

ÉCOLE NATIONALE DES CHARTES

Gilmar Ballestrim

Licencié en histoire à l'Universidade de São Paulo

Vers une persistance numérique des collections :

Développement d'outils IIIF au sein du DaSCH

Mémoire pour le diplôme de master

« Technologies numériques appliquées à l'histoire »

Septembre 2024

Résumé

Dans le cadre de mon stage au sein du DaSCH, j'ai développé trois applications visant à permettre à des versions allégées d'images 2D et objets 3D d'intégrer leurs métadonnées structurelles et descriptives au cadre international d'interopérabilité des images (*International Image Interoperability Framework - IIIF*). La question de recherche est la suivante : est-il réellement possible de garantir la persistance de données archivistiques en utilisant IIIF ?

Le principal défi aujourd’hui réside dans la gestion de données hétérogènes provenant d’objets intrinsèquement variés, ainsi que de différentes organisations ayant des objectifs distincts. L’enjeu est d’éviter l’effet « *setting and forgetting* » en cherchant à rendre la description et l’exploitation des données pérennes entre diverses organisations. De plus, l’utilisation de ces données soulève des problèmes et des impacts environnementaux. Par conséquent, l’optimisation de la préservation numérique et de l’interopérabilité apparaît comme une priorité à développer, car elle permettrait une utilisation efficace des données dans un contexte patrimonial.

Mots-clés : IIIF ; persistance de données ; *green coding* ; gestion de données ; PYTHON ; HTML ; CSS ; JAVASCRIPT ; *front end* .

Informations bibliographiques : Gilmar Ballestrim, *Vers une persistance numérique des collections*, mémoire de Master « Technologies numériques appliquées à l’histoire », dir. Edward Gray, École nationale des chartes, septembre 2024.

Abstract

During my internship at the DaSCH, I developed three applications aimed at enabling lighter versions of 2D images and 3D objects to integrate their structural and descriptive metadata into the International Image Interoperability Framework (*International Image Interoperability Framework - IIIF*). The research question is as follows : Is it truly possible to ensure archival data persistence using IIIF ?

The main challenge today lies in managing heterogeneous data from intrinsically varied objects and different organizations with distinct objectives. The goal is to avoid the « *setting and forgetting* » effect by striving to make data description and utilization sustainable across various organizations. Additionally, the use of this data raises environmental issues and impacts. Therefore, optimizing digital preservation and interoperability emerges as a priority to be developed, as it would allow for the efficient use of data in a heritage context.

Keywords : IIIF ; data persistence ; *green coding* ; data management ; PYTHON ; HTML ; CSS ; JAVASCRIPT ; *front end* .

Bibliographic Information : Gilmar Ballestrim, *Towards Digital Persistence of Collections*, Master's thesis in « Digital Technologies Applied to History », dir. Edward Gray, École nationale des chartes, September 2024.

Dédicace

Tout comme le titre du livre de l'écrivain portugais Valter Hugo Mæe, je me sens aussi un fils de mille hommes, mais j'ajoute : aussi de mille femmes.

Je souhaite dédier ces premiers lignes, à ceux et celles qui m'ont permis d'écrire toutes les lignes restantes de ce travail. À ceux et celles qui, par leur travail physique, dès l'enfance, m'ont permis d'avoir un travail intellectuel à un moment de ma vie.

Ces lignes symbolisent non seulement la liberté, mais aussi de nouvelles pages dans ma vie.

À mes deux grands-pères disparus, Chico et Jobel, qui m'ont donné des trésors invisibles qui demeurent au cœur de ma vie. Un jour, j'ai promis à l'un d'eux que je serai le premier de la famille à entrer à l'université et, aujourd'hui, je peux dire que j'ai osé aller plus loin. Merci.

Avec amour,
Gil.

Remerciements

C'est avec une profonde gratitude que je tiens à remercier toutes les personnes qui ont contribué à la réalisation de ce mémoire de master.

Je tiens tout d'abord à exprimer ma gratitude au corps professoral de l'École nationale des chartes, en particulier à Emmanuelle Bermès, pour leur soutien académique et leur bienveillance.

Je souhaite exprimer ma sincère reconnaissance à mon directeur de mémoire, le Professeur Dr. Edward Gray, pour son encadrement éclairé, ses conseils précieux et ses corrections tout au long de ce parcours. Son expertise et son expérience ont été déterminantes pour mener à bien ce travail.

Je remercie chaleureusement Rita Gautschy, ma cheffe de stage, pour son aide inestimable, sa confiance et les échanges enrichissants que nous avons eus. Je suis également très reconnaissant envers Daniela Meier et Gaby de l'IAESTE pour leur assistance qui a rendu mon séjour à Bâle des plus agréables. Enfin, je tiens à remercier toute l'équipe du DaSCH pour les bons moments passés pendant mon stage, et en particulier Julien Raemy pour ses conseils avisés et nos échanges fructueux.

Mes remerciements vont également à mes amis et collègues de l'École nationale des chartes, en particulier à Anna, Marina et Sarah, pour leur soutien, les discussions et les échanges qui ont été très importants. Je remercie également Amos et Karim pour les discussions très agréables que nous avons eues tout au long de cette année.

Je voudrais exprimer ma gratitude à l'équipe de la Communication de l'École nationale des chartes, notamment à David, pour leur soutien dans le cadre de mon premier emploi en tant que développeur.

Je remercie mes amis et collègues de l'Université de São Paulo, Denil-

ton Donizetti, Gustavo Vicente, Jeferson Benevides, Johnny Esteves, Lucas Soares, Lucas Gandolfi, Thiago Marinho Del Corso, Thiago Kenji, Tomas Mistrorigo, Tomas Pinho et William Valério, pour les moments de partage et les discussions enrichissantes qui ont marqué cette période d'études.

Je souhaite remercier du fond du cœur Addison, Alice, Antônio, Carla, Caroline, Gabriel Bassan, Gabriel Perli, Isaure, João, José Victor, Miguel, Nordine, Pietro Piccioni, Plínio, Rafael et Vitor Sanches pour leur amitié et leur soutien inconditionnel dans les différents moments de ma vie en France.

Un grande merci à mes amis et amies, Camilo, Gabriel Cruz, Gabriel Perli, Greicy Lins, João Pedro et Renata Cavazzana. Votre présence à mes côtés, votre écoute attentive et vos encouragements sans faille ont été les piliers de cette aventure.

Je tiens à exprimer ma profonde gratitude à mes parents, Inês et Osmar, pour leur amour, leur confiance et leur soutien inconditionnel dans tous mes choix.

Un immense merci à Laurène pour ses encouragements, son aide précieuse et sa patience. Sa présence à mes côtés a été inestimable.

Je remercie toutes les institutions qui ont contribué à ce projet, notamment la Société d'Entraide de l'École des chartes, en particulier Isabelle Le Masne de Chermont, Camille Dégez-Selves et Solange Bidou. Je remercie également Florian Bourguignon et l'agence Erasmus. Leur soutien moral et financier a été inestimable et précieux pour moi.

Finalement, je remercie Edward Gray, Rita Gautschy, Gabriel Perli et Laurène pour leur relecture.

Introduction

Programmer, dans notre société contemporaine, ne se résume plus à écrire des lignes de code : c'est également une activité qui s'inscrit dans un contexte social, politique et environnemental complexe. L'essor du concept de *green coding* met en lumière la nécessité de prendre en compte les impacts environnementaux et sociaux de la production de logiciels. C'est pour cette raison que dans le cadre de mon stage au sein du *Swiss National Data and Service Center for the Humanities*, j'ai été amené à développer deux applications éco-responsables visant à intégrer des versions allégées d'images 2D et modèles 3D, ainsi que leurs métadonnées structurelles et descriptives, au sein du cadre international d'interopérabilité des images, connu sous le nom d'*International Image Interoperability Framework* (IIIF). Ce travail s'inscrit dans une problématique plus large, soulevée par la question de recherche suivante : est-il réellement possible de garantir la persistance des données archivistiques en utilisant IIIF ?

Cette interrogation prend toute son importance dans le contexte actuel, où la gestion de données hétérogènes constitue un défi majeur. Ces données, issues d'objets intrinsèquement variés et de différentes organisations, posent des problèmes d'harmonisation et de pérennisation. Le défi réside notamment dans la nécessité d'éviter l'effet « *setting and forgetting* », c'est-à-dire l'abandon de données après leur archivage initial, en développant des solutions qui assurent une description et une exploitation durable des informations à travers diverses organisations. De plus, l'usage de ces données numériques soulève des questions environnementales, rendant l'optimisation de leur préservation et interopérabilité une priorité.

En outre, programmer est aussi une activité qui s'inscrit dans un moment historique précis. Les codes conçus ont pour but la préservation patrimoniale : bien qu'ils aient une finalité spécifique, ils sont souvent créés dans les contextes collectifs et d'*open source* et sont susceptibles de devenir une marchandise. Cette discussion sera approfondie, mais on peut dès à présent souligner que rares sont les questionnements concernant les effets de la codification et la manière dont elle masque les relations sociales qui impactent, directement ou indirectement, non seulement l'environnement, mais aussi la vie des individus. Autrement dit, la réflexion de cette étude n'est pas d'épuiser le sujet de la relation entre la programmation et l'environnement, puisque cela est un débat déjà connu, notre idée est plutôt de mettre en exergue le fait que ces deux dimensions sont forcément liées à un mode de reproduction historique de la vie. En effet, comme Karl Marx l'avait souligné, les idées ne surgissent pas dans le vide, mais sont enracinées dans la réalité matérielle¹. Dans notre société, les moyens de production sont principalement capitalistes. Si le capitalisme n'est pas un système transhistorique, comme l'a également souligné Moishe Postone², il imprègne une grande partie des relations sociales, dont la programmation est l'une des manifestations.

Toutefois, ce travail tente de penser des possibilités pour restituer un autre mode de programmation dans cette configuration sociale. Or, l'histoire

1. Pour Marx, les Jeunes-Hégéliens restaient enfermés dans une pure spéulation philosophique, sans chercher à relier leurs idées à la société concrète dans laquelle ils vivaient. C'est de là que vient la célèbre phrase de Marx « Ce que sont les individus dépend donc des conditions matérielles de leur production » Karl Marx, *L'idéologie allemande : Critique de la philosophie allemande la plus récente dans la personne de ses représentants Feuerbach, B. Bauer et Stirner, et du socialisme allemand dans celle de ses différents prophètes*, Éditions Sociales, traduction par Henri Auger, Gilbert Badia, Jean Baudrillard, Renée Cartelle, Paris, 1968, p. 44-45.

2. A l'instar de Postone, la théorie du travail qui intéresse cette étude est celle historiquement spécifique à la société capitaliste, non une sorte de capitalisme qui s'étend à toutes les sociétés de forme transhistorique. C'est justement cette dernière notion qui représente la critique centrale du livre de l'auteur. En outre, cette théorie va de pair avec la notion qui implique que le capitalisme est l'une des caractéristiques de base de la modernité Moishe Postone, *Time, Labor, and Social Domination : A Reinterpretation of Marx's Critical Theory*, Cambridge University Press, Cambridge, 1996, p. 9-21.

de la programmation évidemment n'est pas une simple histoire de la guerre ou du capitalisme. Les discussions sur l'utilisation des logiciels libres et *open source*, à l'instar des *Free Open Source Software* (FOSS), ont une histoire liée à la contre-culture aux États-Unis : ces aspirations suggéraient beaucoup plus l'idée d'une quête pour la liberté des individus que de répondre aux aspirations capitalistes³.

De plus, ce mémoire est également le moment de remettre en question certains concepts tels que le document, l'objet et l'artefact : ces définitions, qui font l'objet de débats terminologiques depuis longtemps, sont de plus en plus placées dans un contexte d'études interdisciplinaires. Un exemple de ce que nous explorerons et défendrons dans ce travail est le champ interdisciplinaire de la culture matérielle, qui nous intéresse de près, car ces définitions englobent des dimensions historiques qui influent sur les manières dont nous conservons les choses avec lesquelles nous interagissons et, par conséquent, sur la façon dont nous transformons ces choses en données ou comment nous concevons les données, puisque les données présentent ces deux aspects non négligeables : celui de choses qui sont devenues numériques ou celui de choses qui naissent numériques.

Ce mémoire est structuré en trois parties principales, chacune visant à explorer en profondeur les différentes dimensions du travail effectué et à situer les développements réalisés dans leur contexte théorique et pratique.

La première partie, intitulée **Production au sein du DaSCH**, se concentre sur la mise en place du cadre institutionnel et technologique dans lequel s'inscrit ce projet. Cette section commence par un examen détaillé du DaSCH, une institution suisse dédiée à la préservation numérique du patrimoine culturel. Le premier chapitre retrace l'histoire et la fondation du DaSCH, en mettant l'accent sur ses liens avec l'initiative de l'*Open Data* en Suisse et sur

3. Voir Tiebing Shi, « Mainstreaming, Counter-Co-Optation, and Depoliticization by a Counterculture », *Journal of Marketing Development and Competitiveness*, 8-3 (2014), p. 109-117.

les modèles de financement qui soutiennent ses activités. Ensuite, le contexte technique du IIIF est introduit. Ce cadre d’interopérabilité, crucial pour la gestion et le partage des images patrimoniales, est présenté à travers ses différentes versions, depuis l’API 3.0 jusqu’aux développements envisagés pour l’API 4.0. Enfin, cette partie aborde les concepts fondamentaux liés à la persistance des données numériques, en clarifiant les distinctions entre documents, objets et données, tout en soulignant les défis posés par la numérisation du patrimoine. Une attention particulière est mise sur les enjeux spécifiques du stage, les objectifs des *pipelines* développés, et les limitations rencontrées dans l’automatisation des processus à travers l’utilisation de technologies comme PYTHON, FLASK, BOOTSTRAP, et DOCKER.

La deuxième partie, dont le titre est **De l'image au standard IIIF : conceptions des *pipelines* de traitement**, se consacre à l’aspect technique du projet, détaillant la conception et la mise en œuvre des *pipelines* de traitement d’images pour leur intégration dans le standard IIIF. Cette section se divise en plusieurs chapitres qui expliquent, étape par étape, le processus d’optimisation des images et des objets numériques pour répondre aux exigences du IIIF. Elle explore notamment le développement des outils DaSCH Converter et DaSCH Mesh, qui ont été conçus pour faciliter cette intégration. Une attention particulière est accordée à la structure de ces outils, à leur architecture logicielle, ainsi qu’à l’expérience utilisateur. Finalement, la discussion porte sur les défis techniques et des solutions apportées pour enrichir les métadonnées et alimenter les manifestes IIIF lors de la conception d’DaSCH IIIF.

La troisième partie, formulée en tant qu’**Évaluation des outils et méthodes : bilan et perspectives**, propose une analyse critique des résultats obtenus au cours du développement des applications et aussi une réflexion à propos des outils de gestion du temps. Elle commence par une évaluation de la performance des outils créés, en examinant leur efficacité, leur robustesse et leur capacité à répondre aux objectifs initiaux du projet. Est discutée ensuite la façon dont les outils de gestion ont été intégrés lors du stage. Cette

partie se poursuit avec une réflexion sur les perspectives d'évolution des outils, en identifiant des fonctionnalités supplémentaires qui pourraient être développées pour améliorer encore davantage l'intégration des images et métadonnées dans le cadre IIIF. Enfin cette section se termine sur des pistes de recherches futures, en considérant l'impact potentiel des technologies explorées sur la préservation du patrimoine culturel numérique.

Ce mémoire vise ainsi à fournir une analyse des enjeux techniques et institutionnels liés à la gestion des données patrimoniales numériques, tout en proposant des solutions éco-responsables pour améliorer leur interopérabilité et leur préservation à travers l'utilisation du standard IIIF, sans oublier la dimension historique sous-jacente à cette reproduction historique de vie.

Première partie

Production au sein du DaSCH

Chapitre 1

Contexte DaSCH

1.1 Qu'est-ce que DaSCH ?

Le *Swiss National Data and Service Center for the Humanities*, connu sous l'acronyme DaSCH, est une infrastructure nationale sans but lucratif pour les données de recherche en sciences humaines, axée sur les données qualitatives en Suisse, c'est-à-dire les données *bitstream* telles que les images, l'audio, la vidéo et texte.

La mission de DaSCH est essentielle au développement d'une infrastructure de recherche numérique fiable et ouverte pour les sciences humaines en Suisse. En tant que dépositaire de données de recherche, DaSCH s'engage à assurer un accès direct et durable aux données tout en permettant leur modification continue.¹ Cette mission est renforcée par l'accent mis sur l'interopérabilité et l'utilisation de normes reconnues, garantissant ainsi une intégration fluide avec les outils existants utilisés par les communautés scientifiques. De plus, l'engagement de DaSCH dans la promotion des données ouvertes et dans la formation à la gestion des données de recherche témoigne de son rôle central dans l'écosystème de la recherche numérique en Suisse. Ce rôle est encore amplifié par la collaboration étroite avec les communautés de recherche, tant au niveau national qu'international, ce qui permet de main-

1. Voir DaSCH, *Mission / DaSCH*, URL : <https://www.dasch.swiss/mission> (visité le 29/08/2024).

tenir une plateforme en constante évolution, adaptée aux besoins changeants des chercheurs.

Les valeurs de DaSCH constituent le fondement de sa mission et guident toutes ses actions. La fiabilité est au cœur de l'engagement de DaSCH envers ses utilisateurs, assurant une communication transparente et le respect des délais. Cette fiabilité est complétée par une flexibilité qui permet à DaSCH de répondre efficacement aux besoins spécifiques des chercheurs, en envisageant des alternatives et en proposant des scénarios adaptés. L'appréciation et la curiosité, quant à elles, favorisent un environnement de travail constructif et innovant, où les besoins des chercheurs sont constamment pris en compte et où l'exploration de nouvelles technologies est encouragée.

Pour comprendre comment la mission et les valeurs de l'association s'alignent avec d'autres aspects, y compris sa fondation, nous allons procéder à une brève analyse de la gouvernance de DaSCH.

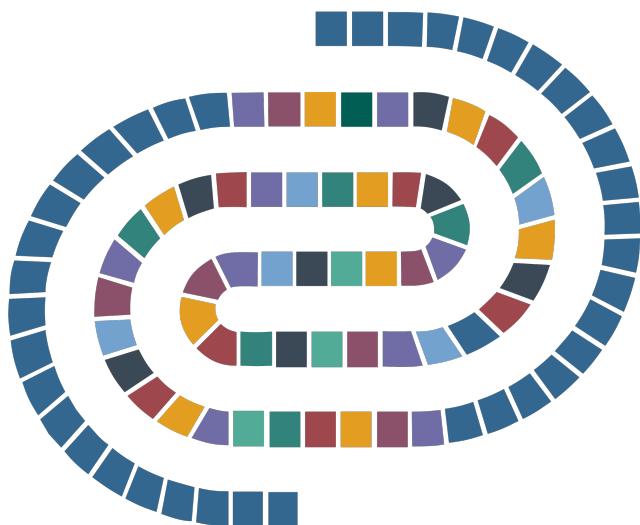


Figure 1.1 – Logo du DaSCH

1.1.1 Gouvernance du DaSCH

Actuellement, l'*Executive Board* du DaSCH est dirigé par la Prof. Dr. Rita Gautschi, qui occupe le poste de *Head* du *Repository Services*. Dr. Ivan Subotic est, quant à lui, le *Deputy Director*. Les *Repository Services*, dont Gautschi est la cheffe, englobent les départements de *Product Management*, *Interoperability and Archiving*. Concernant le *Research Data Unit*, elle est également responsable du *Research Data Management* et du *Scientific Programming*. Ivan, en charge de l'*Engineering*, s'occupe de l'*Infrastructure*, du *Software Development*, de l'*UX* et du *Support*.

Par ailleurs, Rita Gautschi est également l'interlocutrice principale pour d'autres partenariats du DaSCH, tels que DARIAH-CH², et l'*Office Management*, lié à l'Universität Basel.

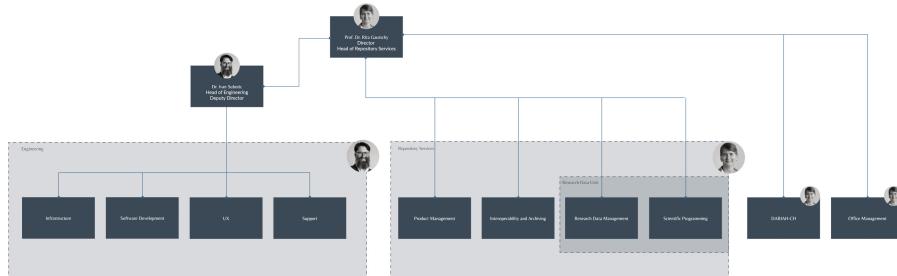


Figure 1.2 – Organigramme du DaSCH

Le DaSCH adopte un modèle de gouvernance participative qui vise à englober l'ensemble des acteurs concernés par la recherche en sciences humaines en Suisse. Cette approche inclusive se traduit par une adhésion ouverte à un large éventail d'institutions, allant des universités aux académies et aux infrastructures de recherche spécialisées. Cette diversité de membres garantit une représentation équilibrée des différents domaines et perspectives de la

2. Le consortium DARIAH-CH, constitué en novembre 2021, regroupe plusieurs institutions académiques suisses afin de favoriser la recherche et l'enseignement numériques dans le domaine des sciences humaines et sociales. Il vise à coordonner les activités liées à DARIAH en Suisse et à assurer une participation active du pays au sein de DARIAH-EU. Voir *DARIAH-CH – Digital Research Infrastructure for the Arts and Humanities in Switzerland*, URL : <https://www.dariah.ch/> (visité le 12/07/2024).

recherche.

Un autre aspect fondamental de la gouvernance du DaSCH est son indépendance vis-à-vis de toute institution d'enseignement supérieur particulière. Cette autonomie permet d'éviter les conflits d'intérêts et d'assurer une prise de décisions objective et impartiale. En outre, l'indépendance du DaSCH renforce sa crédibilité et sa légitimité en tant que plateforme neutre pour la recherche en sciences humaines.

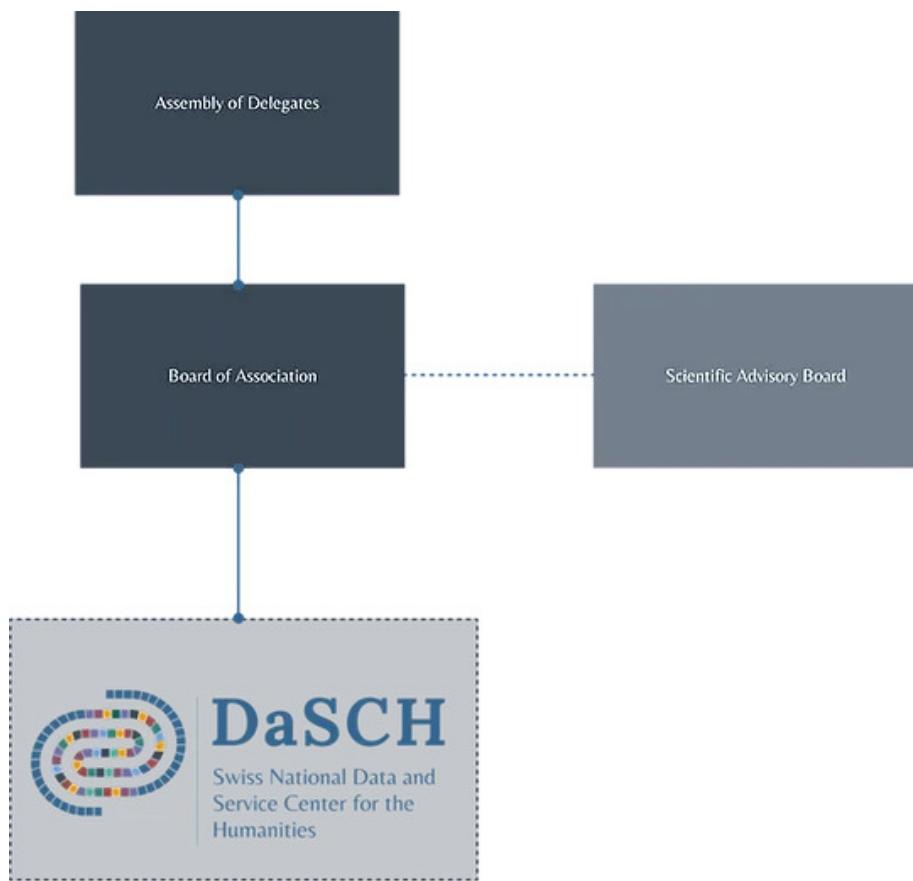


Figure 1.3 – Gouvernance du DaSCH

La gouvernance du DaSCH est également caractérisée par son efficacité et son efficience. En limitant les formalités administratives, l'association peut se concentrer sur ses missions essentielles, telles que la mise à disposition de ressources et de services pour la recherche³. Cette approche pragmatique

3. Selon le site web du DaSCH, l'association a « peu de formalités administratives ».

permet d'optimiser l'utilisation des ressources et de garantir une réponse rapide aux besoins des chercheurs.

La gouvernance participative et indépendante du DaSCH est l'un des piliers de son fonctionnement. Pour comprendre comment cette organisation est parvenue à mettre en place une telle gouvernance efficace et inclusive, il est essentiel de revenir sur ses origines et les raisons de sa création.

1.1.2 Fondation du DaSCH

Depuis sa création, DaSCH est le fruit d'une initiative conjointe du *Digital Humanities Lab* (DHLab) de l'Universität Basel et de la *Swiss Academy of Humanities and Social Sciences* (SAGW). Cette initiative a vu le jour dans le cadre d'un projet pilote ancré à l'*ERI (Education, Research and Innovation) Dispatch 2013–2016*.

L'origine du DaSCH remonte à un consortium réunissant les universités de Bâle, Berne et Lausanne, avec la direction assurée par le Digital Humanities Lab de l'Universität Basel. Ce consortium a remporté un appel d'offres public lancé par SAGW, marquant ainsi le début des efforts pour créer une infrastructure nationale dédiée aux données de recherche en sciences humaines. Pendant la phase pilote, une étude de faisabilité sur la solution technique choisie a été menée à bien, utilisant divers jeux de données.

En 2017, le DaSCH a été officiellement établi en tant qu'infrastructure nationale, opérée par SAGW, avec un mandat défini dans l'*ERI Dispatch 2017–2020*. Cette reconnaissance institutionnelle a permis au DaSCH de se structurer et de s'intégrer pleinement dans le paysage national de la recherche en sciences humaines, en apportant des solutions innovantes pour la gestion et la préservation des données de recherche.

Depuis 2021, le DaSCH fonctionne comme une infrastructure nationale de données de recherche, bénéficiant principalement d'un financement de la part du *Swiss National Science Foundation* (SNSF). Cette évolution témoigne de l'importance croissante des humanités numériques et de la reconnaissance de

Voir Id., *Governance / DaSCH*, URL : <https://www.dasch.swiss/governance> (visité le 12/07/2024).

la nécessité de disposer de structures robustes et durables pour le archivage, l'accès et l'exploitation des données de recherche dans ce domaine.



Figure 1.4 – Schéma de fonctionnement du DaSCH

Ainsi, la fondation du DaSCH illustre une réponse concertée aux besoins émergents des sciences humaines à l'ère du numérique, en mettant en place des infrastructures capables de soutenir les chercheurs dans leurs travaux et de valoriser les données produites par leurs recherches. Fort de ce socle, le DaSCH a mis également en place une plateforme numérique qui va bien au-delà du simple archivage de données. Nous allons donc aborder par la suite comment fonctionne cette plateforme.

1.1.3 DaSCH et l'*Open Data* en Suisse

DaSCH possède une plateforme pour les données de recherche ouvertes dans les sciences humaines en Suisse. Cet environnement de recherche virtuel peut être utilisé tant pour ceux et celles qui veulent publier ces données qu'

pour ceux et celles qui veulent réutiliser ses données⁴. C'est aussi important souligner que DaSCH développe et exploite un dépôt à long terme conforme aux principes FAIR (*Findable, Accessible, Interoperable, Reusable*). Pour ces raisons, nous pouvons voir que DaSCH s'assure que les informations collectées soient non seulement accessibles et réutilisables par la communauté scientifique, mais aussi qu'elles respectent les normes internationales de gestion de données. Cela permet non seulement de préserver le patrimoine scientifique de manière durable, mais aussi de faciliter la collaboration et l'innovation au sein de la communauté académique.

En deuxième place, l'environnement de recherche virtuel proposé par DaSCH offre aux chercheurs une plateforme intégrée qui simplifie l'archivage, la gestion et le partage de leurs données. Cette infrastructure est essentielle pour promouvoir la transparence et l'efficacité dans les processus de recherche, en assurant que les données sont disponibles et exploitables à long terme. Cela va de soi avec l'idée de persistance de données, une fois que ce processus d'archivage permet une identification de données.

Néanmoins, l'archivage de données en soi n'assure pas une persistance de données lors de l'interopérabilité. Pour montrer comment ces données sont interprétées et comprises, ainsi que la manière dont la communauté est intégrée, nous allons maintenant analyser les stratégies de DaSCH.

1.1.4 Stratégies de DaSCH

La stratégie de DaSCH repose sur trois piliers fondamentaux, chacun jouant un rôle essentiel dans la réalisation de ses objectifs à court et long terme⁵. Le premier pilier, l'extension technique de la plateforme, est une réponse directe aux besoins croissants d'évolutivité et de flexibilité dans la gestion des données de recherche. En rendant la plateforme plus accessible et adaptable, DaSCH vise à offrir aux chercheurs un outil performant et intuitif,

4. Voir *Re3Data Repository : Swiss National Data and Service Center for the Humanities*, URL : <https://www.re3data.org/repository/r3d100012374> (visité le 15/07/2024).

5. Les trois piliers mentionnés sont expliqués dans la partie *Strategic Goals* sur le site. Voir Id., *Strategic Goals / DaSCH*, URL : <https://www.dasch.swiss/strategic-goals> (visité le 12/07/2024).

tout en permettant l'intégration de nouvelles technologies. Cela ne se limite pas à l'amélioration de l'expérience utilisateur, mais s'étend également à l'augmentation de l'interopérabilité en encourageant l'adoption de standards reconnus et de données normées. Cette approche permet non seulement de faciliter la réutilisation des données, mais aussi d'assurer leur pérennité dans un environnement de recherche en constante évolution.

Le deuxième pilier de la stratégie de DaSCH est le soutien à la création de communauté et à l'éducation. DaSCH reconnaît que la technologie seule ne suffit pas à garantir le succès de sa plateforme. En connectant les chercheurs entre eux et en leur offrant des opportunités de formation, DaSCH renforce les liens au sein de la communauté scientifique. Cela inclut non seulement l'organisation d'événements pour échanger sur les besoins et les défis rencontrés, mais aussi l'offre de formations adaptées, en particulier pour les jeunes chercheurs. Cette orientation vers l'éducation et le soutien communautaire est cruciale pour garantir que la plateforme DaSCH soit non seulement utilisée, mais aussi optimisée et améliorée grâce aux retours de ses utilisateurs, assurant ainsi une évolution en phase avec les besoins réels des chercheurs.

Enfin, le troisième pilier consiste en un renforcement des réseaux et des coopérations, tant au niveau national qu'international. DaSCH comprend que la collaboration est essentielle pour avancer dans la recherche et la gestion des données. En s'associant avec d'autres institutions et en participant activement à des projets internationaux, DaSCH vise à créer des synergies qui permettent de développer des solutions innovantes pour l'accessibilité, l'interopérabilité et la conservation à long terme des données. L'engagement de DaSCH dans des initiatives comme DARIAH et la définition de standards IIIF pour les images 3D témoigne de sa volonté de jouer un rôle de premier plan dans la normalisation et l'amélioration des pratiques de gestion des données de recherche. Cette stratégie de coopération est non seulement bénéfique pour DaSCH, mais elle contribue également à l'avancement de la recherche en sciences humaines dans son ensemble.

Le développement et le maintien d'une infrastructure et des stratégies aussi complexe que DaSCH nécessitent des investissements conséquents et durables. Nous allons expliquer alors comment s'opère le financement de

l'association.

1.1.5 Financement du DaSCH

Concernant le financement du DaSCH, l'association bénéficie d'un financement principal du SNSF, qui a octroyé un soutien significatif pour la période 2021 à 2024, intitulé *ERI Dispatch 2021-2024*. Cet accord de financement a permis de définir clairement les services que le DaSCH doit fournir à la communauté scientifique suisse. En plus de ce financement public, l'Universität Basel apporte des contributions en nature, telles que l'hébergement de l'infrastructure et la mise à disposition de ressources humaines.

Ces ressources financières permettent au DaSCH de développer et de maintenir une infrastructure numérique. Cependant, pour étendre ses services et répondre à des demandes spécifiques, le DaSCH peut également solliciter des financements supplémentaires auprès de tiers ou des utilisateurs eux-mêmes. Ces contributions peuvent prendre la forme de partenariats, de projets de recherche ou de frais d'utilisation de certains services.

Nous allons maintenant nous pencher plus en détail sur l'*ERI Dispatch* pour comprendre comment il fonctionne et pourquoi il est important pour comprendre les enjeux du DaSCH dans ce contexte de persistance de données.

Fonctionnement de l'*ERI Dispatch*

Le système politique fédéral suisse considère que les investissements dans l'éducation, la recherche et l'innovation sont essentiels pour le succès du pays. C'est pourquoi des investissements sont réalisés dans ces domaines, car il est estimé que l'éducation et la recherche sont les fondements de la créativité et de l'entrepreneuriat. Ces éléments sont également des prérequis pour la capacité d'innovation et la compétitivité des entreprises suisses. Afin de soutenir les initiatives dans ces trois domaines, la Confédération suisse et les cantons financent et régulent le secteur de l'ERI par le biais du plan l'*ERI (Education, Research and Innovation) Dispatch*, accordé tous les quatre ans,

afin de promouvoir ces initiatives⁶.

Il est important de noter que le plan *ERI Dispatch* est soumis à l'Assemblée fédérale avant d'être approuvé. Pour la période 2021-2024, à laquelle DaSCH est associé, un investissement de 28,1 milliards de CHF a été prévu, soit plus de 2 millions de CHF par rapport au plan l'*ERI (Education, Research and Innovation) Dispatch 2017-2020*, qui est à l'origine de DaSCH⁷.

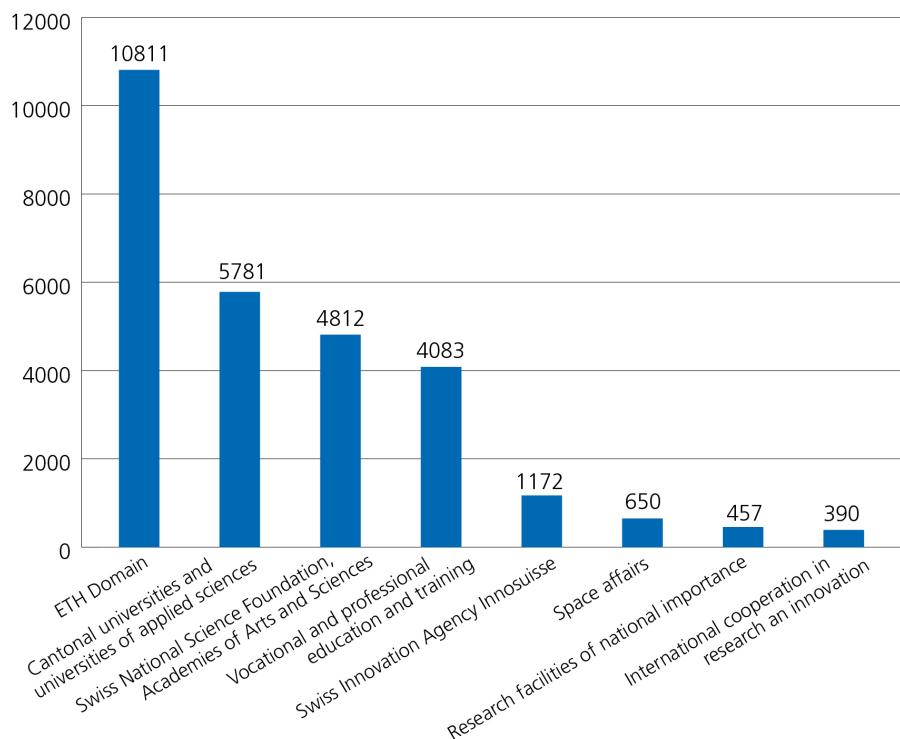


Figure 1.5 – Financement *ERI Dispatch* alloué par l'Assemblée fédérale suisse pour 2021-2024, en millions. **Source :** State Secretariat for Education, Research and Innovation : Education, Research, and Innovation ERI 2021–2024, URL : <https://www.sbf.admin.ch/sbfi/en/home/eri-policy/eri-21-24.html> (visité le 17/07/2024).

Enfin, il est également crucial de souligner que pour la période 2021-2024,

6. Le caractère cyclique des évaluations ERI peut conduire à des fluctuations dans l'importance accordée aux indicateurs DaSCH, ce qui peut créer une certaine instabilité dans les politiques publiques associées.

7. Voir State Secretariat for Education, Research and Innovation : Education, Research, and Innovation ERI 2021–2024, URL : <https://www.sbf.admin.ch/sbfi/en/home/eri-policy/eri-21-24.html> (visité le 17/07/2024).

les politiques du plan l'*ERI (Education, Research and Innovation) Dispatch* se concentrent sur trois thèmes principaux : la numérisation, le développement durable et l'égalité des chances. Ces politiques bénéficient de l'engagement des cantons et de tous les parties prenantes concernées⁸.

1.1.6 Notes conclusives

En conclusion, DaSCH joue un rôle crucial dans la transformation numérique des sciences humaines en Suisse, en fournissant les outils et les services nécessaires pour une gestion optimale des données de recherche ouvertes, tout en respectant les plus hauts standards de qualité et de durabilité. Nous avons également constaté que pour DaSCH, le standard IIIF occupe une place centrale dans ses préoccupations en matière de maintien de la persistance des données. C'est pourquoi nous allons maintenant nous pencher sur ce qu'est le standard IIIF, afin de comprendre l'importance qu'il a revêtue lors de la mise en œuvre des *pipelines* au cours de mon stage.

8. Voir *Ibid.*

Chapitre 2

Contexte IIIF

2.1 Qu'est-ce que l'IIIF ?

Le standard *International Image Interoperability Framework* est un ensemble de normes ouvertes conçues pour la gestion des objets numériques. Soutenues par une communauté internationale, les APIs IIIF constituent un élément fondamental en permettant l'ajout d'informations descriptives aux objets numériques. Cela assure non seulement que ces informations soient accessibles et compréhensibles pour les utilisateurs humains (*human readable*), mais facilite également l'affichage des images. De plus, la communauté IIIF s'efforce constamment de maintenir et d'améliorer ces APIs, garantissant ainsi leur compatibilité avec une large variété d'objets 2D. De plus, avec l'introduction de la version 4.0, des discussions sont en cours pour étendre ce support aux modèles 3D, ouvrant ainsi de nouvelles perspectives dans la gestion des ressources numériques.

Avant de nous plonger plus profondément dans les aspects techniques de l'API 3.0, il est pertinent de comprendre l'histoire de son développement, ce qui nous permet de mieux saisir les enjeux actuels et les futures évolutions envisagées par la communauté.

2.2 Origines d'IIIF

L'origine d'IIIF daterait environ du début des années 2010, à partir d'une discussion informelle entre technologistes de *Stanford University*, *Oxford University* et de la *British Library* dans un café, et, par la suite, différentes bibliothèques nationales et internationales ont rejoint l'initiative¹. IIIF est actuellement un *consortium* international qui rassemble différentes organisations du monde et qui a également une communauté de musées, archives, bibliothèques, instituts de recherche, sociétés de services informatiques qui entretiennent les APIs IIIF.

La communauté IIIF a pour mission de définir, de diffuser et de faire évoluer les normes techniques IIIF en s'appuyant sur des cas d'usage concrets. C'est dans cette perspective que l'ambition de IIIF est de créer un cadre technique commun, permettant aux fournisseurs de ressources numériques de délivrer leurs contenus de manière standardisée sur le Web, afin de les rendre consultables, manipulables et annotables par n'importe quel logiciel compatible².

2.3 Intégration d'IIIF pour la valorisation du patrimoine

Le standard IIIF (International Image Interoperability Framework) s'est imposé comme un outil incontournable pour l'archivage et la diffusion d'objets numériques au sein des institutions patrimoniales. En associant un objet numérique à un fichier JSON, ce standard offre une méthode simple et flexible

1. Les deux versions sont d'accord sur les années du début, néanmoins c'est la thèse de Raemy qui raconte qui étaient les acteurs principaux et les institutions lors de la création d'IIIF. Pour les deux versions, voir la thèse de Raemy J.A. Raemy, *The International Image Interoperability Framework (IIIF) : raising awareness of the user benefits for scholarly editions*, Accessed : 21 May 2024, Bachelor's thesis, HES-SO University of Applied Sciences et Arts, Haute école de gestion de Genève, 2017, URL : <https://sonar.ch/hesso/documents/314853>, p. 13 et le site de Biblissima, Voir Biblissima Documentation, *Introduction à IIIF*, URL : <https://doc.biblissima.fr/iiif/introduction-iiif/#vision> (visité le 13/08/2024)

2. Voir *Ibid.*

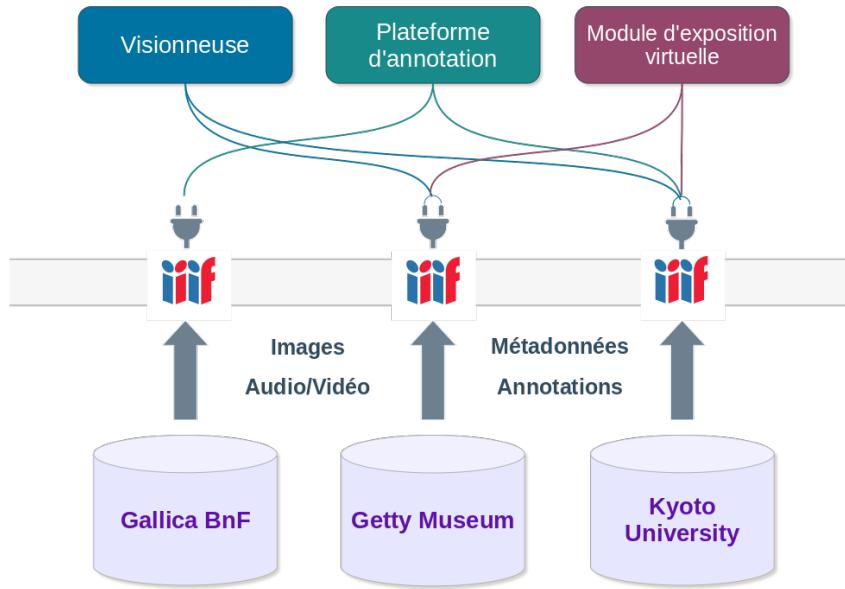


Figure 2.1 – Schéma : principe général d'interopérabilité de IIIF (trois applications différentes sont branchées à trois entrepôts IIIF).

pour décrire et rendre accessible un large éventail de ressources.

Dans un premier temps, il est essentiel de souligner l'adoption généralisée du IIIF par de nombreuses institutions culturelles de renommée mondiale, telles que la Bibliothèque nationale de France ou les Musées du Vatican. Cette adhésion massive témoigne de l'efficacité du standard pour répondre aux besoins spécifiques des institutions patrimoniales, en particulier en matière de préservation et de valorisation des collections.

Ensuite, la popularité du IIIF s'explique par sa capacité à rendre les objets numériques et leurs métadonnées accessibles à un large public grâce à des *viewers* en ligne, qui sont des outils pour rendre les modèles 3D visibles sur le web, tels Mirador, Universal Viewer et OpenSeadragon³. Ces outils permettent également d'explorer les collections de manière interac-

3. Mirador, Universal et OpenSeadragon sont trois projets *open source* qui permettent de visualiser ces modèles 3D. Pour plus de détails sur ces viewers, Voir *Mirador Project*, URL : <https://projectmirador.org/> (visité le 29/08/2024), *Universal Viewer*, URL : <https://universalviewer.io/> (visité le 29/08/2024) et *OpenSeadragon IIIF TileSource Example*, URL : <https://openseadragon.github.io/examples/tilesource-iiif/> (visité le 04/09/2024)

2.3. INTÉGRATION D'IIIF POUR LA VALORISATION DU PATRIMOINE

tive, de réaliser des comparaisons, d'annoter les images et voire de zoomer en haute définition⁴. Cette dimension visuelle est particulièrement appréciée des chercheurs et du grand public, qui peuvent ainsi bénéficier d'une expérience utilisateur enrichie. Voyons un exemple avec Mirador⁵ :

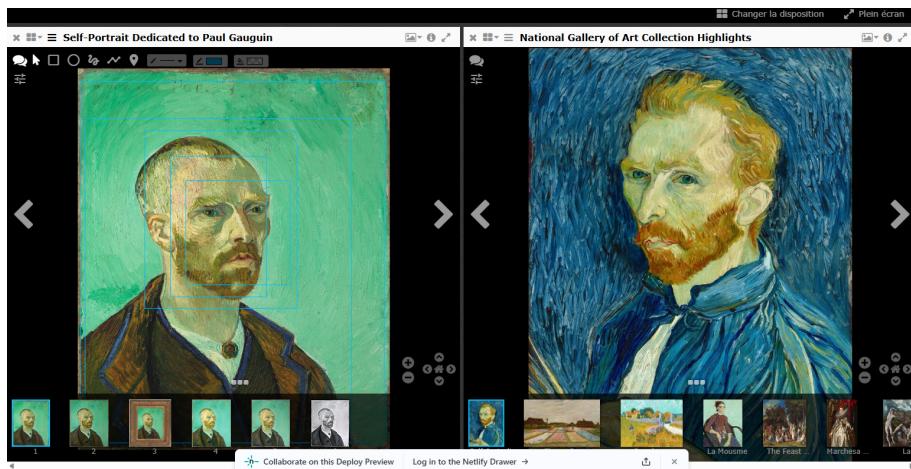


Figure 2.2 – Mirador : comparaison entre images de différentes institutions.

Par ailleurs, le IIIF offre une grande flexibilité dans la structuration des données. Il est ainsi possible d'archiver des objets avec un minimum d'informations, ce qui facilite l'intégration rapide de nouvelles acquisitions dans les collections. De plus, le standard permet de décrire des objets complexes et de gérer des ensembles d'images multiples, favorisant ainsi une approche holistique de l'archivage.

En outre, le IIIF joue un rôle essentiel dans la diffusion d'images haute résolution. En réduisant les contraintes liées au stockage et à la bande passante, il permet de partager des images de grande qualité sans compromettre les performances. Cette caractéristique est particulièrement intéressante pour les institutions qui souhaitent mettre à disposition du public des reproductions numériques de haute fidélité de leurs œuvres.

4. Alors que le texte de Rossenova vise un large public, il faut souligner que les *viewers* permettent ces autres possibilités. Voir Lozana Rossenova, *IIIF for 3D – making web interoperability multi-dimensional*, 2023, URL : %7Bhttps://pro.europeana.eu/post/iiif-for-3d-making-web-interoperability-multi-dimensional%7D (visité le 14/05/2024).

5. Pour regarder la démonstration de Mirador, Voir *Mirador Demo*, URL : <https://deploy-preview-13--mirador.netlify.app/demo/> (visité le 04/09/2024).

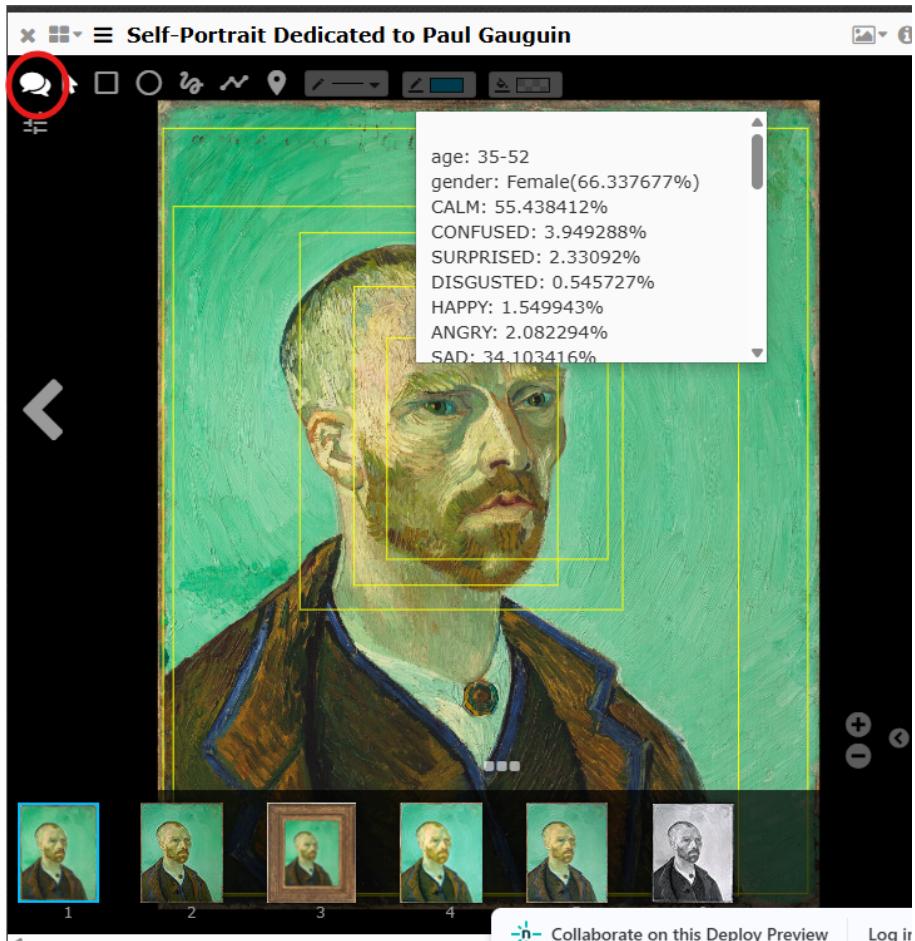


Figure 2.3 – Mirador : exemple d’image annotée, avec l’outil d’annotations du *viewer*.

Enfin, l’une des forces du IIIF réside dans sa capacité à favoriser l’interopérabilité entre les différentes institutions. Grâce aux manifestes IIIF, il est possible de créer des liens entre des objets provenant de sources diverses et de construire des narrations entre eux. Cette fonctionnalité ouvre de nouvelles perspectives pour la recherche et la collaboration entre les institutions culturelles.

En conclusion, le standard IIIF constitue une solution efficace et flexible pour l’archivage et la diffusion d’objets numériques. En offrant une structure simple et ouverte, il permet aux institutions patrimoniales de préserver leur patrimoine et de le rendre accessible à un public toujours plus large. Cepen-

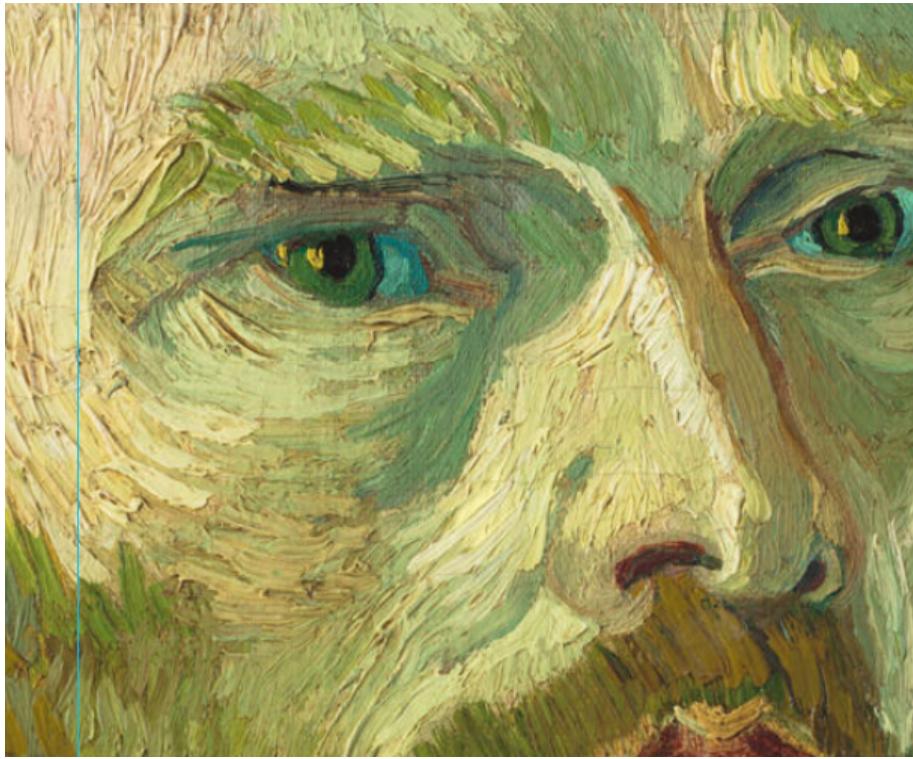


Figure 2.4 – Mirador : zoom en haute définition dans un détail de la première image.

dant, l’exploitation optimale du IIIF nécessite une réflexion approfondie sur les enjeux liés à la gestion des données, à l’interopérabilité et à la pérennité des infrastructures numériques.

2.4 IIIF API 3.0.

L’importance d’expliquer l’API 3.0 réside dans le fait qu’elle est la plus utilisée par les institutions travaillant dans le domaine patrimonial. De plus, elle constitue la base pour la conception de l’API 4.0. Il est donc fondamental de bien comprendre les principes de l’API 3.0 pour saisir non seulement son importance actuelle, mais aussi les raisons pour lesquelles elle a été essentielle dans le cadre de mon stage.

L’API IIIF 3.0 se compose de quatre API RESTful au format JSON-LD : Image API, Presentation API, Content Search API et Authentification API.

Pour bien comprendre chacune de ces API dans notre étude, nous allons en expliquer le fonctionnement et analyser ensuite les enjeux pour DaSCH avec les applications développées durant le stage, ce qui nous aidera à comprendre l'importance future de l'API 4.0 pour notre travail.

Image API

La première version de l'*Image API 3.0* a été lancée en août 2012⁶ et la fonction basique de cette API est de prendre les données techniques d'une image et ses informations associées, pour les rendre visibles sur le web⁷. Pour ce faire, elle requiert une URI définie selon une syntaxe stricte (Voir Tableau 2.1)⁸.

Table 2.1 – Structure des requêtes d'image et d'information

Type de Requête	Structure de la Requête
Image Request	{scheme} ://{server}/{prefix}/{identifier}/{region}/{size}/{rotation}/{quality}.{format}
Exemple Image Request	http://www.example.org/image-service/abcd1234/full/full/0/default.jpg
Image Information Request	{scheme} ://{server}/{prefix}/{identifier}/info.json
Exemple Information Request	http://www.example.org/image-service/abcd1234/info.json

Comme nous pouvons le constater, les informations sont prises dans l'ordre indiqué dans le tableau. Il est important de mentionner que l'API ne pose aucune restriction quant à la forme des identifiants qu'un serveur peut utiliser ou prendre en charge, ce qui offre une grande flexibilité, même dans la conception des identifiants.

L'image renvoyée est toujours la représentation de l'URI, ce qui signifie que l'URI peut modifier la nature de l'image affichée. Cela est crucial pour certaines institutions, car des paramètres comme la taille en *pixels*, la couleur, ou d'autres aspects peuvent être ajustés en fonction des besoins spécifiques.

6. Voir Pour ces prochains chapitres, nous avons suivi l'ordre chronologique explicatif suggéré par les travaux de Julien Raemy. Voir J. Raemy, *The International Image Interoperability Framework (IIIF) : raising awareness of the user benefits for scholarly editions...*

7. Tous les concepts du chapitre ont été extraits et analysés à partir de la partie dédiée au *Image API 3.0* sur le site d'IIIF. Voir *IIIF Image API 3.0*, URL : <https://iiif.io/api/image/3.0/> (visité le 29/08/2024).

8. L'exemple de syntaxe a été extrait de la section d'Image API sur le site d'IIIF. Voir *Ibid*. En ce qui concerne l'idée de faire un tableau avec les requêtes, ce travail s'est inspiré de la suggestion de Raemy. Voir Id., *The International Image Interoperability Framework (IIIF) : raising awareness of the user benefits for scholarly editions...*, p. 20.

Presentation API

Le *Presentation API* travaille avec des objets numérisés ou nativement numériques et son objectif principal est de fournir les informations nécessaires pour permettre une visualisation riche en ligne des objets numériques composés, souvent en combinaison avec l'*Image API 3.0*⁹. Ces objets sont standardisés suivant une description en format JSON. Un objet dit composé peut être une série de pages, de surfaces, etc. Par exemple, il peut s'agir des deux faces d'une photographie, des quatre vues cardinales d'une statue, de papyrus, etc. Toutes ces spécifications suivent les principes du Linked Data et de l'architecture du Web pour garantir leur interopérabilité. De plus, le modèle de données Shared Canvas et JSON-LD permet de créer un format JSON facile à mettre en œuvre.

Pour bien comprendre le fonctionnement du *Presentation API* d'IIIF, il est essentiel de savoir comment les types de ressources sont classés dans la spécification des objets composés. Le modèle est structuré de la manière suivante : Collection, Manifest, Canvas et Range¹⁰.

Premièrement, une Collection est une liste ordonnée de Manifests et/ou d'autres Collections. Les Collections permettent de regrouper des Manifests et d'autres Collections dans une structure hiérarchique pour la présentation. Cette présentation peut être utilisée pour naviguer, afficher des résultats de recherche dynamiques ou fournir un ensemble de ressources fixes, quel que soit l'objectif¹¹.

Deuxièmement, le Manifest décrit la structure et les propriétés d'un objet composé. Il contient les informations nécessaires pour qu'un client puisse afficher le contenu à un utilisateur, tels que le titre, les annotations ou les éléments intellectuels pertinents à l'objet. Bien qu'un Manifest peut représenter un seul objet (une image, donc), il est courant qu'un Manifest décrive un seul objet composé, comme un livre, un album musical, un objet archéologique,

9. Tous les concepts du chapitre ont été extraits et analysés à partir de la partie dédiée au *Presentation API 3.0* sur le site d'IIIF. Les sources seront citées à chaque fois que cela sera nécessaire pour garantir la clarté sur les références. Voir *IIIF Presentation API 3.0*, URL : <https://iiif.io/api/presentation/3.0/> (visité le 20/07/2024).

10. Voir *Ibid.*

11. Voir *Ibid.*

etc¹².

Troisièmement, le Canvas est un espace virtuel, une sorte de conteneur, dont la fonction est de représenter une vue particulière d'un objet, associée à des ressources de contenu ou à des parties de celui-ci. Le Canvas sert principalement de référence pour la mise en page du contenu, tant sur le plan spatial que temporel. Le concept de Canvas est dérivé de normes telles que PDF et HTML, ainsi que de logiciels comme PHOTOSHOP et POWERPOINT, où une surface de départ accueille des images, des vidéos, du texte et d'autres contenus « peints » par annotations, rassemblés dans des pages d'annotation. Les annotations peuvent être affichées ou non, et cette archive associée peut varier, ce qui signifie qu'un même objet composé peut avoir plusieurs Canvas pour le représenter¹³.

Finalement, un Range est une liste ordonnée de Canvas et/ou d'autres Ranges. Les Ranges permettent de regrouper des Canvas, ou des sections de celles-ci, de manière structurée. Cela peut être motivé par des raisons liées au contenu, comme celles présentées dans une table des matières ou dans la succession des scènes d'une pièce de théâtre. De même, des caractéristiques physiques peuvent jouer un rôle important, par exemple la compilation de pages dans un livre, ou lorsque de la musique enregistrée est répartie sur différents supports physiques, tels que deux CD¹⁴.

2.5 Vers l'IIIF API 4.0

Actuellement, la communauté IIIF développe le *Presentation API 4.0* afin de gérer les modèles 3D¹⁵. Cette importance croissante provient du fait que les collections numériques, de plus en plus nombreuses, nécessitent non seulement des traitements spécifiques pour gérer leur capacité d'archivage et

12. Voir *Ibid.*

13. Voir *Ibid.*

14. Voir *Ibid.*

15. Toutes les discussions sur le développement de l'API peuvent être trouvées sur le GitHub de l'*International Image Interoperability Framework*. Voir *3D Community Group - IIIF*, URL : <https://github.com/IIIF/3d> (visité le 03/08/2024). Pour plus de détails sur le groupe, les buts et l'organisation, Voir *Technical Specification Group - IIIF 3D Community*, URL : <https://iiif.io/community/groups/3d/tsg/> (visité le 29/08/2024).

de rendu, mais aussi parce que d'autres types d'objets gagnent en importance dans le contexte de la recherche. Au-delà des œuvres d'art et des collections naturalistes, les objets archéologiques, les plans de sites et même les modèles 3D d'architectures historiques gagnent en importance dans le contexte de la recherche, comme c'était le cas chez DaSCH.

De plus, le contexte de production des objets est mis en évidence lors de leur numérisation. C'est pourquoi le *Presentation API 4.0* a été conçu pour répondre à ces nouvelles demandes et proposer de nouvelles façons de valoriser les archives. Cela se remarque déjà à partir du remplacement d'un concept clé pour le *Presentation API 3.0* : nous parlons dorénavant de « Scene », pas de « Canvas » ; le fichier JSON qui définissait les fichiers comme « Canvas », maintenant définit les objets comme une seule « Scene » ou un ensemble de « Scenes ». Ces paramètres minimaux explicités par les développeurs, comme Julie Winchester¹⁶, sont documentés et on les abordera lors de l'explication du code DaSCH IIIF.

En outre, de nouvelles options de description ont été ajoutées, permettant au *Presentation API 4.0* de proposer de nouvelles interactions avec les objets, comme par exemple les caméras et les lumières, qui changent le comportement des objets lors de l'affichage sur des *viewers*¹⁷.

C'est donc dans ce contexte de création de nouvelles formes de valorisation du patrimoine et de recherche de solutions adaptées à ces nouveaux besoins que DaSCH s'intéresse à répondre, en priorité, aux besoins de la communauté travaillant avec ces différents objets, mais aussi à envisager des stratégies pour s'adapter au futur API 4.0. Quelles informations sont pertinentes ? Ce sont des questions qui seront analysées dans le quatrième chapitre de ce travail. Avant de procéder à cette analyse, nous avons encore besoin de quelques

16. Julie Winchester est la Directrice technique (*Technical director*) pour MorphoSource, un dépôt pour les *scans* 3D d'échantillons biologiques et d'autres objets physiques, et a été nommée l'éditrice de spécifications (*specification Editor*) d'IIIF en janvier 2024 *IIIF welcomes Julie Winchester as specification Editor*, URL : <https://iiif.io/news/2024/01/23/welcome-julie/> (visité le 21/07/2024).

17. Les « Scenes » offrent aussi des dimensions infinies (X, Y, Z ou polaires), une origine centrale et permettent d'intégrer divers éléments (images, modèles 3D, etc.), Petros Patias (éd.), *3D Research : Challenges in Cultural Heritage III : Complexity and Quality in Digitisation*, Springer, 2023, p. 64.

définitions théoriques fondamentales pour mieux comprendre le problème auquel DaSCH est confronté.

Chapitre 3

Persistante des données

La manière dont nous concevons et représentons les données est fondamentale. Les normes et standards, souvent sources de débats terminologiques, influencent directement notre capacité à préserver et à diffuser les données. Ce chapitre explorera les différentes notions utilisées pour définir les "choses" que nous représentons en données numériques ou sont déjà nées numériques, et les implications de ces choix sur la manière dont nous les codons et les conservons. Cette réflexion vise également à repenser la notion de durabilité des données.

3.1 Document, artefact et données

Bien qu'il soit vrai, comme le soulignait Marc Bloch, qu'un seul document ne suffit pas à faire l'histoire¹, il est tout aussi exact d'affirmer, avec Langlois et Seignobos, que sans document, il n'y a pas d'histoire². Ces deux affirmations révèlent la nature intrinsèquement historique des documents, quels qu'ils soient. De plus, les documents, outre leur diversité, s'inscrivent pleinement dans la culture matérielle, au même titre que les objets. Pour mieux comprendre ces implications, il convient de réviser la définition tradi-

1. M. Bloch, *Apologie pour l'histoire ou métier d'historien*, t. 3, Librairie Armand Colin, 1952 (Cahier des Annales), p. 37-42.

2. Charles-Victor Langlois et Charles Seignobos, *Introduction aux études historiques*, Editions Kimé, 1ère édition : Librairie Hachette, Paris, 1898, Paris, 1992, p. 284, p. 13

tionnelle de document.

Lors d'une conférence, l'archéologue Ulpiano Meneses nous rappelle que le terme « document » vient de la même racine du mot latin *doceo*, qui signifie enseigner. Or, enseigner implique la transmission d'un savoir établi, ce qui fait du mot *documentum* un modèle, un paradigme préétabli³. Cependant, Ulpiano nous met en garde contre cette conception trop restrictive du document comme simple source d'information⁴.

Ainsi, pour dépasser cette vision, nous devons élargir la notion de document au-delà de la simple idée d' « enregistrer et conserver une information ». En effet, si nous nous limitons à cette définition, nous risquons de réduire le document à un texte écrit. Or, un document peut également être une source orale, mais aussi un objet. Cette idée, bien qu'elle ne soit pas nouvelle, mérite d'être rediscutée parce qu'elle met en lumière le fait que les conceptualisations peuvent être assez floues pour un document : les documents dont la nature n'est pas nécessairement d'être conservés, comme les lettres⁵, ou encore, comme mentionné, la restriction de source écrite à la notion de document. Selon Meneses, les artefacts, tout ce qui est produit par l'action humaine, sont l'un des principaux composants de la culture matérielle, voire des produits et des vecteurs de relations sociales. Selon l'auteur, ils sont à la fois issus de modes d'organisation sociale spécifiques et historiquement déterminés, et orientent et facilitent les relations sociales dans certaines directions⁶.

Par ailleurs, il est essentiel de prendre en compte la dimension temporelle inhérente non seulement aux documents, mais aussi à leur conservation. En effet, un même document peut contenir des temporalités différentes,

3. Ulpiano Bezerra de Meneses, *O objeto como documento*, 1980, URL : https://edisciplinas.usp.br/pluginfile.php/5577860/mod_resource/content/1/MENESES%2C%20Ulpiano%20Bezerra%20de.%200bjeto%20como%20documento.%201980.pdf (visité le 29/08/2024), p. 1-3.

4. Un travail sur la philosophie de l'information que serait intéressant d'aborder dans un étude futur, c'est la formulation de la théorie sémantique de l'information sémantique de Floridi. Voir Chapitre 4 e Chapitre 5 Luciano Floridi, *The Philosophy of Information*, Oxford University Press, Oxford, 2011.

5. U. B. de Meneses, *O objeto como documento...*, p. 2.

6. Id., « A cultura material no estudo das sociedades antigas », *Revista de História*, 115 (août 1983), URL : <https://www.revistas.usp.br/revhistoria/article/view/61796> (visité le 29/08/2024), p. 112-113.

c'est-à-dire des traces de différentes époques. Comme le rappelle Ulpiano en citant Radley, « artefacts survive in ways unintended by makers and owners »⁷. La numérisation de ces documents n'est donc pas une simple opération technique de conservation, mais un véritable travail de mémoire. Ce travail consiste à rendre compte des différentes strates temporelles contenues dans un document et à en tirer des informations sur les dynamiques sociales. La manière dont nous conservons un document aujourd'hui nous renseigne peut-être moins sur la société qui l'a produit.

Enfin, comme le souligne le poète brésilien Carlos Drummond de Andrade dans son poème *A Suposta existêcia*, « Les choses existent-elles/si elles ne sont pas vues ? »⁸. Ainsi, nous devons nous interroger sur ce que nous choisissons de conserver. En effet, la sélection historique des documents à numériser implique une prise de position qui reflète notre conception de ce qui est important de transmettre, de documenter.

Ainsi, la culture matérielle nous sert à penser la notion de document et, par conséquent, de donnée.

3.2 Culture matérielle et la persistance de données

D'après Krzysztof Pomian, étudier l'histoire des musées permet comprendre les sociétés dans lesquelles s'est inscrite cette institution⁹. Cet argument est aussi éloquent en ce qui concerne la production des données numériques, étant donné que produire des données est aussi une forme de collectionner au sens premier du mot, de collecte. Étudier la formation d'une collection renvoie non seulement à la compréhension de l'origine d'une institution, y compris dans le système de pensée déjà créée dans lequel l'objet

7. La phrase peut être traduite comme « les artefacts survivent d'une manière que leurs créateurs et leurs propriétaires n'avaient pas prévue ». Id., « Memória e Cultura Material : Documentos Pessoais no Espaço Público », *Revista de História*, s/n (août 1998), URL : <https://periodicos.fgv.br/reh/article/view/2067> (visité le 29/07/2024), p. 90.

8. Id., *O objeto como documento...*, p. 1.

9. Krzysztof Pomian, *Le musée, une histoire mondiale. Tome I : Du trésor au musée*, Gallimard, Paris, 2020, p. 18.

30.3.2. CULTURE MATÉRIELLE ET LA PERSISTANCE DE DONNÉES

s'inscrit, mais aussi des « choses » dans un contexte historique précis.

Réfléchir comme nous concevons une collection peut constituer une source historique de la culture matérielle. Pour discuter la notion que nous intéresser, il est important de dire que la construction de cette théorie a été faite de différentes manières par le monde, sans suivre un programme strict. À partir de l'ouvrage *The socialness of things : essays on the socio-semiotics of objects* (1994) de Stephen H. Riggins, Daniel Miller a déjà souligné que les études à propos des « formes culturelle » sont issues des « analyses structurales et sémiotiques » dont les principales références étaient Roland Barthes, Jean Baudrillard, Mary Douglas et Claude Lévi-Strauss, aussi que Le système des objets (1968), écrit par Jean Baudrillard, a été une référence essentielle pour l'élaboration de la notion de culture matérielle dans la sémiologie¹⁰.

La notion de culture matérielle qui nous intéresse vient, d'abord, de la théorie de la circulation des objets chez Igor Kopytoff. En fait, le principe selon lequel la marchandisation des choses n'est qu'une phase dans la biographie des choses¹¹, présupposé soutenu aussi par Appadurai dans l'article « Introduction : commodities and the politics of value »¹², est fondamental pour ce travail. En effet, dans la théorie classique de la circulation marxienne¹³, la marchandisation, bien que nuancée entre les différents acteurs de l'échange, tend à être un moment essentiel des sociétés capitalistes¹⁴. Braudel souligne

10. Voir Ann Smart Martin, « Material Things and Cultural Meanings : Notes on the Study of Early American Material Culture », *The William and Mary Quarterly* (, 1996), Consulté le 20 novembre 2021, DOI : <https://doi.org/10.2307/2946821> ; Daniel Miller, *Material Cultures : Why Some Things Matter*, Armand Colin, London, 1988 ; Keith Moxey, « Les études visuelles et le tournant iconique », *Intermediality : History and Theory of the Arts, Literature and Technologies*-11 (2008), Consulté le 20 novembre 2021, URL : <https://www.erudit.org/fr/revues/im/2008-n11-im3117/037542ar/> ; Marcelo Rede, « Estudos de cultura material : uma vertente francesa », *Anais do Museu Paulista : História e Cultura Material*, 8 (2001), Mis en ligne le 24 août 2009, consulté le 12 avril 2021, DOI : <https://doi.org/10.1590/S0101-47142001000100008>.

11. Igor Kopytoff, *The Social Life of Things : Commodities in Cultural Perspective*, dir. Arjun Appadurai, Cambridge University Press, Cambridge et New York, 1986, p. 83.

12. A. Appadurai, *The Social Life of Things : Commodities in Cultural Perspective*, dir. Arjun Appadurai, Cambridge University Press, Cambridge et New York, 1986, p. 16-17.

13. On utilise le mot marxienne pour les théories venant directement de Marx, tandis que le mot marxiste se rapporte à ceux qui ont théorisé Marx par la suite.

14. Marx ne remet pas en cause le mécanisme de l'échange en tant que tel, mais plutôt son caractère spécifique et dominant dans le mode de production capitaliste. K. Marx, *Le Capital*, Gallimard, Traduit de l'allemand par Maximilien Rubel, première publication en

l'intérêt de la théorie de la circulation en montrant comment, au cours de l'histoire, différentes temporalités du marché (traditionnelle, archaïque, moderne) se sont entremêlées au sein de la « sphère de la circulation »¹⁵.

Dans cette perspective, la manière dont les objets nous parviennent n'est jamais neutre ni dénuée de médiations : ils nous renseignent sur des dynamiques sociales. Le moment où nous conceptualisons ces objets comme des données correspond à un croisement de différentes temporalités historiques. En appliquant cela à notre cas, les données nous indiquent les médiations qui ont façonné les objets et, par conséquent, la manière dont nous les numérisons et nous nous les approprions numériquement aura un impact sur leur nouvelle existence dans le monde numérique. Cette nouvelle circulation numérique engendrera une nouvelle conception de l'objet, susceptible d'évoluer au gré de nouvelles interactions.

De plus, la notion de persistance de données prend ici toute son importance. Ce concept fondamental en informatique permet de stocker des informations de manière durable, facilitant ainsi leur réutilisation par différents programmes, que ce soit simultanément ou à des moments différents¹⁶. Cette notion est particulièrement fondamentale dans les environnements de développement logiciel et de conception assistée par ordinateur. Nous pouvons étendre cette réflexion à notre étude pour comprendre comment cette dimension de la persistance s'est construite historiquement dans la production de données.

Pour cette raison, la réflexion sur les données ne doit jamais être dissociée d'une réflexion théorique approfondie sur le produit que nous sommes en train de construire. Comprendre les dynamiques sociales dans lesquelles il va s'insérer et les objectifs qu'il vise est essentiel dans ce processus de création collective qu'est la programmation. De plus, cette réflexion doit s'inscrire dans une perspective de *green coding*.

1867, Paris, 2008, p. 10.

15. C'est Marx qui a forgé le concept de « sphère de la circulation ». Fernand Braudel, *Civilisation matérielle, économie et capitalisme, XVe-XVIIIe siècle. Tome 2. Les jeux de l'échange*, Armand Colin, Paris, 1967, p. 11-12.

16. Voir M. Ancona, G. Nani et M. Paci, « An object oriented approach to data persistence », *Microprocessing and Microprogramming*, 32-1-5 (1991), p. 263-270, URL : 10.1016/0165-6074(91)90357-y (visité le 05/07/2024).

323.2. CULTURE MATÉRIELLE ET LA PERSISTANCE DE DONNÉES

Green coding, ou codage durable, est un terme émergent qui se concentre sur la création de logiciels plus écologiques et moins énergivores. Il implique de développer des applications ou d'écrire du code avec une empreinte carbone réduite et une meilleure efficacité énergétique¹⁷. Cette approche s'inscrit dans le contexte actuel d'une société de consommation où tout produit, y compris les logiciels, est susceptible de devenir une marchandise.

La réflexion sur la marchandise dans le cadre du *green coding* nous invite à considérer comment les logiciels, en tant que produits culturels, participent à la consommation de ressources naturelles et contribuent à l'empreinte écologique. Il s'agit de questionner la manière dont nous concevons et développons des logiciels, afin de minimiser leur impact environnemental. En adoptant une approche de *green coding*, nous pouvons contribuer à une économie plus durable et réduire notre dépendance aux ressources non renouvelables.

Il ne s'agit pas simplement de rejeter systématiquement toute initiative se revendiquant « verte » sous prétexte de *greenwashing*. L'objectif est plutôt d'inscrire la pratique du codage dans le contexte historique actuel : une société où tout produit humain est susceptible de devenir une marchandise.

Étant donné que ces objets sont catalogués à l'aide de codes ou d'applications, cette transformation génère de nouvelles données. Dans le cas du DaSCH, le principe est similaire : nous attribuons une nouvelle signification aux données en les traitant et en les normalisant selon un modèle prédéfini, même si ce modèle est élaboré en concertation avec les demandeurs. Cette nouvelle forme de donnée peut ensuite être réutilisée à d'autres fins, car le code sous-jacent est susceptible d'être adapté. Ainsi, les données sont non seulement transformées, mais elles acquièrent également une nouvelle signification, reflétant des relations sociales spécifiques. Par ailleurs, le code lui-même peut devenir un objet d'échange, voire une marchandise.

Fort de ces considérations sur le *green coding* et la nature des logiciels dans une société capitaliste, nous allons désormais nous pencher sur les choix de conception qui ont présidé à la création des codes.

17. *Green Coding : How to Reduce the Carbon Footprint of Your Code - IBM Blog*, URL : <https://www.ibm.com/blog/green-coding/> (visité le 27/06/2024)

Chapitre 4

Enjeux et défis de la numérisation patrimoniale

4.1 Les *pipelines*

L'objectif de ce stage, selon la fiche de mission, était de « développer un pipeline automatisé en Python pour créer des modèles 3D allégés à partir des modèles d'origine volumineux »¹. En d'autres termes, il s'agissait de concevoir du code Python pour résoudre les problèmes liés à la taille excessive des modèles 3D, qui rend leur chargement et leur visualisation difficiles sur les ordinateurs et les sites web. Les compétences requises pour ce stage étaient notamment des connaissances sur Python, la modélisation de données et la maîtrise du langage de requête SPARQL.

Afin de mieux comprendre comment les objectifs de ce stage ont été atteints et comment les compétences requises ont été mises en œuvre, cette section détaille le traitement des modèles 3D au sein du DaSCH et les solutions mises en place pour relever les défis rencontrés.

4.1.1 Utilisation de logiciels libres chez DaSCH

DaSCH utilise et encourage l'utilisation de logiciels libres et de codes réutilisables. Cependant, DaSCH ne disposait pas d'une solution spécifique

1. Voir Fiche de Stages, dans les Annexes.

pour répondre à certains besoins, d'où l'idée de créer des *pipelines* automatisés en Python pour gérer des tâches telles que la simplification des modèles 3D et la réduction de la résolution des images.

Chez DaSCH, il y avait deux types de logiciels utilisés. Premièrement, des logiciels pour réduire la taille (en Ko) des modèles 3D, et deuxièmement, des logiciels pour diminuer la résolution des images 2D. Parmi ces logiciels gratuits et *open source*, MeshLab, Blender et GIMP étaient fréquemment utilisés.

Premièrement, MeshLab est un système open source pour le traitement et l'édition de maillages triangulaire 3D. Il offre un ensemble d'outils pour éditer, nettoyer, réparer, inspecter, rendre, texturer et convertir des maillages. Il est particulièrement utile pour traiter les données brutes produites par des outils de numérisation 3D et préparer des modèles pour l'impression 3D . Dans notre cas, MeshLab est utilisé pour simplifier la quantité de sommets (*vertex*) d'un objet 3D au format **.obj**, ce qui réduit leur taille.

Deuxièmement, **Blender** est un logiciel capable de créer des visualisations 3D telles que des images fixes, des animations 3D et des séquences VFX. Il permet également l'édition vidéo. Il est particulièrement adapté aux utilisateurs et aux petits studios grâce à son *pipeline* unifié et son processus de développement réactif. Dans notre cas, Blender était utilisé pour gérer les modèles 3D, en ouvrant les fichiers réduits au format **.obj** et en exportant cela comme un fichier **.glb**.

Troisièmement, **GIMP**, acronyme de *GNU Image Manipulation Program*, est un programme distribué librement pour des tâches telles que la retouche photo, la composition d'images et la création d'images. Les conditions d'utilisation et les règles de copie sont clairement énoncées dans la *General Public License* (GNU). Chez DaSCH, GIMP est utilisé pour réduire la résolution des images 2D vers 1024 x 1024 pixels. Une fois cette tâche accomplie, des versions allégées des images 2D pouvaient être obtenues.

Toutefois, ces méthodes ne sont pas très fiables en large échelle. Tout d'abord, logiciels sont peu flexibles, parce que les logiciels sont des solutions techniques qui sont souvent conçus pour effectuer des tâches spécifiques, ce qui peut limiter leur capacité à s'adapter à des besoins particuliers ou à

évoluer avec les nouvelles technologies.

Deuxièmement, les interfaces graphiques, bien que conviviales pour les nouveaux utilisateurs, peuvent être contraignantes pour des tâches complexes ou répétitives et voire décourageant pour ceux qui ne sont pas habitués à utiliser des logiciels dans le quotidien.

En outre, les logiciels dépendent souvent de systèmes d'exploitation spécifiques ce qui peut rendre leur utilisation plus complexe et moins portable.

4.2 Les limites de l'automatisation des logiciels

Initialement, l'idée envisagée était d'automatiser les logiciels existants par le biais de **SCRIPTS PYTHON**. Pour ce faire, une recherche de bibliothèques permettant de simuler des actions utilisateur a été faite, dont le but est de simuler des clics de souris ou des saisies de clavier, similaire à des bibliothèques d'automatisation de interface de sites, tels **BEAUTIFUL SOUP** et **SELENIUM**. Une première tentative a été faite avec **PYAUTOGUI**, bibliothèque qui automatise ces mouvements, à partir de l'interface graphique des systèmes d'exploitations². Cette approche semblait prometteuse en raison de sa simplicité apparente.

Toutefois, cette solution s'est rapidement révélée inadéquate pour plusieurs raisons. Premièrement, elle souffre d'une forte dépendance vis-à-vis des systèmes d'exploitation, rendant les scripts fragiles face à des changements de configuration ou de version. Deuxièmement, cette approche est sensible aux modifications de l'interface graphique des logiciels, qui peuvent varier d'une version à l'autre, voire d'un système d'exploitation à l'autre.

Afin de surmonter ces limitations, nous avons opté pour une solution plus flexible : le développement de *pipelines* automatisés entièrement personnalisés. Plutôt que d'automatiser des systèmes d'exploitation ou des logiciels existants, nous avons choisi de créer des scripts Python sur mesure, conçus

2. Pour consulter la documentation officielle de PYAUTOGUI, Voir PyAutoGUI, *PyAutoGUI Documentation*, 2024, URL : <https://pyautogui.readthedocs.io/en/latest/> (visité le 29/08/2024).

pour réaliser les tâches de traitement d’images 2D et d’modèles 3D. Ces scripts nous permettent de contrôler chaque étape du processus, tels la réduction de taille des images 2D et des modèles 3D et également l’association de métadonnées à ces fichiers.

4.3 Utilisation de Python

Intéressons-nous maintenant à l’importance du choix de Python pour ce stage. Il est essentiel de souligner que, comme indiqué dans le cahier des charges, Python était le langage de programmation privilégié pour mener à bien cette mission. Cela ne signifie pas pour autant que j’étais limité à l’utilisation exclusive de Python, puisque une certaine latitude m’était laissée dans les choix techniques.

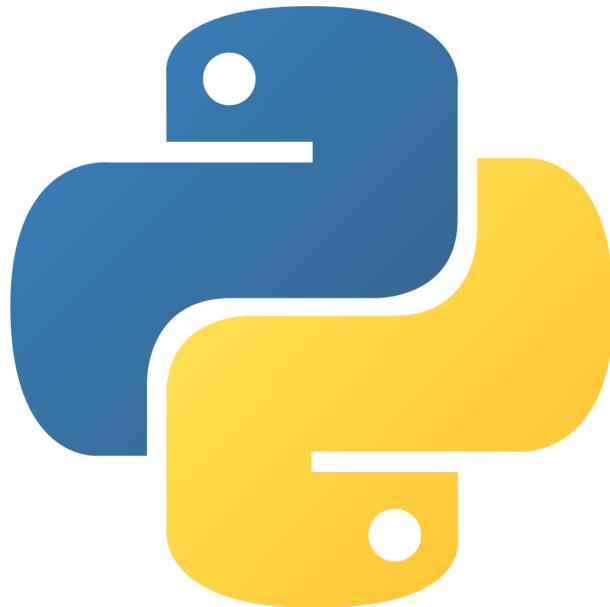


Figure 4.1 – Logo de PYTHON

En réalité, le choix de Python a été fait pour plusieurs raisons. Tout d’abord, Python offre un écosystème riche en bibliothèques stables et performantes, ce qui facilite grandement la résolution de problèmes variés. De plus, son caractère flexible et fluide entre différents systèmes d’exploitation

est un atout indéniable³. La popularité de Python assure une abondante documentation et une communauté active, soit sur des blogs soit sur des forums, comme stack overflow, ce qui est précieux pour trouver des réponses à d'éventuelles questions. Enfin, Python est reconnu pour sa simplicité et sa lisibilité, ce qui en fait un langage facile à apprendre et à maintenir.

4.4 Flask

Flask est un framework web Python léger et puissant, qui permet créer des applications web. Il se distingue par sa simplicité et son contrôle flexible sur ses composants, ce qui en fait un choix idéal pour les débutants comme pour les développeurs expérimentés. Ce projet utilise la version 3.x de Flask, dont la documentation se trouve sur le site officiel du framework⁴. Pour une compréhension de Flask, il est essentiel de connaître le concept de décorateur et comment il gère les requêtes *HTTP*⁵.



Figure 4.2 – Logo de Flask 3.0.x

Les décorateurs dans Flask

Un élément clé de Flask sont les décorateurs, des fonctions spéciales qui modifient le comportement d'autres fonctions. Dans le contexte de Flask,

3. Il est important de noter que pour certaines bibliothèques, il peut y avoir des petits changements qui vont dépendre des systèmes d'exploitation. C'est le cas de l'installation des environnements virtuels, mais aussi pour la bibliothèque PYTHON FLASK, où la description d'une route est différente entre Windows et MacOS.

4. Voir Flask Documentation, *Flask Documentation*, URL : <https://flask.palletsprojects.com/en/3.0.x/> (visité le 03/08/2024).

5. Pour une explication plus détaillée sur ces éléments de Flask, Voir Moraneus, « Python Flask : A Comprehensive Guide from Basic to Advanced », *Medium* (, 2023), URL : <https://medium.com/@moraneus/python-flask-a-comprehensive-guide-from-basic-to-advanced-fbc6ec9aa5f7>.

les décorateurs sont utilisés pour définir les *routes*, les méthodes *HTTP* et d'autres fonctionnalités importantes.

Gestion des requêtes HTTP

Flask fournit des méthodes pour gérer différents types de requêtes *HTTP*, tels que *GET*, *POST*, *PUT*, *DELETE*, etc. Cela permet de créer des applications web *RESTful* qui suivent les principes de l'architecture orientée ressources.

Pourquoi choisir Flask ?

Nous avons choisi Flask pour ce projet en raison de plusieurs facteurs :

- **Courbe d'apprentissage simple** : Flask est populaire pour le développement web et réputé pour sa facilité d'apprentissage, ce qui en fait un excellent choix pour le développement web avec Python.
- **Connaissance préalable** : Ayant étudié Flask en profondeur lors de notre deuxième année de master, il y avait une compréhension de ses fonctionnalités et de son fonctionnement.
- **Flexibilité et contrôle** : Flask offre un contrôle granulaire sur les différentes composantes de l'application, permettant une personnalisation avancée.
- **Communauté active** : Flask bénéficie d'une communauté dynamique et *supportive*, ce qui facilite la résolution de problèmes.

Pour compléter notre deux applications Flask avec une interface utilisateur soignée, nous avons choisi d'utiliser Bootstrap. Ce framework CSS nous permettra de créer des pages web esthétiques et réactives.

4.5 Bootstrap

Bootstrap est un framework CSS *open source* populaire, créé initialement par un designer et un développeur chez Twitter. Il a connu une croissance rapide et est devenu l'un des projets *open source* les plus utilisés dans le monde du développement web⁶.

6. L'analyse suivante est dérivée de la section About sur le site de Bootstrap. Voir Bootstrap, *About - Bootstrap v4.6*, 2023, URL : <https://getbootstrap.com/docs/4.6/about/overview/> (visité le 15/07/2024). Pour la documentation de Bootstrap, Voir Id.,



Bootstrap

Figure 4.3 – Logo de Bootstrap v4.x

À l'origine, Bootstrap a été développé chez Twitter en 2010 sous le nom de Twitter Blueprint. Après une semaine de hackathon interne, le projet a connu un succès fulgurant grâce à l'engouement des développeurs de tous niveaux. Pendant plus d'un an, Bootstrap a servi de guide de style pour le développement d'outils internes chez Twitter avant d'être publié au public.

Depuis sa première version en 2011, Bootstrap a connu plusieurs mises à jour majeures, notamment avec les versions 2 et 3. La version 2 a introduit la fonctionnalité de *responsive design*, tandis que la version 3 a été entièrement réécrite pour être responsive par défaut et adopter une approche *mobile-first*.

Bootstrap 4.0

Avec Bootstrap 4, le projet a été à nouveau réécrit pour intégrer deux changements architecturaux clés : la migration vers Sass et l'adoption de Flexbox pour la mise en page. L'objectif était de contribuer à l'évolution du développement web en promouvant de nouvelles propriétés CSS, en réduisant

Introduction - Bootstrap v4.6, URL : <https://getbootstrap.com/docs/4.6/getting-started/introduction/> (visité le 15/07/2024).

les dépendances et en adoptant de nouvelles technologies compatibles avec les navigateurs modernes.

Pour notre projet, le choix de Bootstrap 4 s'est fait pour convenir à cette envie de créer une application responsive, mais aussi pour d'autres raisons. Tout d'abord, la documentation de Bootstrap 4 est réputée pour sa clarté et sa stabilité, ce qui facilite l'apprentissage et l'utilisation du framework. De plus, Bootstrap 4 offre une intégration facile grâce à des instructions de copier-coller pour les différents composants, ce qui accélère le processus de développement. Enfin, l'intégration de Bootstrap 4 dans les templates Flask s'est révélée simple et efficace, permettant de créer rapidement des interfaces utilisateur modernes.

Bien que Bootstrap 4 offre un socle solide et rapide pour développer des interfaces utilisateur, son utilisation extensive peut parfois conduire à une certaine homogénéité visuelle entre les *sites web*. En effet, l'utilisation de templates pré-conçus facilite le développement rapide, mais peut limiter la créativité et rendre les sites moins distinctifs. De plus, une compréhension basique des fonctionnalités de Bootstrap est nécessaire pour pouvoir personnaliser efficacement les composants et les adapter à des besoins spécifiques. Néanmoins, ces contraintes sont principalement liées à l'apprentissage du framework lui-même et ne constituent pas un obstacle majeur à la maintenance et à la pérennité d'une application *web*.

4.5.1 Flask-Bootstrap

Pour le développement de cette application, nous avons opté pour le framework Flask couplé à l'extension Flask-Bootstrap⁷. La version 3.3.7.1 a été

7. Flask-Bootstrap et Bootstrap-Flask sont deux extensions Python avec le même but de simplifier l'intégration du framework CSS Bootstrap dans les applications web Flask. Bien qu'elles partagent un objectif commun, elles fonctionnent différemment : Flask-Bootstrap intègre les composants de Bootstrap directement dans les *templates* de Flask, alors que Bootstrap-Flask est plutôt une abstraction des macros de Jinja et intègre ces composants indirectement. Pour la documentation de Flask-Bootstrap 3.x, Voir Flask-Bootstrap, *Welcome to Flask - Flask Documentation*, URL : <https://pythonhosted.org/Flask-Bootstrap> (visité le 03/08/2024). Pour la documentation de Bootstrap-Flask 2.2.x, Voir Bootstrap-Flask, *Bootstrap-Flask - Bootstrap-Flask 2.2.x documentation*, URL : <https://bootstrap-flask.readthedocs.io/en/stable/index.html> (visité le

utilisée pour ce projet.

Ce choix s'est avéré judicieux pour plusieurs raisons. Tout d'abord, Flask-Bootstrap permet une intégration transparente de Bootstrap dans les templates Flask, offrant ainsi un accès direct et intuitif à une vaste gamme de composants visuels. Cette facilité d'utilisation a considérablement réduit le temps de développement et nous a permis de nous concentrer sur la logique métier plutôt que sur la mise en forme. De plus, étant donné que Flask-Bootstrap est une bibliothèque couramment utilisée dans le cadre de nos cours, nous maîtrisions déjà ses fonctionnalités, ce qui a facilité notre prise en main du projet.

4.6 Docker

Docker est une plateforme ouverte conçue pour le développement, l'expédition et l'exécution d'applications⁸. Grâce à Docker, il est possible de gérer l'infrastructure de la même manière que l'on gère les applications, ce qui réduit considérablement le délai entre l'écriture du code et son déploiement en production. Cette plateforme est donc particulièrement utile pour automatiser et simplifier les processus de test, de déploiement et de mise en production du code.

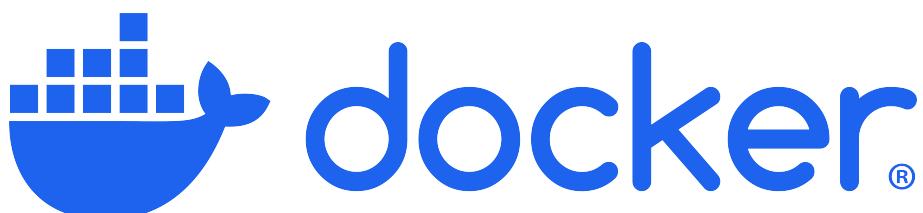


Figure 4.4 – Logo de Docker

03/08/2024).

8. Cette définition, tout comme l'analyse du chapitre, vient de la documentation de Docker. Voir Docker, *What is Docker ? - Docker Docs*, URL : <https://docs.docker.com/get-started/docker-overview/> (visité le 03/08/2024).

Objet Docker : l'image

Dans l'écosystème Docker, l'image représente un modèle en lecture seule qui contient les instructions nécessaires pour créer un conteneur Docker. Souvent, une image est construite à partir d'une autre image, avec des personnalisations supplémentaires spécifiques à l'application. Par exemple, une image pourrait être basée sur l'image Ubuntu, mais inclure des installations supplémentaires comme un serveur web Apache et l'application elle-même. La légèreté et la rapidité des images Docker, comparées à d'autres technologies de virtualisation, sont dues au fait que chaque modification apportée à une image ne reconstruit que les couches affectées, rendant ainsi le processus de création et d'exécution plus efficace. De plus, cela permet aux applications d'être exécutées dans des environnements différents, puisqu'il permet de créer des images adaptées à chaque distribution Linux. Par exemple, nous pouvons avoir une image pour Ubuntu, une autre pour Debian, et ainsi de suite⁹.

Objet Docker : le conteneur

Un conteneur (*container*) est une instance exécutable d'une image Docker. C'est un environnement isolé dans lequel une application s'exécute avec toutes ses dépendances. Un conteneur peut être créé, démarré, arrêté, déplacé ou supprimé à l'aide de l'API ou de la CLI Docker. Par défaut, les conteneurs sont relativement bien isolés les uns des autres et du système hôte, mais Docker permet de configurer cette isolation en fonction des besoins. Les conteneurs sont essentiels pour la portabilité des applications, permettant de les exécuter de manière cohérente sur différents environnements, que ce soit sur un poste de développeur ou dans un centre de données.

L'architecture Docker

L'architecture Docker est basée sur un modèle client-serveur. Le client Docker communique avec le *daemon* Docker, qui est responsable de la construc-

9. Bien que Docker puisse faciliter le déploiement d'applications sur différents systèmes d'exploitation, il est moins courant de créer des images Docker pour des systèmes d'exploitation comme Windows ou macOS. Les raisons principales sont liées aux différences fondamentales au niveau du noyau et des outils système.

tion, de l'exécution et de la distribution des conteneurs. Cette communication se fait via une API REST, utilisant des sockets UNIX ou une interface réseau. Docker Compose, un autre client Docker, permet de gérer des applications composées de plusieurs conteneurs. En outre, le *daemon* Docker (*dockerd*) gère non seulement les conteneurs, mais aussi d'autres objets Docker tels que les images, les réseaux et les volumes, facilitant ainsi la gestion et l'orchestration des services.

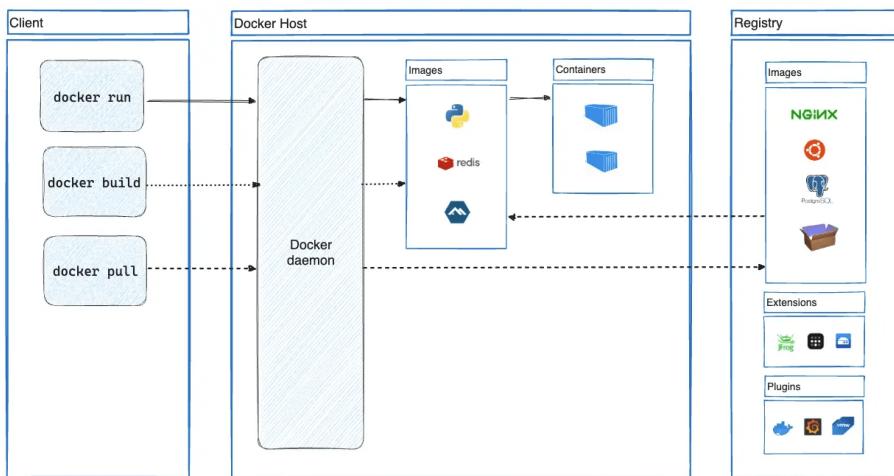


Figure 4.5 – Schéma de l'architecture Docker

L'adoption de Docker au sein de DASCH est donc stratégique. L'organisation a un Docker Hub dédié et cela permettra d'accélérer la livraison des applications et de les diffuser¹⁰.

10. Pour plus de détails sur les applications de DaSCH sur Docker Hub, Voir DaSCH, *DaSCH Swiss on Docker Hub*, URL : <https://hub.docker.com/u/daschswiss> (visité le 03/08/2024).

Deuxième partie

**De l'image au standard IIIF :
conceptions des *pipelines***

Chapitre 5

Optimisation des images et objets pour IIIF : outils et méthodes

Le standard IIIF a ouvert de nouvelles voies pour la gestion et la visualisation de contenus numériques, notamment les images 2D et les modèles 3D. Cependant, l'intégration harmonieuse de ces différents types de ressources au sein de ce cadre reste un défi. Si des applications spécifiques ont été développées pour répondre à cette problématique¹, leurs limites en termes de flexibilité et de maintenance ont rapidement été mises en évidence. C'est dans ce contexte que des chercheurs se sont tournés vers des solutions plus automatisées, telles que les scripts Python².

Dans cette étude, nous nous proposons d'explorer plus en détail les pos-

1. Par exemple, l'emploi de GIMP, Blender et MeshLab pour cette tâche est déjà connu dans le milieu patrimoniale, Voir Julien A. Raemy et Rita Gautschi, « Élaboration d'un processus pour les images 3D reposant sur IIIF », dans *Colloque Humanistica 2023*, Université de Genève, Suisse, 2023.

2. Dans son article de 2023, Rossenova constate également ce problème, et cite l'exemple du *The Smithsonian Cook* comme tentative pour contourner ce défi. Voir L. Rossenova, *IIIF for 3D – making web interoperability multi-dimensional...* Bien que l'étude de Ioannides et Patias ne se concentre pas spécifiquement sur cette question, il est intéressant de noter que, bien que récente, elle évoque les difficultés liées à la manipulation de modèles 3D sans aborder la possibilité d'utiliser des scripts Python. Cet ouvrage constitue néanmoins une réflexion précieuse sur les défis rencontrés dans ce domaine. Voir *3D Research : Challenges in Cultural Heritage III : Complexity and Quality in Digitisation...*

sibilités offertes par les *pipelines* automatisés en Python pour traiter les défis liés à l'intégration de contenus 2D et 3D au sein du standard IIIF. En s'appuyant sur les travaux existants, nous présenterons ces *pipelines*, capables de s'adapter à une grande variété de données.

5.1 Conception du DaSCH Converter

DaSCH Converter est une application conçue pour convertir des images aux formats .jpg, .jpeg et .png en versions plus légères. Il visait remplacer le logiciel GIMP.

5.1.1 Bibliothèques

Ce code Python utilise plusieurs bibliothèques pour effectuer la conversion et le téléchargement d'images redimensionnées. Voici un aperçu de chacune d'entre elles :

Flask

```
from ..app import app, ALLOWED_EXTENSIONS
```

Importe l'application FLASK app et la variable **ALLOWED_EXTENSIONS** depuis un module parent nommé app. Cette variable contient une liste d'extensions de fichiers d'images autorisés pour le traitement qui sont les suivants : .jpg, .jpeg et .png.

io

```
from io import BytesIO
```

Importe la classe BytesIO du module io. Cette classe permet de créer un objet ressemblant à un fichier en mémoire, ce qui est utile pour conserver temporairement l'image redimensionnée avant son téléchargement.

flask

```
from flask import render_template, redirect, request,  
url_for, send_file:
```

Importe plusieurs fonctions utiles de Flask pour la gestion des requêtes HTTP. Par exemple, **render_template** permet de rendre des templates HTML pour l'affichage des pages *web*; **redirect** redirige l'utilisateur vers une autre URL; **request** : Donne accès aux informations de la requête en cours, y compris les fichiers téléchargés; **url_for** : Génère l'URL d'un endpoint Flask en fonction de son nom ; finalement, **send_file** : Permet de renvoyer un fichier à l'utilisateur en téléchargement.

werkzeug.utils

```
from werkzeug.utils import secure_filename
```

Importe la fonction **secure_filename** du module werkzeug.utils. Cette fonction sécurise le nom de fichier téléchargé afin d'éviter des problèmes potentiels de sécurité.

Pillow

```
from PIL import Image
```

Importe la classe **Image** du module **PIL** (aussi connu sous le nom de **Pillow**). Cette bibliothèque est utilisée pour la manipulation d'images en **PYTHON**.

5.1.2 Fonctionnalités du code

Le code **PYTHON** implémente une application *web* simple qui permet aux utilisateurs de télécharger une image et de la redimensionner avant de la télécharger à nouveau. Voici une explication détaillée des différentes parties du code :

Configuration globale (en dehors des fonctions)

```
Image.MAX_IMAGE_PIXELS = None
```

Supprime toute limite imposée par Pillow sur la taille maximale des images en *pixels*.

Fonction allowed_file (filename)

Vérifie si l'extension du fichier téléchargé correspond à l'une des extensions autorisées conservées dans la variable **ALLOWED_EXTENSIONS**.

Fonction resize_image()

Ouvre l'image en utilisant le nom de fichier **filename** fourni. Calcule le rapport de réduction nécessaire pour que l'image ne dépasse pas les dimensions maximales définies par **max_width** et **max_height** (ici 1024x1024 *pixels*).

Redimensionne l'image en utilisant le filtre anti-aliasing LANCZOS pour une meilleure qualité. Renvoie l'image redimensionnée.

Route @app.route(“/”)

Correspond à la racine de l'application. Renvoie le template HTML principal conservé dans le dossier pages/main.html. Ce template est responsable de l'affichage du formulaire de téléchargement de l'image.

Route @app.route(“/convert”, methods=[‘POST’])

Correspond à l'URL /convert et ne gère que les requêtes HTTP de type POST (souvent utilisées pour les formulaires de téléchargement). Exécute la fonction **upload_image** pour traiter le fichier téléchargé.

Fonction upload_image()

Vérifie si le formulaire de téléchargement contient un fichier nommé "file". Si aucun fichier n'est présent, redirige l'utilisateur vers la même page. Ré-

cupère le fichier téléchargé depuis la requête et vérifie si son nom n'est pas vide. Si le fichier est valide et son extension est autorisée (vérification par `allowed_file`), sécurise le nom de fichier et récupère son extension. Redimensionne l'image en utilisant la fonction `resize_image`. Crée un objet `BytesIO` pour conserver l'image redimensionnée en mémoire. Enregistre l'image redimensionnée dans l'objet.

5.2 Conception du DaSCH Mesh

DaSCH Mesh est une application conçue pour convertir des images 3D en versions plus légères à l'aide d'une méthode de simplification par décimation quadratique des arêtes, similaire à celle disponible dans des logiciels comme MeshLab.

5.2.1 Bibliothèques

Dans le cadre de cette étude, l'objectif était de développer une méthode permettant de réduire la taille des modèles 3D. Pour ce faire, le prérequis était de trouver une bibliothèque qui utilise la méthode de contraction *Method Iterative Contraction Edges with Quadric Error Metrics*, telle qu'implémentée dans l'outil MeshLab. Cette technique, détaillée dans l'article de Garland et Heckbert, consiste à simplifier progressivement un maillage en contractant les arêtes les moins significatives³.

3. Cet algorithme est le développement d'un algorithme précédent, créé aussi pour les auteurs en 1997, dans lequel ils se sont intéressés seulement par la réduction de surfaces. Il est donc intéressant de souligner que cet article les auteurs ont envisagé bien les problèmes des modèles 3D : ils souvent présentent d'autres propriétés, telles les couleurs, textures et surface normales, qui ont été finalement traitées en 1998. De plus, en effet l'usage de mémoire est réduite comme ils ont prévu. Pour l'article de 1997, Voir Michael Garland et Paul S. Heckbert, « Surface Simplification Using Quadric Error Metrics », dans *SIGGRAPH '97 Proceedings*, 1997, URL : <https://www.cs.cmu.edu/~garland/Papers/quadratics.pdf>. Pour l'article de 1998 sur le *Method Iterative Contraction Edges with Quadric Error Metrics* qui sert de base à cette fonction utilisée sur MeshLab, PyMeshLab et Open3D, Voir Id., « Simplifying Surfaces with Color and Texture using Quadric Error Metrics », dans *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, ACM Press/Addison-Wesley Publishing Co., 1998, URL : <https://www.cs.cmu.edu/~garland/Papers/quadratic2.pdf>.

Afin de mettre en œuvre cette méthode, deux bibliothèques Python ont été étudiées : PyMeshLab et Open3D. Ces deux bibliothèques offrent des fonctionnalités de traitement de maillages 3D et font référence au méthode de contraction des arêtes pour les surfaces avec des couleurs et textures de 1998.

Après une phase de tests comparatifs qui a eu lieu pendant trois semaines, la bibliothèque Open3D a été sélectionnée. En effet, PyMeshLab, bien que proposant un large éventail de fonctionnalités, s'est révélée peu performante sur des modèles de grande taille, entraînant des temps de calcul excessivement longs, et voire des boucles infinies.

Open3D s'est avéré être une solution efficace pour réduire la complexité des modèles 3D. Les performances de cette bibliothèque seront analysées en détail dans la troisième partie de ce travail. Cependant, il convient de noter que Open3D ne permet pas de reproduire intégralement les paramètres de l'application MeshLab. Cette limitation restreint quelque peu la flexibilité de l'outil, bien que le cœur de l'algorithme de contraction des arêtes étant identique dans les deux bibliothèques, des résultats satisfaisants ont pu être obtenus. Finalement, voici une explication détaillée de chaque bibliothèque du code **main.py** :

Flask

```
from ..app import app, ALLOWED_EXTENSIONS
```

Importez l'application FLASK `app` et la variable `ALLOWED_EXTENSIONS` depuis un module parent nommé `app`. Cette variable contient une liste d'extensions de fichiers d'images, de modèles 3D, de textures et de couleurs autorisés pour le traitement, qui sont les suivants : `.glb`, `.gltf`, `.jpg`, `.mtl`, `.obj`, `.ply`, `.png`, `.stl`, `.tiff`.

os

Non explicitement importée, mais utilisée dans la fonction `tempfile.NamedTemporaryFile`. Fournit des fonctions pour interagir avec le système d'exploitation, telles que la création de fichiers temporaires.

io

```
import io
```

Importe le module `io` qui fournit des classes pour la manipulation de flux de données.

open3d

```
import open3d as o3d
```

Importe la bibliothèque `open3d` sous le alias `o3d`. Cette bibliothèque est essentielle pour le traitement de nuages de points et de maillages 3D. Elle permet de charger, manipuler, visualiser et exporter des données 3D. Dans cette étude, elle a été utilisée en combinaison avec la méthode de réduction de maillages *Method Iterative Contraction Edges with Quadric Error Metrics*.

numpy

```
import numpy as np
```

Importe la bibliothèque `NUMPY` sous le alias `np`. **NumPy** est une bibliothèque fondamentale pour le calcul scientifique en `PYTHON`. Elle fournit des structures de données multidimensionnelles (tableaux) et des fonctions efficaces pour les opérations matricielles et algébriques linéaires.

tempfile

```
import tempfile
```

Importe le module `tempfile` qui fournit des fonctions pour créer des fichiers temporaires.

flask

Importé à partir de `..app`, comme vu précédemment. Fournit des fonctionnalités pour la création d'applications *web* en `PYTHON`.

werkzeug.utils

Importé à partir de ..app, comme vu précédemment. Fournit des utilitaires divers, dont la fonction `secure_filename` utilisée pour sécuriser les noms de fichiers téléchargés.

5.2.2 Fonctionnalités du code

Ce code PYTHON implémente une application *web* qui permet aux utilisateurs de monter une image sur le site, la convertir en un maillage 3D simplifié et la renvoie en téléchargement. Voici une explication détaillée des différentes parties du code :

Fonction `allowed_file(filename)`

Comme pour la fonction précédente, vérifie si l'extension du fichier mis en ligne correspond à l'une des extensions autorisées conservées dans la variable `ALLOWED_EXTENSIONS`.

Fonction `simplify_image(filename)`

Création de fichier temporaire :

Utilise `tempfile.NamedTemporaryFile` pour créer un fichier temporaire avec l'extension `.obj`. Écrit le contenu du fichier téléchargé (représenté par `filename`) dans le fichier temporaire. Stocke le nom du fichier temporaire dans la variable `temp_filename`.

Lecture du maillage :

Utilise la fonction `o3d.io.read_triangle_mesh` pour lire le fichier temporaire (considéré comme un maillage 3D au format `.obj`). Stocke le maillage lu dans la variable `mesh`.

Centrage du maillage :

Calcule la boîte englobante (bounding box) du maillage à l'aide de `mesh.get_axis_aligned_bounding_box`. Récupère les coordonnées minimum et maxi-

mum de la boîte englobante. Calcule le centre de la boîte englobante. Calcule le vecteur de translation nécessaire pour déplacer le maillage vers son centre. Déplace le maillage en utilisant la fonction **mesh.translate**.

Simplification des maillages :

Utilise la fonction **mesh.simplify_quadric_decimation(number_of_triangles)** pour simplifier le maillage. Le paramètre **target_number_of_triangles** est fixé à 65000 dans ce code. Cette fonction réduit le nombre de triangles du maillage en s'appuyant sur une méthode de décimation quadratique.

Calcul et rendu des normales :

Calcule les normales des sommets et des triangles du maillage simplifié à l'aide des fonctions **mesh_simplified.compute_vertex_normals** et **mesh_simplified.compute_triangle_normals**. Les normales permettent d'obtenir un éclairage réaliste lors du rendu du maillage.

Sauvegarde du maillage simplifié :

Similaire à la création du fichier temporaire, il utilise **tempfile.NamedTemporaryFile** pour créer un nouveau fichier temporaire avec l'extension **.obj**. Enregistre le maillage simplifié dans le fichier temporaire à l'aide de la fonction **o3d.io.write**

5.3 Structure de DaSCH Converter et DaSCH Mesh

Concernant la structure du code, les deux applications utilisent la même structure de dossiers avec de légères modifications adaptées à leur cas d'utilisation. Le modèle de dossiers de ces deux applications a été inspiré par celui proposé sur GitHub par Maxime Challon, l'un de mes professeurs à l'École

nationale des chartes⁴. En ce qui concerne le code de DaSCH Converter, notre travail s'est bénéficié des idées de l'application PNG-to-JPEG⁵.

De plus, les deux applications sont publiées de manière privée sur mon propre GitHub et seront finalement rendues publiques une fois la soutenance de mémoire effectuée. Une autre étape consistera à mettre à disposition la version finale de ces applications sur le *Docker Hub* de DaSCH, qui, comme mentionné précédemment dans le Chapitre 4, sert à archiver et à mettre à disposition les applications développées par l'organisation.

Concernant le DaSCH Converter, il est possible d'observer sa structure générale à partir de cette arborescence sur GitHub :

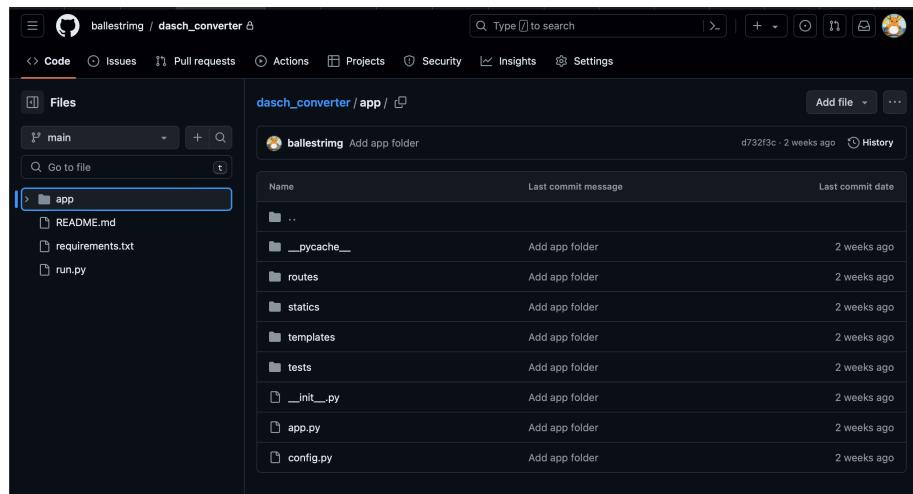


Figure 5.1 – Structure de l'application DaSCH Converter sur GitHub

Quant au DaSCH Mesh, sa structure est la suivante :

L'arborescence des applications a pour but d'organiser les différents fichiers en dossiers, chacun correspondant à une fonctionnalité spécifique. Le dossier principal contient des sous-dossiers pour le code source, les tests, les données, la documentation et les fichiers de configuration. Cette structure permet une bonne séparation des préoccupations et facilite la maintenance

4. Voir Maxime Challon, *CoursM2TNAH_Flask_code*, 2023, URL : https://github.com/MaximeChallon/CoursM2TNAH_Flask_code (visité le 15/07/2024).

5. Voir HerreraCarlos81, *PNG-to-JPEG : A Python program to convert all png images within a specific folder to jpeg files on a target folder*, 2023, URL : <https://github.com/HerreraCarlos81/PNG-to-JPEG> (visité le 12/08/2024).

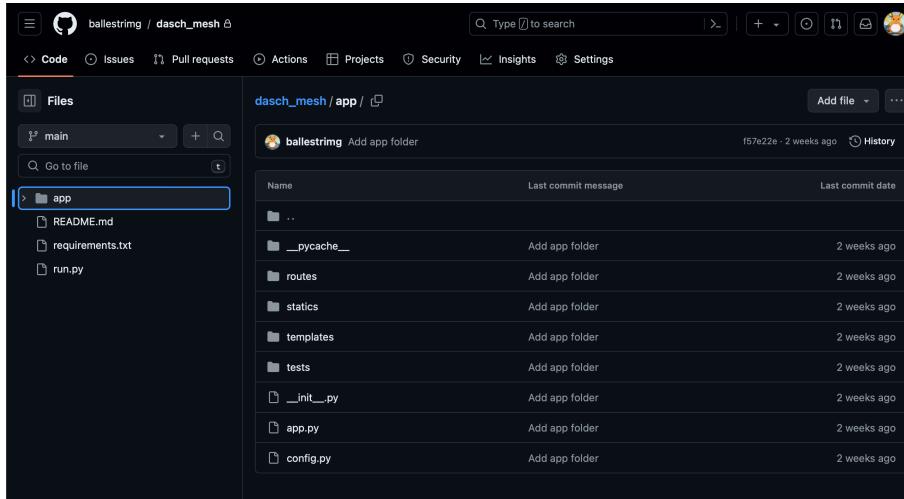


Figure 5.2 – Structure de l’application DaSCH Mesh sur GitHub

du projet.

Comme d’habitude pour les applications sur GitHub, on trouve un fichier **README.md** qui fournit une description du projet et un fichier **requirements.txt** qui liste les dépendances nécessaires pour exécuter l’application (bibliothèques, version Python, etc.).

Quant au *design* de la page, il est inspiré du site **Smallpdf**. L’idée des boîtes pour y glisser des fichiers, ainsi qu’un bouton pour ajouter des contenus ont été utilisés en tant que principes des applications.



Figure 5.3 – Page de conversion de fichier sur Smallpdf

Concernant les tests exécutés dans différentes versions PYTHON, sur *macOS M1 (Apple Silicon)* les versions PYTHON pour DaSCH Converter et DaSCH Mesh ont été, respectivement, **3.11.0** et **3.9.0**; sur *Windows*, les deux applications ont utilisé la version **3.9.0** de PYTHON; les tests sur *Ubuntu 22.04.3* ont utilisé PYTHON **3.10.12**. Il est important d'expliciter cela, car certaines bibliothèques comme BLINKER 1.7.0 et COUNTOURPY 1.2.1 n'acceptent pas des versions PYTHON inférieures, respectivement, à **3.7.x** et **3.9.x**. De plus, jusqu'à ce jour, OPEN3D fonctionne seulement dans les versions PYTHON inférieures à **3.11.x**. A partir de Python 3.8, il est possible d'intégrer le framework Flask.

Chapitre 6

Expérience utilisateur

6.1 Intégration de l'UX dans les applications

Le développement de ces applications, DaSCH Converter et DaSCH Mesh, a mis en lumière l'importance de l'expérience utilisateur (*UX* ou *user experience*). Ce qui était autrefois perçu comme un domaine réservé aux designers s'est imposé comme un élément incontournable de la programmation, complétant la logique et le code de l'application. Dans ce chapitre, les raisons pour lesquelles l'intégration de l'*UX* a été au cœur de nos préoccupations et comment il a été mis en œuvre seront discutés en détail.

Pour mieux comprendre cette démarche, l'analyse abordera des différents aspects de l'application.

6.1.1 Vers une *UX* inclusive

La conception d'applications numériques est souvent marquée par une dichotomie entre les utilisateurs experts, tels que les programmeurs et les administrateurs de plateformes, et les utilisateurs grand public, regroupant chercheurs, étudiants, amateurs et professionnels. Cette distinction, bien que pratique pour certains aspects de la conception, peut entraver la création d'une expérience utilisateur (*UX*) véritablement inclusive. Or, en privilégiant un langage visuel efficace et en proposant une interface intuitive, notre application a cherché à combler ce fossé. En effet, en misant sur des éléments

graphiques clairs et des interactions simples, cet outil devient accessible à un large public, sans pour autant sacrifier les fonctionnalités attendues par les utilisateurs les plus expérimentés. Cette approche a permis de créer une expérience utilisateur unifiée, où les spécialistes peuvent tirer parti d'options avancées tout en offrant aux néophytes une prise en main rapide et efficace. Ainsi, en déconstruisant la traditionnelle opposition entre experts et grand public, notre application s'inscrit dans une démarche de démocratisation de l'accès aux outils numériques.

6.1.2 Fonctionnalités faciles à apprendre

Les deux applications ont été conçues pour répondre à un besoin spécifique : réduction de la taille des fichiers 2D et 3D. Pour atteindre cet objectif, les fonctionnalités simples et efficaces ont été mises en place, telles que le *drag and drop* et les clics pour ajouter du contenu. Ces choix ont été guidés par le désir de minimiser la courbe d'apprentissage et de rendre l'application accessible à tous, y compris aux utilisateurs novices.

6.1.3 La palette de couleurs

La palette de couleurs a également été soigneusement sélectionnée. L'identité visuelle du DaSCH a été maintenue tout en privilégiant des teintes qui ont la même couleur de l'association¹. Les applications DaSCH Converter et DaSCH Mesh avaient, respectivement, les *hex* couleurs #336790 et #74A2CF². Le choix de la couleur bleue avec peu de luminosité a deux raisons principales : en premier lieu, cette couleur est souvent associée dans notre culture à des entreprises reconnues telles qu'*IBM* et *Facebook*; en deuxième

1. De même pour la police de caractères LATO SANS SERIF, puisqu'elle est la police de DaSCH.

2. Ces codes *hex*, ainsi que les logos comme mentionné dans le Chapitre 2, ont été trouvés sur le document d'identité visuelle présent sur le compte Figma de DaSCH. Il faut préciser que les codes de couleurs hexadécimaux, plus communément appelés codes *Hex*, sont normalement composés d'un symbole dièse (#) suivi de trois paires de chiffres représentant chacune les couleurs primaires RVB (rouge, vert, bleu). Adobe, *Comprendre les codes de couleurs hexadécimaux*, URL : <https://www.adobe.com/fr/creativecloud/design/discover/color-hex-codes.html> (visité le 21/07/2024).

lieu, sa basse luminosité et saturation suscitent plus d'émotions³.



Figure 6.1 – Page principale du DaSCH Converter



Figure 6.2 – Page principale du DaSCH Mesh

3. Une étude menée avec des personnes de différentes cultures et niveaux de scolarité, fait pendant plusieurs années, a montré que la signification d'une couleur est plutôt influencé par sa luminosité et sa saturation et peu par sa teinte. C'est-à-dire qu'il n'y a pas une émotion liée de forme indissociable à une couleur à cause de sa teinte. Cela réfute l'idée reçue dans le design et la mode selon laquelle il y aurait une émotion universelle qui pourrait être associée à une couleur. En outre, pour les couleurs chaudes (*warm-cool*) bien que la saturation et la teinte aient leur degré d'importance, les personnes n'avaient pas une opinion consistante sur quel aspect était le plus important, Voir Xiao-Ping Gao, John Xin, Tetsuya Sato, Aran Hansuebsai, Marcello Scalzo, Kanji Kajiwara, Shing-Sheng Guan, Josep Valldeperas-Morell, Manuel Lis et Monica Billger, « Analysis of cross-cultural color emotion », *Color Research & Application*, 32 (juin 2007), 2024-08-29, p. 223-229, DOI : 10.1002/col.20321. Il est aussi intéressant de mentionner qu'une autre étude, plus ancienne, a déjà remarqué le fait que la perception visuelle est tellement complexe que d'autres aspects de nature distincte de la vision pourraient aussi jouer, tels que le goût, l'odeur et l'ambiance. Cela montre, encore une fois, l'importance de l'interdisciplinarité dans les champs d'études, Voir Arne Valberg, *Light Vision Color*, John Wiley & Sons Ltd, Chichester, 2005.

6.1.4 Fluidité de l'*UX*

Enfin, une attention particulière à la fluidité de l'expérience utilisateur a été accordée. Le téléchargement automatique des fichiers une fois la conversion terminée permet de gagner du temps et d'éviter toute manipulation complexe. De plus, l'ensemble des fonctionnalités de l'application a été conçu pour être intuitif et auto-explicatif, ne nécessitant aucune connaissance préalable en informatique de la part de l'utilisateur.

6.1.5 UX et *green coding*

Un UX design minimaliste, caractérisé par une interface épurée et des fonctionnalités essentielles, s'inscrit naturellement dans une démarche éco-responsable. En effet, en réduisant l'encombrement visuel et en simplifiant les interactions, ce type de design contribue à diminuer la charge cognitive des utilisateurs, et aussi pour une prise de décision plus rapide des utilisateurs. Cela se traduit par une consommation moindre de ressources informatiques, telles que la mémoire et la puissance de calcul. De plus, un design minimaliste limite le nombre d'éléments à charger, ce qui optimise la vitesse de chargement des pages et réduit ainsi l'empreinte carbone liée à la consommation énergétique des serveurs.

Enfin, l'intégration de l'*UX* dans notre application s'est avérée être un choix stratégique essentiel. En privilégiant la simplicité de fonctionnalités, une courbe d'apprentissage courte et une esthétique soignée, ces applications répondent aux besoins du DaSCH. Cet *UX* minimaliste en allégeant les applications et en facilitant leur maintenance, permet des codes plus durables, plus accessibles et plus respectueux de l'environnement, tout en offrant une expérience utilisateur optimale.

Chapitre 7

Enrichissement métadonnée : vers les *Scenes* IIIF

7.1 Conception du DaSCH IIIF

DaSCH IIIF est un code PYTHON encore en développement dont le but est enrichir des fichiers 3D avec des métadonnées conformes aux standard IIIF.

7.1.1 Modèles utilisés

L'*IIIF Workbench* a été un modèle privilégié afin de regarder la structure base des dossiers contenant des images pour les *Image API* et *Presentation API* d'IIIF, ainsi que pour concevoir un fichier JSON¹.

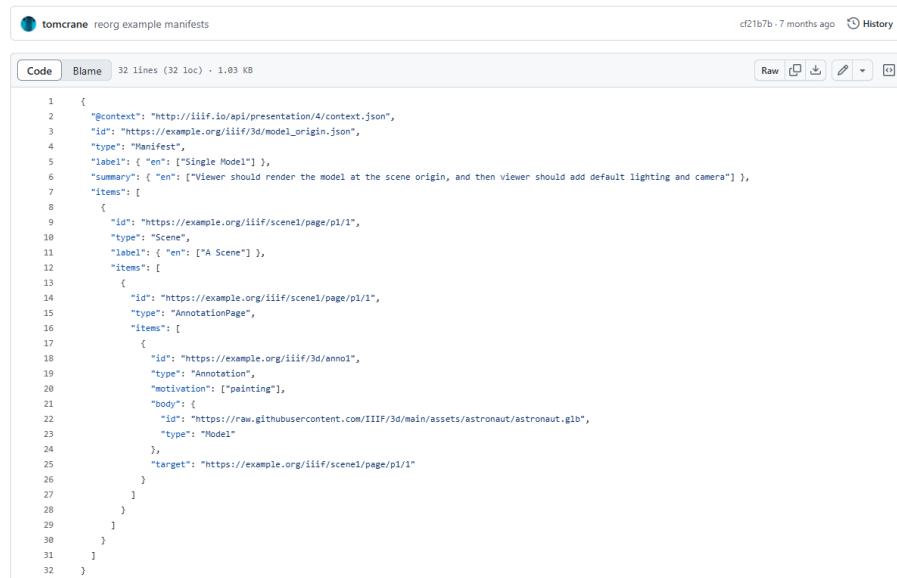
Ce site nous permet d'envoyer des images qui sont archivées dans nos ordinateurs au site par le biais d'une compte GitHub, auxquels sont associés au *Image API* 2.x ou 3.x. Par la suite, ces images vont avoir un fichier JSON associé, possédant un **id** qui permettra, entre autres choses, créer l'*IIIF Manifest*. Dans ce sens, l'**id** est une information très importante lors de la conceptualisation, et c'est en même temps la caractéristique la plus basique

1. Nous avons utilisé ce site lors du cours « Anglais langue de l'informatique » dispensé par Edward Gray dans le cadre du master Technologies numériques appliquées à l'histoire à l'École nationale des chartes. Voir référence *GDMR Digital Workbench Login*, URL : <https://workbench.gdmrdigital.com/login.xhtml> (visité le 21/07/2024).

qui autorise à un fichier JSON devenir un *IIIF Manifest*. Muni de ces deux informations, la modélisation des données est plus profiteuse, une fois que cela montre une application qui enrichie des fichiers avec de métadonnées conformes aux standards d'IIIF.

7.1.2 Prérequis

Pour conceptualiser le fichier JSON avec les paramètres minimaux, le modèle utilisé était extrait de cet exemple d'IIIF `model_origin.json`² :



```

1  {
2    "@context": "http://iiif.io/api/presentation/4/context.json",
3    "id": "https://example.org/iiif/3d/model_origin.json",
4    "type": "Manifest",
5    "label": { "en": "Single Model" },
6    "summary": { "en": ["Viewer should render the model at the scene origin, and then viewer should add default lighting and camera"] },
7    "items": [
8      {
9        "id": "https://example.org/iiif/scene1/page/p1/1",
10       "type": "Scene",
11       "label": { "en": ["A Scene"] },
12       "items": [
13         {
14           "id": "https://example.org/iiif/scene1/page/p1/1",
15           "type": "AnnotationPage",
16           "items": [
17             {
18               "id": "https://example.org/iiif/3d/anno1",
19               "type": "Annotation",
20               "motivation": ["painting"],
21               "body": [
22                 {
23                   "id": "https://raw.githubusercontent.com/IIIF/3d/main/assets/astronaut/astronaut.gltf",
24                   "type": "Model"
25                 },
26                 "target": "https://example.org/iiif/scene1/page/p1/1"
27               ]
28             }
29           ]
30         }
31       ]
32     }
33   }

```

Figure 7.1 – Code du fichier `model_origin.json`

Il s'agit d'un fichier basique dans lequel force est de constater la présence d'un `id` qui permet d'utiliser cet objet et le conformer au standard IIIF.

7.1.3 Bibliothèques

Ce code PYTHON utilise des bibliothèques, tels `os`, pour concevoir des fichiers JSON, qui sont les fichiers *Manifest* IIIF, et pour travailler avec les

2. Ce modèle minimal pour les modèles 3D, ainsi que d'autres avec plus de paramètres, peuvent être trouvés sur le GitHub d'IIIF. Cf *Basic Model in Scene - IIIF 3D*, URL : https://github.com/IIIF/3d/blob/main/manifests/1_basic_model_in_scene/model_origin.json (visité le 20/07/2024).

*CHAPITRE 7. ENRICHISSEMENT MÉTADONNÉE : VERS LES
SCENES IIIF*

65

fichiers d'entrée et de sortie.

Troisième partie

Évaluation des outils et méthodes : bilan et perspectives

Chapitre 8

Performance des applications

8.1 DaSCH Converter

En examinant le graphique des tailles des fichiers PNG avant et après conversion avec GIMP et PYTHON, il est clair que les fichiers PNG originaux sont généralement beaucoup plus volumineux que leurs versions converties en JPEG. Par exemple, le fichier `3DM_FN170_Mallet_0.5mm_filled_ZG`, dont la taille originale est de 69,3 Mo, est compressé à 285,4 Ko avec GIMP et à 145,2 Ko avec PYTHON. Cette réduction significative des tailles des fichiers PNG en JPEG est évidente pour tous les modèles étudiés.

Les différences entre les tailles obtenues avec GIMP et PYTHON montrent que PYTHON atteint des tailles de fichiers légèrement plus petites dans certains cas, ce qui pourrait refléter une compression plus efficace. Par exemple, pour le modèle `3DM_TT95_BL4_1mm_ZG`, PYTHON réduit la taille à 294 Ko, contre 490,6 Ko avec GIMP. Cette tendance indique que l'application PYTHONDaSCH Converter pourrait offrir une meilleure compression pour les fichiers JPEG, ce qui est important pour optimiser le stockage et la gestion des fichiers.

Nom du fichier PNG	Taille originale	GIMP (JPEG)	Python (JPEG)
3DM_FN170_Mallet_0.5mm_filled_ZG	69,3 Mo	285,4 Ko	145,2 Ko
3DM_TT95_BL4_1mm_ZG	38,9 Mo	490,6 Ko	294 Ko
3DM_FN312_FN313_FN749_FN789_Sandstone_0.5mm_filled_ZG	116,9 Mo	411,6 Ko	217,8 Ko
3DM_FN571_Sandstone_0.5mm_filled_ZG	42,9 Mo	244,8 Ko	134,1 Ko
3DM_TT95_BL3_1mm_ZG	25,1 Mo	562,7 Ko	383,6 Ko

Table 8.1 – Comparaison des tailles de fichiers avant et après conversion (GIMP et DaSCH Converter)

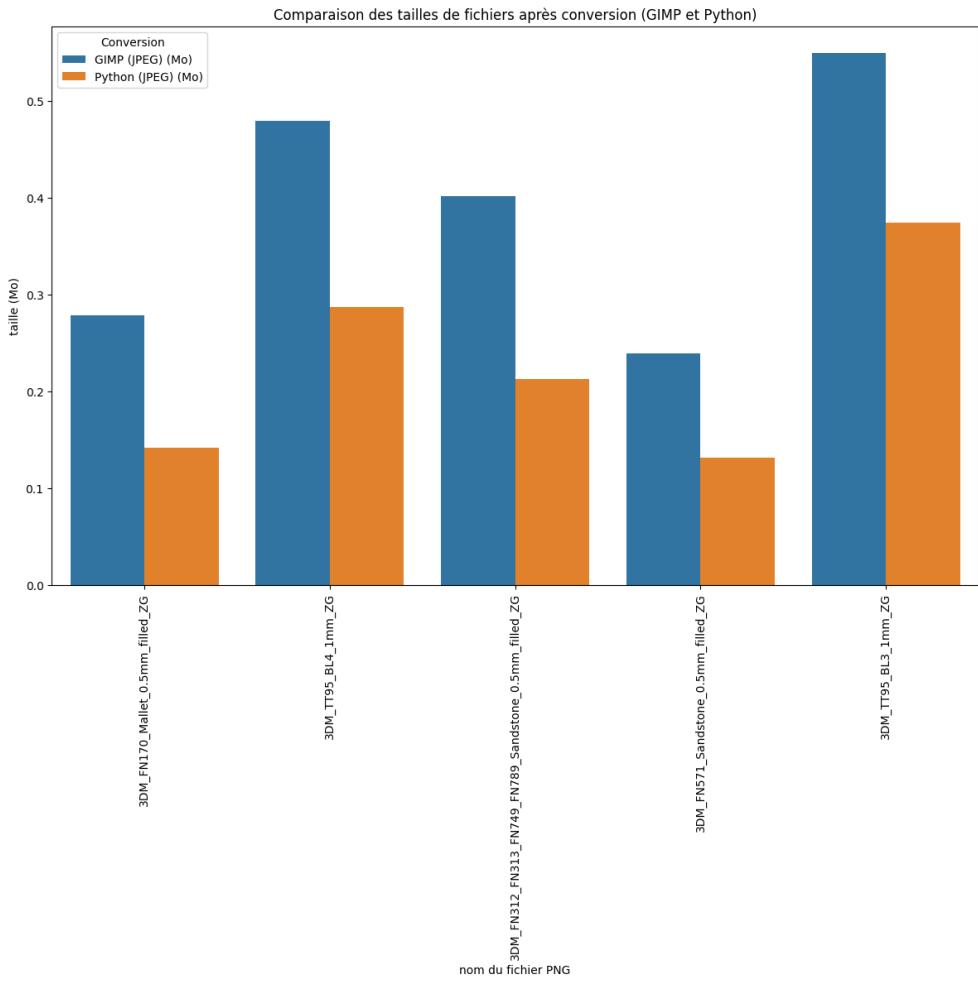


Figure 8.1 – Performance DaSCH Converter

8.2 DaSCH Mesh

L’analyse des données sur les tailles des fichiers avant et après conversion révèle des informations cruciales sur l’efficacité des méthodes de compression utilisées. La réduction de la taille des fichiers est un facteur important dans

le traitement et la gestion des données numériques, et les résultats obtenus permettent de comparer directement l'efficacité de différentes techniques de conversion.

L'analyse du graphique illustrant les tailles des fichiers avant et après conversion avec MeshLab et PYTHON met en évidence une réduction notable de la taille des fichiers grâce à l'utilisation de MeshLab. Par exemple, le fichier **3DM_FN170_Mallet_0.5mm_ZG**, initialement de 91 Mo, est réduit à seulement 6,78 Mo après conversion avec MeshLab, ce qui indique une compression significative. Cette tendance se confirme pour tous les modèles examinés, où MeshLab permet de réduire considérablement la taille des fichiers tout en limitant le nombre de sommets à 65 000. En comparaison, les conversions réalisées avec PYTHON montrent une augmentation significative des tailles de fichiers dans la plupart des cas. Par exemple, le modèle **3DM_TT95_BL4_-1mm_ZG** passe de 21,4 Mo avec MeshLab à 175,56 Mo avec Python, indiquant une augmentation notable de la taille des fichiers.

Nom du modèle 3D	Taille originale	MeshLab (65k sommets)	Python (65k sommets)
3DM_FN170_Mallet_0.5mm_ZG	91 Mo	6,78 Mo	21,56 Mo
3DM_TT95_BL4_-1mm_ZG	1,1 Go	21,4 Mo	175,56 Mo
3DM_FN312_FN313_FN749_FN789_Sandstone_0.5mm_filled_ZG	625 Mo	7,06 Mo	107,46 Mo
3DM_FN169_Mallet_ZG	157 Mo	14,4 Mo	14,7 Mo
3DM_FN1848_Mortar-3Cone_Imprints	22,8 Mo	12,8 Mo	4,6 Mo

Table 8.2 – Comparaison des tailles de fichiers avant et après conversion avec MeshLab et DaSCH Mesh

Cela suggère que l'application PYTHON DaSCH Mesh n'a pas encore été aussi efficace dans la gestion de la compression des fichiers pour les données 3D que MeshLab. Une explication plausible pour la baisse significative des performances pourrait résider dans l'utilisation de la bibliothèque Open3D. En effet, cette dernière n'offre pas les mêmes options de configuration que MeshLab. De plus, bien que la bibliothèque PyMeshLab offre les mêmes options de configuration que MeshLab¹, elle engendrait des boucles infinies lors de la conversion de fichiers volumineux. Pour de futures implémentations, il pourrait être envisagé d'utiliser Open3D ou d'explorer d'autres solutions au sein de PyMeshLab.

1. Voir PyMeshLab, *Decimation Quadric Edge Collapse with Texture*, 2024, URL : https://pymeshlab.readthedocs.io/en/latest/filter_list.html#meshing-decimation_quadric_edgeCollapse_with_texture (visité le 05/08/2024).

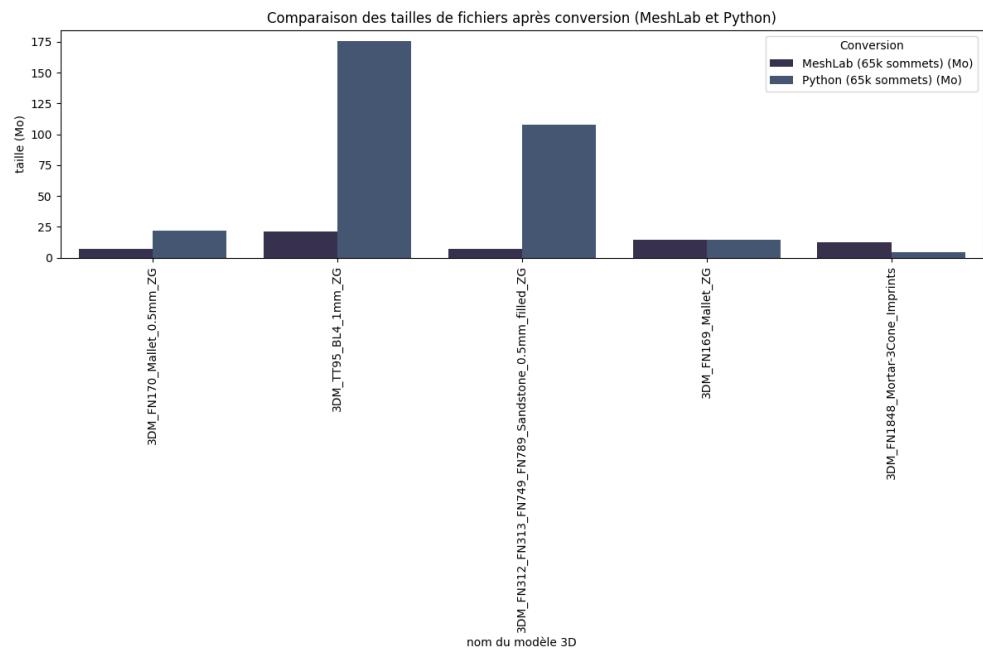


Figure 8.2 – Performance DaSCH Mesh

Chapitre 9

Perspectives d'évolution : fonctionnalités supplémentaires

9.1 Implémentations plus imminentes

Le développement logiciel est un processus collaboratif. Un programme est rarement le fruit du travail d'un seul développeur, mais plutôt le résultat d'une chaîne de contributions successives. Les codes sources sont partagés, modifiés, améliorés par différents acteurs au fil du temps. Cette nature collaborative du développement nécessite également une gestion rigoureuse des versions et une documentation précise pour garantir la cohérence du projet. Pour cette étude, les applications ont encore des implémentations à faire et voire des améliorations conséquentes.

En ce qui concerne la gestion d'erreur, les deux applications DaSCH Converter et DaSCH Mesh en présentent une gestion sommaire. Seules les erreurs 404 et 500 ont été traitées, respectivement avec un modèle de *template* qui affiche des images lors de l'erreur. Il serait intéressant de travailler ces erreurs pour les prochaines étapes du développement.

Il serait pertinent d'intégrer des écrans de chargement lors du traitement de fichiers volumineux. L'objectif est de maintenir l'utilisateur au sein de l'application pendant toute la durée de l'opération. Une simple barre de progression, dont de nombreux modèles sont disponibles sur Bootstrap par

exemple, pourrait suffire¹.

9.2 D'autres implémentations possibles

DaSCH Converter et DaSCH Mesh, comme dit, ont été codés en PYTHON à cause des raisons spécifiées dans la chapitre 4, à savoir : . néanmoins, ces applications peuvent, une fois finies et distribuées sur le *DockerHub* de DaSCH, être codées avec d'autres langages dont l'accent est mis sur la gestion de données et la performance, à l'instar de RUST et GO.

Les temps de réponse, en particulier pour les fonctionnalités qui nécessitent le traitement de grandes quantités de données, serait aussi une possibilité d'amélioration. Cela pourrait inclure la refactorisation du code pour le rendre plus efficace, l'implémentation de techniques de mise en cache, qui n'ont pas pu être développés lors du stage.

Par ailleurs, le recours à l'apprentissage automatique offre de nouvelles perspectives pour optimiser le processus de DaSCH Mesh. En entraînant un modèle sur un ensemble de données variées, il devient possible d'automatiser la classification des images et des objets, et ainsi de proposer des traitements adaptés à chaque type de fichier. Par exemple, en fonction de la présence ou de l'absence de textures, le modèle pourrait sélectionner automatiquement les algorithmes de conversion les plus pertinents (par exemple, un méthode A pour les fichiers sans texture et un méthode B pour les fichiers texturés). Cette approche permettrait de personnaliser davantage le processus de conversion et d'améliorer la qualité des résultats.

L'interface utilisateur de l'application pourrait également être enrichie afin de faciliter l'utilisation et d'améliorer l'expérience utilisateur. En effet, il serait pertinent de développer des fonctionnalités permettant d'identifier rapidement les fichiers téléchargés.

Pour répondre aux besoins des utilisateurs ayant de grands volumes de fichiers à convertir, il serait intéressant d'implémenter une fonctionnalité per-

1. ces icônes de chargement d'un composant ou d'une page sont appelés de *spinners*. Pour les exemples, Voir *Bootstrap 4.4 Documentation - Spinners*, URL : <https://getbootstrap.com/docs/4.4/components/spinners/> (visité le 23/08/2024).

mettant de traiter plusieurs fichiers simultanément. Cette option permettrait de gagner en productivité et de simplifier le flux de travail. En outre, la création d'une liste de téléchargement pourrait être proposée afin de permettre à l'utilisateur de suivre l'avancement des conversions et de télécharger les fichiers résultants de manière organisée.

En ce qui concerne DaSCH IIIF, même si l'application n'est pas finalisée, il est concevable de tirer quelques pistes : dans un premier temps, l'inspiration serait créer une application intuitive, similaire à DaSCH Converter et DaSCH Mesh, car en termes d'*UX*, elles sont intéressantes pour avoir intégré des éléments importants du *green coding*; deuxièmement, transformer le code en application et le diffuser sur DOCKER serait aussi un moyen de le rendre accessible au plus grand nombre d'utilisateurs possible.

Chapitre 10

Outils de travail

10.1 Aspects techniques

Tout d'abord, l'étude a bénéficié d'un MacBook Air équipé d'un processeur M1 d'Apple Silicon, offrant ainsi un outil performant et portable pour mener à bien les différentes tâches de la recherche.

De plus, un disque dur externe contenant l'ensemble des données du projet DaSCH a été mis à disposition, permettant ainsi la mise en œuvre de *pipelines* d'analyse automatisés. La portabilité de ce support a favorisé une grande flexibilité dans le déroulement des travaux.

En somme, l'ordinateur ainsi que la disponibilité des données ont permis de mener des analyses approfondies, notamment des tests de performance, favorisant ainsi une exploitation optimale des données. Le télétravail a également été facilité grâce à cette configuration. À l'issue du stage, le MacBook Air a été restitué. En revanche, le disque dur externe nous a été attribué.

10.2 Gestion du temps

Le déroulement du stage s'est distingué par une grande autonomie et une organisation méthodique, structurée en petites tâches hebdomadaires, gérées à l'aide de l'outil TRELLO. Ce tableau de tâches, partagé avec Rita Gautschi, la superviseure du stage, a permis un suivi rigoureux de l'avancement

des *pipelines* tout en servant de base pour les réunions hebdomadaires. Ces réunions, associées à cette approche structurée, ont favorisé une productivité réaliste, offrant un équilibre entre suivi et liberté. Elles ont également permis l'établissement d'un dialogue constructif, où des solutions, réflexions, et pistes d'amélioration ont pu être explorées pour débloquer certains problèmes, tout en offrant une compréhension approfondie du métier de la superviseure.



Figure 10.1 – Logo de l'application TRELLO

Au sein du projet DaSCH, la communication se faisait principalement par le biais d'une suite d'outils *Google Workspace*, mettant à disposition des chercheurs une boîte mail professionnelle et un espace de travail collaboratif. Les échanges instantanés, particulièrement fréquents, étaient privilégiés via la messagerie instantanée de *Google Chat*. Par ailleurs, l'outil *Google Meet* était largement utilisé pour les réunions en ligne, offrant ainsi la possibilité de conférences et de partages d'écran en temps réel.

En ce qui concerne les réunions, deux formats principaux étaient privilégiés. Des réunions d'équipe hebdomadaires permettaient de faire le point sur l'avancement du projet et de coordonner les tâches. Afin d'assurer un suivi régulier de l'exécution des codes, comme mentionné, il y avait aussi des réunions hebdomadaires dédiées. Ces rendez-vous réguliers ont permis de garantir une progression efficace du développement et de lever rapidement les éventuelles difficultés rencontrées. Cela n'a pas empêché de réaliser d'autres réunions entre les meMores de l'équipe, mais celles-là étaient moins

régulières.

En ce qui concerne le travail de codage, un accord a été établi dès le début du stage pour que les horaires du stage soient organisées selon la technique Pomodoro, qui est une technique créée en 1992 pour réaliser une gestion efficace du temps¹. Pour ce faire, l'application utilisée pour compter les minutes, nommée FLOW, a permis de structurer le temps en quatre sessions de 25 minutes avec des pauses de 5 minutes, suivies d'une longue pause de 30 minutes après le quatrième cycle. Cette méthode visait à améliorer la concentration lors de l'exécution des tâches, rendant le travail plus efficace. Dans le cadre de ce stage, la technique a contribué au développement de deux applications tout en aidant à trouver un rythme de travail adéquat.



Figure 10.2 – Logo de l'application FLOW, disponible que pour les *macOS* et *iOS*

La participation aux réunions, bien que non obligatoire pour les stagiaires,

1. D'après Cirillo, cette technique, répandu dans le milieu du mangement depuis le final des années 90, a pour but, entre autres, améliorer le processus de travail et d'études. Voir Francesco Cirillo, *The Pomodoro Technique*, Lulu.com, San Francisco, 2009, p. 20.

s'est révélée extrêmement bénéfique. Ces réunions ont offert une compréhension globale du travail, ainsi qu'une vision claire du fonctionnement quotidien d'un groupe de travail en sciences humaines. Cette immersion a permis la conception d'applications réalistes, tenant compte des spécificités de chaque groupe, et l'identification d'axes d'amélioration. De plus, cette expérience a enrichi la maîtrise de l'anglais et a fourni l'opportunité d'apprendre à gérer un projet de codage en collaboration avec plusieurs acteurs, facilitant ainsi une meilleure compréhension de la contribution de chaque partie à la réussite du projet.

En conclusion, ce stage a été une expérience enrichissante à plusieurs niveaux. L'autonomie dans l'organisation du travail, la méthode structurée de gestion du temps, et l'implication active dans les réunions ont non seulement permis la réalisation d'applications concrètes mais ont aussi favorisé un apprentissage profond du fonctionnement d'un projet de données en sciences humaines. Ces compétences acquises, tant techniques que relationnelles, constituent une base solide pour mes futurs projets professionnels.

Conclusion

Programmer, dans notre société contemporaine, ne se résume plus à écrire des lignes de code : c'est une activité qui s'inscrit dans un contexte social, politique et environnemental complexe. L'essor du concept de *green coding* met en lumière la nécessité de prendre en compte les impacts environnementaux et sociaux de la production de logiciels. L'IIIF, en tant que standard pour la manipulation et la présentation de ressources numériques, est souvent présenté comme une solution pour assurer la persistance des données.

L'IIIF, de par sa conception flexible et ouverte, offre indéniablement de perspectives pour garantir la persistance des ressources numériques. En tant que standard, il favorise l'interopérabilité entre différents systèmes et facilite ainsi la préservation à long terme des collections numériques, comme c'est le cas chez DaSCH. Sa modularité permet aux institutions culturelles et aux chercheurs d'adapter les outils et les services à leurs besoins spécifiques, sans être contraints par des normes trop rigides.

Sur le plan technique, le stage a permis d'atteindre l'objectif fixé, à savoir créer de *pipelines* automatisés en PYTHON afin de gérer les images 2D et les modèles 3D. Les résultats obtenus sont très encourageants, démontrant une efficacité significative par rapport aux logiciels tels que GIMP et MeshLab, tout en utilisant des méthodes similaires.

De plus, ce stage a ouvert de nouvelles perspectives de développement dans le domaine du patrimoine numérique. En particulier, la possibilité de créer DaSCH IIIF, un code PYTHON pour interagir avec les nouvelles APIs IIIF. Bien que cette solution soit encore en développement, elle offre de prometteuses perspectives d'association de fichiers JSON aux modèles 3D. Le défi consiste à lier les fichiers .GLB et .GLTF aux manifestes IIIF de manière

intuitive, en créant une structure similaire à celle de DaSCH Converter et DaSCH Mesh, accessible à tous types d'utilisateurs, des novices aux experts. Cette approche s'inscrit dans une démarche de conception centrée utilisateur, visant à offrir une expérience utilisateur complète.

Les compétences en modélisation de données se sont révélées essentielles pour ce projet, chaque *pipeline* nécessitant une bonne capacité d'abstraction et d'anticipation des problèmes. En revanche, bien que les connaissances en SPARQL soient utiles pour comprendre certains contextes, notamment ceux liés aux données RDF, elles n'ont pas été utilisées de manière intensive dans ce projet. Les problématiques liées à SPARQL étaient plutôt liées au développement de réponses pour les APIs IIIF *Presentation* et *Image*.

Il est important de noter que si PYTHON a été choisi pour sa flexibilité, sa scalabilité et sa facilité d'utilisation, d'autres langages comme Rust et Go pourraient être envisagés pour certaines tâches spécifiques, notamment la gestion efficace de données.

Enfin, il serait intéressant de publier l'application développée sur le DockerHub du DaSCH afin de la mettre à disposition de la communauté et de faciliter son utilisation par d'autres institutions confrontées à des problématiques similaires de gestion des images 2D et modèles 3D.

Sur le plan théorique, le *green coding* ne constitue pas seulement une démarche écologiquement responsable, il s'inscrit également dans les intérêts des grandes corporations qui, souvent, utilisent ce discours environnemental à des fins de *greenwashing*. Ce dernier obscurcit les véritables implications du développement technologique en réduisant la complexité des relations sociales à des questions individuelles de consommation consciente. En agissant ainsi, le *greenwashing* empêche une analyse critique des structures de pouvoir qui sous-tendent la production et la consommation technologiques.

Néanmoins, cette réflexion a enseigné qu'il est possible d'adopter d'autres manières de programmer, même si cette activité est ancrée dans une certaine configuration sociale qui vise souvent le profit. Il est intéressant d'intégrer les réflexions autour du *green coding* au processus de codage.

La réflexion sur les différentes conceptions des données révèle également un substrat social tout aussi crucial. Dans les sciences dites dures, la notion

de données est purement technique, alors que dans les sciences humaines, le contexte de production des données est souvent d'un grand intérêt. Bien que cette notion dans les sciences humaines ouvre la discussion sur la nature des données en relation avec les archives, elle n'aborde pas nécessairement comment ces données sont utilisées et dans quels buts. Il est important de rappeler que les données peuvent, dès leur conception, être l'objet de conflits. Ces conceptions distinctes suggèrent que programmer est également un acte politique, et tout comme les organisations poursuivent différents objectifs avec les données, la manière dont nous codons est fondamentale.

Une autre réflexion suscitée par la rédaction de ce travail concerne la gestion des données, qui implique la manière dont nous utilisons nos ressources. En effet, l'utilisation de *bits* correspond concrètement à l'utilisation d'un espace physique sur la planète. Ces données sont fréquemment stockées dans des centres de données. Dans ce contexte, notre pratique, qui pourrait sembler anodine, peut être perçue comme une pratique opérant un fétichisme dont nous devons nous rendre compte.

Enfin, programmer dans une société capitaliste, sans réfléchir aux implications de nos actions, relève de la dialectique du discours technologique. Comme d'autres auteurs l'ont observé, même dans une perspective de discussion sur la liberté, il est difficile d'affirmer qu'il existe une manière totalement libre de programmer dans une société où certains présupposés sont naturalisés, typiques d'une société capitaliste. Nous avons tendance à voir la discussion environnementale et technologique comme distincte de la lutte pour l'émancipation sociale.

Programmer, tout comme produire n'importe quel autre bien, s'inscrit dans un système économique qui prône le profit et l'accumulation de capital. La recherche d'alternatives plus justes et durables exige une réflexion approfondie sur le rôle des programmeurs dans la construction d'un avenir plus équitable et écologique.

En ce sens, programmer, c'est aussi rêver d'une société différente, où les individus pourraient bénéficier non seulement des avantages de la programmation, mais aussi de la préservation de l'environnement. Il ne s'agit pas d'une perspective morale individualiste, où la responsabilité incombe à l'in-

dividu, mais plutôt d'une perspective collective de changement social, où nous utilisons les contradictions inhérentes au système capitaliste pour proposer une transformation ou, dans une vision plus utopique, son démantèlement.

Table des figures

1.1	Logo du DaSCH	4
1.2	Organigramme du DaSCH	5
1.3	Gouvernance du DaSCH	6
1.4	Schéma de fonctionnement du DaSCH	8
1.5	Financement <i>ERI Dispatch</i> alloué par l’Assemblée fédérale suisse pour 2021-2024, en millions. Source : <i>State Secretariat for Education, Research and Innovation : Education, Research, and Innovation ERI 2021–2024</i> , URL : https://www.sbfi.admin.ch/sbfi/en/home/eri-policy/eri-21-24.html (visité le 17/07/2024).	12
2.1	Schéma : principe général d’interopérabilité de IIIF (trois applications différentes sont branchées à trois entrepôts IIIF) . . .	17
2.2	Mirador : comparaison entre images de différentes institutions.	18
2.3	Mirador : exemple d’image annotée, avec l’outil d’annotations du <i>viewer</i>	19
2.4	Mirador : zoom en haute définition dans un détail de la première image.	20
4.1	Logo de PYTHON	36
4.2	Logo de Flask 3.0.x	37
4.3	Logo de Bootstrap v4.x	39
4.4	Logo de Docker	41
4.5	Schéma de l’architecture Docker	43

5.1	Structure de l'application DaSCH Converter sur GitHub	56
5.2	Structure de l'application DaSCH Mesh sur GitHub	57
5.3	Page de conversion de fichier sur Smallpdf	57
6.1	Page principale du DaSCH Converter	61
6.2	Page principale du DaSCH Mesh	61
7.1	Code du fichier <code>model_origin.json</code>	64
8.1	Performance DaSCH Converter	70
8.2	Performance DaSCH Mesh	72
10.1	Logo de l'application TRELLO	78
10.2	Logo de l'application FLOW, disponible que pour les <i>macOS</i> et <i>iOS</i>	79

Bibliographie

- 3D Community Group - IIIF*, URL : <https://github.com/IIIF/3d> (visité le 03/08/2024).
- ADOBE, *Comprendre les codes de couleurs hexadécimaux*, URL : <https://www.adobe.com/fr/creativecloud/design/discover/color-hex-codes.html> (visité le 21/07/2024).
- ANCONA (M.), NANI (G.) et PACI (M.), « An object oriented approach to data persistence », *Microprocessing and Microprogramming*, 32–1–5 (1991), p. 263–270, URL : [10.1016/0165-6074\(91\)90357-y](https://doi.org/10.1016/0165-6074(91)90357-y) (visité le 05/07/2024).
- APPADURAI (Arjun), *The Social Life of Things : Commodities in Cultural Perspective*, dir. Arjun Appadurai, Cambridge University Press, Cambridge et New York, 1986.
- Basic Model in Scene - IIIF 3D*, URL : https://github.com/IIIF/3d/blob/main/manifests/1_basic_model_in_scene/model_origin.json (visité le 20/07/2024).
- BIBLISSIMA DOCUMENTATION, *Introduction à IIIF*, URL : <https://doc.biblissima.fr/iiif/introduction-iiif/#vision> (visité le 13/08/2024).
- BLOCH (M.), *Apologie pour l'histoire ou métier d'historien*, t. 3, Librairie Armand Colin, 1952 (Cahier des Annales).
- BOOTSTRAP, *About - Bootstrap v4.6*, 2023, URL : <https://getbootstrap.com/docs/4.6/about/overview/> (visité le 15/07/2024).
- *Introduction - Bootstrap v4.6*, URL : <https://getbootstrap.com/docs/4.6/getting-started/introduction/> (visité le 15/07/2024).
- Bootstrap 4.4 Documentation - Spinners*, URL : <https://getbootstrap.com/docs/4.4/components/spinners/> (visité le 23/08/2024).

- BOOTSTRAP-FLASK, *Bootstrap-Flask - Bootstrap-Flask 2.2.x documentation*, URL : <https://bootstrap-flask.readthedocs.io/en/stable/index.html> (visité le 03/08/2024).
- BRAUDEL (Fernand), *Civilisation matérielle, économie et capitalisme, XV^e-XVIII^e siècle. Tome 2. Les jeux de l'échange*, Armand Colin, Paris, 1967.
- CHALLON (Maxime), *CoursM2TNAH_Flask_code*, 2023, URL : https://github.com/MaximeChallon/CoursM2TNAH_Flask_code (visité le 15/07/2024).
- CIRILLO (Francesco), *The Pomodoro Technique*, Lulu.com, San Francisco, 2009.
- DARIAH-CH – *Digital Research Infrastructure for the Arts and Humanities in Switzerland*, URL : <https://www.dariah.ch/> (visité le 12/07/2024).
- DASCH, *DaSCH Swiss on Docker Hub*, URL : <https://hub.docker.com/u/daschswiss> (visité le 03/08/2024).
- *Governance / DaSCH*, URL : <https://www.dasch.swiss/governance> (visité le 12/07/2024).
 - *Mission / DaSCH*, URL : <https://www.dasch.swiss/mission> (visité le 29/08/2024).
 - *Strategic Goals / DaSCH*, URL : <https://www.dasch.swiss/strategic-goals> (visité le 12/07/2024).
- DOCKER, *What is Docker ? - Docker Docs*, URL : <https://docs.docker.com/get-started/docker-overview/> (visité le 03/08/2024).
- DOCUMENTATION (Flask), *Flask Documentation*, URL : <https://flask.palletsprojects.com/en/3.0.x/> (visité le 03/08/2024).
- FLASK-BOOTSTRAP, *Welcome to Flask - Flask Documentation*, URL : <https://pythonhosted.org/Flask-Bootstrap> (visité le 03/08/2024).
- FLORIDI (Luciano), *The Philosophy of Information*, Oxford University Press, Oxford, 2011.
- GAO (Xiao-Ping), XIN (John), SATO (Tetsuya), HANSUEBSAI (Aran), SCALZO (Marcello), KAJIWARA (Kanji), GUAN (Shing-Sheng), VALLDEPERAS-MORELL (Josep), LIS (Manuel) et BILLGER (Monica), « Analysis of cross-cultural color emotion », *Color Research & Application*, 32 (juin 2007), 2024-08-29, p. 223-229, DOI : 10.1002/col.20321.

- GARLAND (Michael) et HECKBERT (Paul S.), « Surface Simplification Using Quadric Error Metrics », dans *SIGGRAPH '97 Proceedings*, 1997, URL : <https://www.cs.cmu.edu/~garland/Papers/quadratics.pdf>.
- « Simplifying Surfaces with Color and Texture using Quadric Error Metrics », dans *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, ACM Press/Addison-Wesley Publishing Co., 1998, URL : <https://www.cs.cmu.edu/~garland/Papers/quadratic2.pdf>.
- GDMR Digital Workbench Login*, URL : <https://workbench.gdmrdigital.com/login.xhtml> (visité le 21/07/2024).
- Green Coding : How to Reduce the Carbon Footprint of Your Code - IBM Blog*, URL : <https://www.ibm.com/blog/green-coding/> (visité le 27/06/2024).
- HERRERACARLOS81, *PNG-to-JPEG : A Python program to convert all png images within a specific folder to jpeg files on a target folder*, 2023, URL : <https://github.com/HerreraCarlos81/PNG-to-JPEG> (visité le 12/08/2024).
- IIIF Image API 3.0*, URL : <https://iiif.io/api/image/3.0/> (visité le 29/08/2024).
- IIIF Presentation API 3.0*, URL : <https://iiif.io/api/presentation/3.0/> (visité le 20/07/2024).
- IIIF welcomes Julie Winchester as specification Editor*, URL : <https://iiif.io/news/2024/01/23/welcome-julie/> (visité le 21/07/2024).
- KOPYTOFF (Igor), *The Social Life of Things : Commodities in Cultural Perspective*, dir. Arjun Appadurai, Cambridge University Press, Cambridge et New York, 1986.
- LANGLOIS (Charles-Victor) et SEIGNOBOS (Charles), *Introduction aux études historiques*, Editions Kimé, 1ère édition : Librairie Hachette, Paris, 1898, Paris, 1992, p. 284.
- MARTIN (Ann Smart), « Material Things and Cultural Meanings : Notes on the Study of Early American Material Culture », *The William and Mary Quarterly* (, 1996), Consulté le 20 novembre 2021, DOI : <https://doi.org/10.2307/2946821>.

- MARX (Karl), *L'idéologie allemande : Critique de la philosophie allemande la plus récente dans la personne de ses représentants Feuerbach, B. Bauer et Stirner, et du socialisme allemand dans celle de ses différents prophètes*, Éditions Sociales, traduction par Henri Auger, Gilbert Badia, Jean Bau-drillard, Renée Cartelle, Paris, 1968.
- *Le Capital*, Gallimard, Traduit de l'allemand par Maximilien Rubel, première publication en 1867, Paris, 2008.
- MENESES (Ulpiano Bezerra de), *O objeto como documento*, 1980, URL : https://edisciplinas.usp.br/pluginfile.php/5577860/mod_resource/content/1/MENESES%2C%20Ulpiano%20Bezerra%20de.%200%20objeto%20como%20documento.%201980.pdf (visité le 29/08/2024).
- « A cultura material no estudo das sociedades antigas », *Revista de História*, 115 (août 1983), URL : <https://www.revistas.usp.br/revhistoria/article/view/61796> (visité le 29/08/2024).
 - « Memória e Cultura Material : Documentos Pessoais no Espaço Pú-blico », *Revista de História*, s/n (août 1998), URL : <https://periodicos.fgv.br/reh/article/view/2067> (visité le 29/07/2024).
- MILLER (Daniel), *Material Cultures : Why Some Things Matter*, Armand Colin, London, 1988.
- Mirador Demo*, URL : <https://deploy-preview-13--mirador.netlify.app/demo/> (visité le 04/09/2024).
- Mirador Project*, URL : <https://projectmirador.org/> (visité le 29/08/2024).
- MORANEUS, « Python Flask : A Comprehensive Guide from Basic to Ad-vanced », *Medium* (, 2023), URL : <https://medium.com/@moraneus/python-flask-a-comprehensive-guide-from-basic-to-advanced-fbc6ec9aa5f7>.
- MOXEY (Keith), « Les études visuelles et le tournant iconique », *Interme-diality : History and Theory of the Arts, Literature and Technologies*-11 (2008), Consulté le 20 novembre 2021, URL : <https://www.erudit.org/fr/revues/im/2008-n11-im3117/037542ar/>.
- OpenSeadragon IIIF TileSource Example*, URL : <https://openseadragon.github.io/examples/tilesource-iiif/> (visité le 04/09/2024).

- Petros Patias (éd.), *3D Research : Challenges in Cultural Heritage III : Complexity and Quality in Digitisation*, Springer, 2023.
- POMIAN (Krzysztof), *Le musée, une histoire mondiale. Tome I : Du trésor au musée*, Gallimard, Paris, 2020.
- POSTONE (Moishe), *Time, Labor, and Social Domination : A Reinterpretation of Marx's Critical Theory*, Cambridge University Press, Cambridge, 1996.
- PyAutoGUI, *PyAutoGUI Documentation*, 2024, URL : <https://pyautogui.readthedocs.io/en/latest/> (visité le 29/08/2024).
- PyMESHLAB, *Decimation Quadric Edge Collapse with Texture*, 2024, URL : https://pymeshlab.readthedocs.io/en/latest/filter_list.html#meshing_decimation_quadric_edgeCollapse_with_texture (visité le 05/08/2024).
- RAEMY (J.A.), *The International Image Interoperability Framework (IIIF) : raising awareness of the user benefits for scholarly editions*, Accessed : 21 May 2024, Bachelor's thesis, HES-SO University of Applied Sciences et Arts, Haute école de gestion de Genève, 2017, URL : <https://sonar.ch/hesso/documents/314853>.
- RAEMY (Julien A.) et GAUTSCHY (Rita), « Élaboration d'un processus pour les images 3D reposant sur IIIF », dans *Colloque Humanistica 2023*, Université de Genève, Suisse, 2023.
- Re3Data Repository : Swiss National Data and Service Center for the Humanities, URL : <https://www.re3data.org/repository/r3d100012374> (visité le 15/07/2024).
- REDE (Marcelo), « Estudos de cultura material : uma vertente francesa », *Anais do Museu Paulista : História e Cultura Material*, 8 (2001), Mis en ligne le 24 août 2009, consulté le 12 avril 2021, DOI : <https://doi.org/10.1590/S0101-47142001000100008>.
- ROSSENOVA (Lozana), *IIIF for 3D – making web interoperability multi-dimensional*, 2023, URL : %7B<https://pro.europeana.eu/post/iiif-for-3d-making-web-interoperability-multi-dimensional%7D> (visité le 14/05/2024).

SHI (Tiebing), « Mainstreaming, Counter-Co-Optation, and Depoliticization by a Counterculture », *Journal of Marketing Development and Competitiveness*, 8–3 (2014), p. 109-117.

State Secretariat for Education, Research and Innovation : Education, Research, and Innovation ERI 2021–2024, URL : <https://www.sbfi.admin.ch/sbfi/en/home/eri-policy/eri-21-24.html> (visité le 17/07/2024).

Technical Specification Group - IIIF 3D Community, URL : <https://iiif.io/community/groups/3d/tsg/> (visité le 29/08/2024).

Universal Viewer, URL : <https://universalviewer.io/> (visité le 29/08/2024).

VALBERG (Arne), *Light Vision Color*, John Wiley & Sons Ltd, Chichester, 2005.

Table des matières

Résumé	iii
Abstract	v
Dédicace	vii
Remerciements	ix
Introduction	xi
I Production au sein du DaSCH	1
1 Contexte DaSCH	3
1.1 Qu'est-ce que DaSCH ?	3
1.1.1 Gouvernance du DaSCH	5
1.1.2 Fondation du DaSCH	7
1.1.3 DaSCH et l' <i>Open Data</i> en Suisse	8
1.1.4 Stratégies de DaSCH	9
1.1.5 Financement du DaSCH	11
1.1.6 Notes conclusives	13
2 Contexte IIIF	15
2.1 Qu'est-ce que l'IIIF ?	15
2.2 Origines d'IIIF	16
2.3 Intégration d'IIIF pour la valorisation du patrimoine	16

2.4	IIIF API 3.0	20
2.5	Vers l'IIIF API 4.0	23
3	Persistance des données	27
3.1	Document, artefact et données	27
3.2	Culture matérielle et la persistance de données	29
4	Enjeux et défis de la numérisation patrimoniale	33
4.1	Les <i>pipelines</i>	33
4.1.1	Utilisation de logiciels libres chez DaSCH	33
4.2	Les limites de l'automatisation des logiciels	35
4.3	Utilisation de Python	36
4.4	Flask	37
4.5	Bootstrap	38
4.5.1	Flask-Bootstrap	40
4.6	Docker	41
II	De l'image au standard IIIF : conceptions des <i>pipelines</i>	45
5	Optimisation des images et objets pour IIIF : outils et méthodes	47
5.1	Conception du DaSCH Converter	48
5.1.1	Bibliothèques	48
5.1.2	Fonctionnalités du code	49
5.2	Conception du DaSCH Mesh	51
5.2.1	Bibliothèques	51
5.2.2	Fonctionnalités du code	54
5.3	Structure de DaSCH Converter et DaSCH Mesh	55
6	Expérience utilisateur	59
6.1	Intégration de l'UX dans les applications	59
6.1.1	Vers une <i>UX</i> inclusive	59
6.1.2	Fonctionnalités faciles à apprendre	60

TABLE DES MATIÈRES	95
6.1.3 La palette de couleurs	60
6.1.4 Fluidité de l' <i>UX</i>	62
6.1.5 UX et <i>green coding</i>	62
7 Enrichissement métadonnée : vers les <i>Scenes</i> IIIF	63
7.1 Conception du DaSCH IIIF	63
7.1.1 Modèles utilisés	63
7.1.2 Prérequis	64
7.1.3 Bibliothèques	64
III Évaluation des outils et méthodes : bilan et perspectives	67
8 Performance des applications	69
8.1 DaSCH Converter	69
8.2 DaSCH Mesh	70
9 Perspectives d'évolution : fonctionnalités supplémentaires	73
9.1 Implémentations plus imminentes	73
9.2 D'autres implémentations possibles	74
10 Outils de travail	77
10.1 Aspects techniques	77
10.2 Gestion du temps	77
Conclusion	81
Table des figures	85
Bibliographie	92
Table des matières	93