les commentaires de texte et les

analyses de cas sont intégrées au début, et les entretiens seront ajoutés à la fin. Il est possible de lire le mémoire « dans le désordre », en commençant par la troisième partie puis en naviguant jusqu'aux deux premières.

Par ailleurs certains des contenus du mémoire ont pu faire l'objet de publications, nous assistons alors non plus à une convergence, mais à une ouverture vers d'autres espaces de diffusion, en résonance avec d'autres travaux, domaines et chercheurs^[2].

Une analyse de cas – finalement non retenue pour le mémoire – a été publiée dans une revue en ligne, une partie du mémoire a été développée pour une communication, et cet article lui-même est un texte satellite constituant un élément clef de la démarche globale. Au-delà de la dimension profondément numérique de cette publication, il faut considérer ce travail comme un déploiement.

Bibliographie

Chacon, S., & Straub, B. (2018). *Pro Git Book*. Consulté à l'adresse <https://git-scm.com/book/fr/v2/>

Crawford, M. B. (2016). *Éloge du carburateur: essai sur le sens et la valeur du travail*. (M. Saint-Upéry, Trad.). Paris, France: La Découverte.

Epron, B., & Vitali-Rosati, M. (2018). *L'édition à l'ère numérique*. Paris: La Découverte. Consulté à l'adresse <https://www.cairn.info/l-edition-a-l-ere-numerique-9782707199355.htm>

Fauchié, A. (2017, janvier 23). Publier des livres avec un générateur de site statique. Consulté 30 avril 2018, à l'adresse <https://jamstatic.fr/2017/01/23/produire-des-livres-avec-le-statique/>

Fauchié, A. (2018). Markdown comme condition d'une norme de l'écriture numérique. *Réel – Virtuel*, (6). Consulté à l'adresse <http://www.reel-virtuel.com/numeros/numero6/sentinelles/markdown-condition-ecriture-numerique>

Fauchié, A., & Parisot, T. (2018). Repenser les chaînes de publication par l'intégration des pratiques du développement logiciel. *Sciences du Design*, n° 8(2), 45-56.

Gardner, E. (2017, janvier 23). Publier des livres avec un générateur de site statique. Consulté à l'adresse <https://jamstatic.fr/2017/01/23/produire-des-livres-avec-le-statique/>

Simondon, G. (2012). *Du mode d'existence des objets techniques*. Paris, France: Aubier, impr. 2012.

Taquet, J. (2018, juillet 22). Paged.js – sneak peeks. Consulté 8 mars 2019, à l'adresse <https://www.pagedmedia.org/pagedjs-sneak-peeks/>

Vitali-Rosati, M. (2018, mars 11). Les chercheurs en SHS savent-ils écrire? Consulté 20 mars 2018, à l'adresse <http://theconversation.com/les-chercheurs-en-shs-savent-ils-ecrire-93024>

Web en Vert. (2016). *Documentation as code (expliqué à mon père)* – Hubert Sablonnière. Consulté à l'adresse <https://www.youtube.com/watch?v=1rKgVF5CEEY>

[1] Ce mémoire a été réalisé dans le cadre d'une Validation des acquis de l'expérience pour le Master Sciences de l'information et des bibliothèques, spécialité Publication numérique.

[2] Anthony Masure, maître de conférences en design à l'université Toulouse – Jean Jaurès et de Marcello Vitali-Rosati, professeur au département des littératures de langue française de l'Université de Montréal et titulaire de la Chaire de recherche du Canada sur les écritures numériques.

[3] De nombreuses ressources existent pour comprendre et apprendre à utiliser Markdown, par exemple : <https://commonmark.org/>

[4] Un dépôt est un espace où l'on peut retrouver tous les fichiers d'un projet versionné avec Git.

[5] Voir à ce sujet les échanges avec Louis-Olivier Brassard : <https://gitlab.com/antoinentl/systeme-modulaire-de-publication/issues/26>

[6]. Pour reprendre une expression empruntée à Benoît Epron.

[7] Hypothes.is ou Hypothesis est un logiciel et un service en ligne d'annotation, principalement conçu pour un usage dans le domaine académique (<https://web.hypothes.is/>).

[8] Le dépôt de ce modèle est disponible en ligne : <https://github.com/michaelravedoni/jekyll-book-framework>

[9]. Une bibliographie intitulée « Autour du mémoire » correspond à ces différentes publications : <https://memoire.quaternum.net/6-annexes/autour-du-memoire/>

INTEROPÉRABLE, MODULAIRE ET MULTIFORME : L'ÂGE D'OR DE L'ÉDITION NUMÉRIQUE ? DÉMONSTRATION PAR UNE DÉMARCHE PERFORMATIVE

dlis.hypotheses.org/4381

[Antoine Fauché](#)

ÉCOSYSTÈME DE PUBLICATION NUMÉRIQUE

ÉCRITURES NUMÉRIQUES GIT LANGAGE DE BALISAGE

LOGICIEL DE GESTION DE VERSIONS MARKDOWN

PUBLICATION NUMÉRIQUE
