

TIGA

THINK & DO TANK Sciences, Sociétés et Industrie

Fiche-action RECHERCHE du projet TIGA de la Métropole de Lyon Saint-Etienne

Les Métropoles de Lyon et Saint-Etienne, en répondant conjointement en 2018 à l'appel à projet de l'Etat dans le cadre du programme « Territoires d'Innovation » du Plan d'Investissement d'Avenir (PIA 3), ont affirmé leur volonté de mener ensemble des actions communes pour le développement économique de leur territoire. Cette candidature est l'illustration qu'une alliance territoriale impliquant citoyens, acteurs du privé, ensemble des acteurs publics et collectivités doit accompagner et faire émerger une transformation profonde de l'industrie. Ce vaste projet intitulé « L'industrie [re]connectée et intégrée à son territoire et à ses habitants » est construit autour de 18 actions pilotées par la Métropole de Lyon. Elles répondent à un triple objectif :

- (1) Favoriser l'ancrage et la transformation de l'industrie en l'amenant vers le développement durable.
- (2) Créer un cadre de confiance aux échanges pour créer de la valeur.
- (3) Faciliter l'accès et la création d'emploi grâce au développement d'une culture industrielle, à l'adaptation des compétences et à l'amélioration de la mobilité.

Cette action a pour objectif de déceler, prototyper, expérimenter et valoriser sous un horizon de 3 ans des méthodes et des outils innovants pour la médiation industrielle. Elle vise à identifier des verrous scientifiques et proposer des solutions innovantes (supports matériels, numériques et méthodologiques) pour une interaction fluide et continue entre les acteurs du territoire. L'action 14 engage une dynamique qui vise à s'inscrire dans la pérennité, au-delà du projet TIGA Lyon Saint-Etienne, pour devenir un laboratoire permanent de co-création, de recherche-action et d'interactions entre les acteurs du territoire. L'action 14 est un laboratoire de co-création et de recherche-action associant l'Université de Lyon, l'agence d'Urbanisme de Lyon, TUBA et la métropole de Lyon (ERASME, Direction de la Prospective et du Dialogue Public) qui s'enrichissent mutuellement et se combinent autour d'un processus commun de production.

Dans le cadre du projet TIGA, le LIRIS (Laboratoire d'InfoRmatique en Image et Système d'information) a pour objectif d'étudier les mécanismes de médiation, et leurs possibles applications. Il s'agit de détecter, étudier, puis proposer de nouvelles modalités de médiation sur le territoire qui permettent un rapprochement des acteurs qu'ils soient citoyens, issus de l'industrie, des collectivités, d'organismes de formation ou de recherche. Ce rapprochement peut se faire traditionnellement par la rencontre, l'objectif étant ici de voir comment favoriser celle-ci et de l'outiller afin de rendre le contexte plus favorable. Il s'agit aussi d'apporter des contenus qui devront pouvoir s'inscrire dans la durée et ainsi favoriser un échange à plus long terme permettant une méthodologie de suivi des acteurs, du territoire et de son évolution. Ce dernier point est important car il permettra la mise en place de modalités d'observation de ce qui est fait afin de pouvoir mesurer les éventuels impacts, apporter des améliorations, voire proposer un transfert pour dissémination (vers d'autres territoires ou d'autres cibles par exemple).

Afin de mieux répondre à ce besoin de reconnexion, il nous fallait effectuer une veille de l'existant pour s'inspirer et être le plus efficace dans la conception d'outils de médiation.

Livrable 2020 : Médiations et jeux , vers une meilleure interaction usagers-territoire

Un premier livrable fourni en décembre 2020 cherchait à regrouper les outils de médiation. Dans cette phase de veille, le premier axe d'“augmentation” choisi est l'utilisation de la ludification dans les processus d'interactions entre publics et d'acculturation. L'étude est basée sur une trentaine de jeux. Même si pour l'instant le filtre employé se veut assez large afin d'étudier les mécanismes et éventuellement les détourner pour une médiation plus orientée vers les objectifs de l'Action 14, un travail complémentaire est actuellement en cours avec une orientation plus forte vers la médiation industrielle.

Lien en direction du livrable 2020^{^1}, Médiations et jeux , vers une meilleure interaction usagers-territoire.

- Veille^{^2} : Veille sur les outils de médiations lié au thème de l'industrie.
- Mind map^{^3} : Carte mentale de la veille des outils de médiations lié à TIGA donnant une autre approche de visualisation de celle-ci afin de mieux les catégoriser.

Livrable 2022 : Développement technique pour des outils de médiations industrielles

Dans ce contexte de reconnexion et grâce à la veille effectuée en amont, le LIRIS peut passer à la deuxième phase de son objectif qui est de concevoir, prototyper, et transférer des nouvelles modalités de médiation industrielle mobilisant les cultures numériques du territoire. Ces modalités peuvent être sous différentes formes, telles que des représentations 3D numériques d'un territoire, des outils de médiation ou bien des socles techniques pour aider à la construction d'outils de médiation. Ces socles techniques cherchent à être reproductibles et documentés afin qu'ils puissent être utilisables sous différentes thématiques, le tout dans un contexte open source.

Sommaire :

- Développement
 - 3D Tiles
 - Py3DTiles
 - Py3DTilers
 - UD-Viz
 - Couleurs et textures
 - intégration de contenus multi-médias
 - intégration de couches de données 2D
 - Démos UD-Viz
 - Démo Py3DTilers
 - Démo UI-driven
 - Démo Vallée de la chimie
 - Docker
- Conclusion A compléter

Développement

Modèles 3D

L'interaction avec des données urbaines nécessite de résoudre un certain nombre de problématiques, telles que la visualisation massive d'objets 3D (bâtiments, ponts, végétation, etc) et la liaison des ces objets avec de la donnée sémantique. Pour répondre à ces problématiques, nous avons fait le choix d'utiliser le standard **3D**

[Tiles](#), décrit par Cesium et l'OGC. 3D Tiles a été pensé pour aider à la visualisation massive de contenu géospatial 3D. Ce standard permet de décrire un *tileset* : un arbre de tuiles 3D. Chaque tuile contient des modèles 3D auxquels sont associés des données sémantiques. Un tileset permet une organisation spatiale des tuiles, optimisée pour le rendu d'objets 3D urbains, notamment grâce à la hiérarchisation spatiale des objets, mais aussi grâce aux niveaux de détails. Les niveaux de détail permettent d'alterner entre des géométries plus ou moins détaillées en fonction des besoins, par exemple en affichant des modèles très simplifiés de loin et détaillés lorsqu'on est proche.

Les tuiles peuvent avoir différents formats :

- Batched 3D Model (B3DM) : Modèles 3D hétérogènes.
- Instanced 3D Model (I3DM) : Instances de modèles 3D.
- Point Cloud (PNTS) : Nombre massif de points colorés.
- Composite : Mélange de différents formats.

Dans les outils développés dans le cadre de TIGA, nous utilisons principalement le format B3DM. Ce format décrit les géométries à l'aide du format glTF, libre et facilitant le transfert et rendu de modèles 3D sur le web. Chaque tuile contient un ensemble de modèles 3D représentants par exemple des bâtiments ou des arbres. Chaque modèle 3D peut-être associé à un ensemble d'informations sémantiques.

La méthode utilisée pour créer les 3D Tiles peut avoir un impact direct sur la visualisation des objets. Il est donc nécessaire de disposer d'outils permettant de tester différentes méthodes de tuilage afin d'optimiser le rendu et la visualisation des modèles 3D. Il y a aussi le besoin d'offrir à l'utilisateur des outils lui permettant de créer des 3D Tiles depuis différentes données, en y ajoutant de la couleur, des textures et des niveaux de détail. C'est dans ces objectifs qu'ont été développé [Py3DTiles](#) et [Py3DTilers](#).

Py3DTiles

[Py3DTiles](#) est une librairie libre permettant de produire des [3D Tiles](#). Elle offre un ensemble de fonctionnalités pour écrire des 3D Tiles depuis de la donnée. Originellement développé par [Oslandia](#), cette bibliothèque a été enrichie et robustifiée au cours du projet TIGA.

Le premier objectif des modifications et ajouts apportés à Py3DTiles est de faciliter la production de modèles 3D Tiles et leur personnalisation. Pour cela, nous avons notamment ajouté le système de création d'extension : l'utilisateur peut étendre le standard afin d'y ajouter ses propres fonctionnalités. Nous avons aussi introduit dans la librairie la création de matériaux. Ces matériaux permettent de changer l'apparence des modèles 3D en y appliquant des couleurs et des textures.

Le second objectif du travail sur Py3DTiles est de s'assurer que les 3D Tiles produits avec cette librairie puissent être visualisés avec différents outils. Dans ce but, des développements ont été effectués afin de vérifier que les 3D Tiles soient fidèles au standard. Cela permet d'être certain que les 3D Tiles créés avec Py3DTiles puissent être manipulés par tous les autres outils suivant aussi le standard.

Py3DTilers

[Py3DTilers](#) est un outil libre développé dans le cadre du projet TIGA. Il permet de créer des 3D Tiles depuis différents formats de données: [OBJ](#), [GeoJSON](#), [IFC](#) et [CityGML](#). Ces formats de données sont utilisés pour représenter des données géographiques et urbaines, 2D ou 3D. Le support de ces différents formats permet de créer des 3D Tiles depuis un grand nombre de données, où chaque donnée peut être spécialisée pour

représenter des objets de la ville en particulier. Par exemple, le format CityGML permet de représenter les géométries des bâtiments et du relief. Le format IFC décrit quant à lui plus en détails l'intérieur des bâtiments.

Py3DTilers se base sur la librairie [Py3DTiles](#), décrite précédemment, pour produire des 3D Tiles. Cet outil vient rajouter des *Tilers*, permettant chacun de transformer un format précis de données en 3D Tiles. Py3DTilers intègre également un grand nombre d'options et de fonctionnalités pour personnaliser la création, parmi lesquelles :

- **La création de niveaux de détails:** permet d'ajouter un ou plusieurs niveaux de détail aux géométries. Cela permet de fluidifier le rendu en affichant des modèles 3D plus ou moins détaillés.
- **La reprojection des données:** permet de modifier le système de projection utilisé pour les coordonnées. Cela permet par exemple de visualiser dans un même contexte des données ayant originellement des systèmes de projection différents. Cela permet aussi de s'adapter aux contraintes pouvant être imposées par les outils de visualisation de 3D Tiles.
- **La translation et mise à l'échelle des modèles 3D:** permet de transformer la donnée en changeant l'échelle ou en déplaçant les géométries. Cela permet notamment de corriger les erreurs de placement ou de taille. Ces fonctionnalités permettent aussi de placer de la donnée non géo-référencée dans un contexte géospatiale, ou inversement de placer de la donnée géo-référencée à des coordonnées centrées autour de (0, 0, 0).
- **L'export en modèle OBJ:** permet d'exporter les géométries au format OBJ. Ce format peut être visualisé dans la majorité des outils, ce qui permet de rapidement vérifier l'apparence des géométries.
- **L'ajout de textures:** si la donnée d'entrée est texturée, l'utilisateur peut choisir de conserver ou non les textures. Lorsque les textures sont conservées, elles sont stockées dans des atlas de textures sous forme de fichiers JPEG.
- **La création de 3D Tiles colorés:** permet d'ajouter des couleurs aux 3D Tiles. Les couleurs peuvent être choisies par l'utilisateur. Les couleurs peuvent être appliquées en fonction d'attributs des modèles (par exemple hauteur du bâtiment) ou en fonction du type des objets (toit, mur, sol, etc.).

Toutes les options de Py3DTilers permettent d'obtenir un outil offrant une grande polyvalence. De plus, l'architecture de code permet de rapidement ajouter des nouvelles fonctionnalités ou de personnaliser celles qui existent déjà. Py3DTilers permet à l'utilisateur un contrôle total du processus de création de 3D Tiles, que ce soit via les options ou via la modification du code. L'outil permet de personnaliser les 3D Tiles produits, de tester de nouvelles modalités de répartition des tuiles ou de création de niveaux de détail. Py3DTilers se veut l'outil idéal pour innover ou expérimenter autour des 3D Tiles, et proposer des améliorations du standard.

En complément de la documentation présente sur le dépôt github de Py3dtilers un article scientifique a été rédigé sur Py3dTilers et ses fonctionnalités dans le cadre d'une conférence internationale ([Smart Data Smart Cities & 3D GeoInfo](#)) qui regroupe des experts de l'analyse des villes, des SIG, des jumeaux numériques, des villes intelligentes et de la science des données ([Article Py3DTilers](#)).

Les 3D Tiles générés avec l'outil Py3DTilers peuvent être [visualisés avec différents logiciels](#). Néanmoins, nous utilisons dans la majorité des cas le visualisateur UD-Viz, un logiciel libre qui a été en partie développé au cours du projet TIGA.

UD-Viz

[UD-Viz](#) est une librairie JavaScript permettant de visualiser de la donnée urbaine et d'interagir avec cette donnée via navigateur Web. UD-Viz se base sur [iTowns](#), développé par l'IGN, pour pouvoir charger et afficher

des couches de données géographiques ainsi que des modèles 3D Tiles.

Dans le cadre du projet TIGA, UD-Viz a été enrichi afin d'offrir à l'utilisateur de nouvelles interactions avec la donnée ainsi qu'un meilleur contrôle sur l'apparence des modèles 3D. Pour cela, des développements ont été effectués afin de permettre l'intégration de contenus multi-médias et de couches de données 2D. Ces contenus permettent d'enrichir la visualisation et d'améliorer sa compréhension. De plus, le système de gestion du style des 3D Tiles a été revu afin de permettre une plus grande liberté sur l'affichage des couleurs et des textures des modèles 3D. Cela permet par exemple d'afficher des bâtiments colorés en fonction d'un attribut (hauteur, pollution, densité, etc) ou d'afficher alternativement des modèles colorés et des modèles texturés afin d'offrir plusieurs modalités de visualisation.

Couleurs et textures

UD-Viz intègre une gestion des styles permettant d'appliquer des couleurs aux modèles des 3D Tiles. Ces styles peuvent être appliqués modèle par modèle ou à un ensemble de modèles. Auparavant, un style arbitraire était appliqué par défaut à tous les 3D Tiles lors du chargement. Le problème avec ce style par défaut est qu'il détruit les styles déjà présents dans les modèles 3D, pour appliquer un style arbitraire à l'intégralité des modèles. En conséquence, il n'était pas possible de charger des 3D Tiles avec des textures ou des couleurs, car elles étaient détruites dès le chargement.

Des modifications ont été apportées à cette gestion des styles afin de laisser à l'utilisateur le choix entre:

- Appliquer un style par défaut aux 3D Tiles lors du chargement (comme auparavant).
- Conserver le style déjà présent dans les 3D Tiles et ne pas appliquer de style par défaut.

La deuxième option permet un plus grande diversité des styles, puisque ces derniers ne sont plus limités à une couleur unique. De plus, les différentes options de [Py3DTilers](#), l'outil permettant de créer les 3D Tiles, offre beaucoup d'options pour ajouter des couleurs et des textures aux 3D Tiles. Ces couleurs et textures peuvent maintenant être chargées dans UD-Viz.

En plus de ce travail sur le style par défaut des 3D Tiles, des modifications ont été apportées à la gestion du style dans UD-Viz afin de corriger des erreurs et de robustifier le code.



Figure 1 : UD-viz démonstration, application des textures sur 1er arrondissement de Lyon

Intégration de contenus multi-médias

Afin d'améliorer la compréhension d'un territoire, nous voulions contextualiser les modèles 3D de bâtiments avec des multi-médias. Ces multi-médias peuvent être des images d'archives, des vidéos d'acteurs du territoire ou des photos d'un observatoire photographique. Ce nouveau contenu apporte une autre dimension à la déambulation dans un environnement 3D et l'améliore avec plus d'informations sur celui-ci. Nous avons donc développé une méthode d'intégration de multi-medias qui se base sur la librairie [UD-viz](#). L'intégration est découpée en deux parties :

- [Configuration du fichier JSON](#) : documentation sur la configuration du fichier JSON afin de lier contenus multi-medias et positions géospatiales.
- [Episode visualizer object](#) : programme d'intégration des multi-médias avec comme point d'entrée le fichier de configuration JSON.

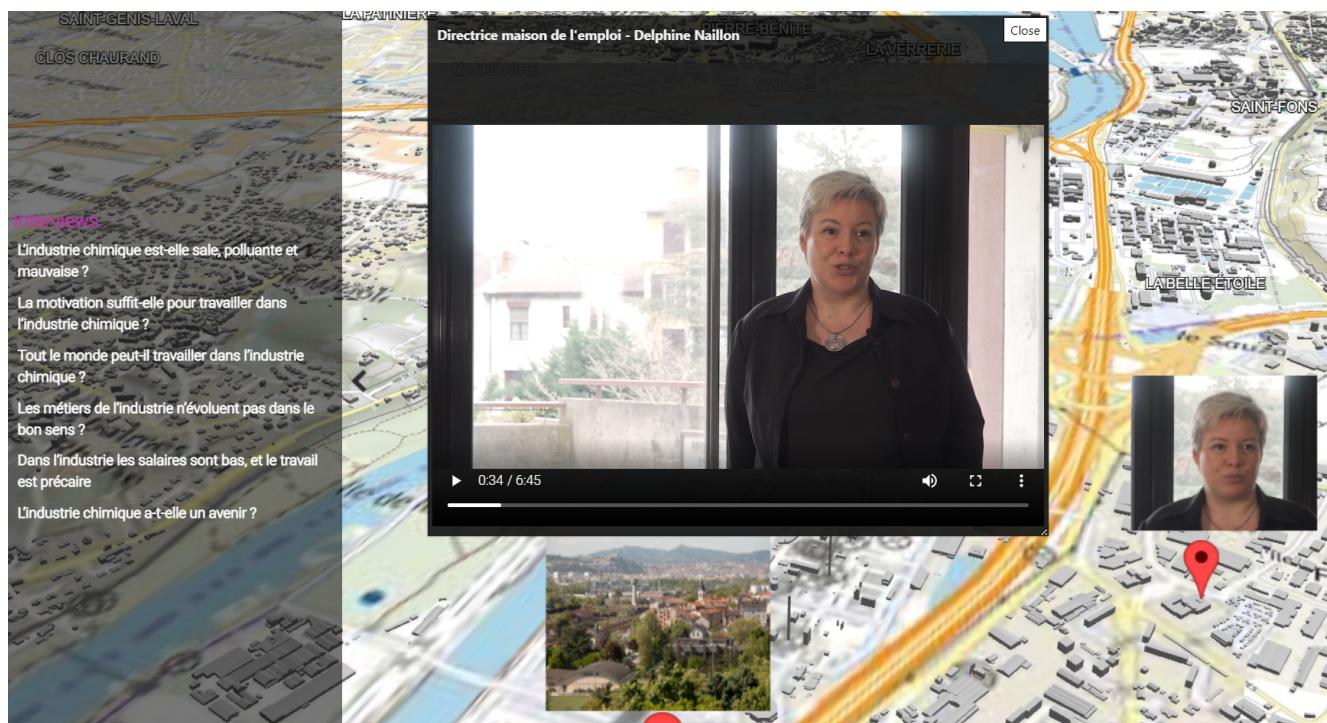


Figure 2 : "Derrière les fumées", intégration d'une interview de la directrice de la maison de l'emploi dans la représentation numérique de la vallée de la chimie

Cette approche a pu être utilisée dans le contexte du webdocumentaire du projet "Derrière les fumées". Des interviews d'acteurs de la vallée de la chimie ont été disposées dans la représentation numérique de cette zone afin d'avoir plus d'informations sur l'activité de ce territoire.

Ce socle technique est documenté dans la librairie UD-Viz ainsi que dans un article scientifique sur l'intégration de multimedia dans une représentation 3D numérique soumis à la même conférence internationale que Py3dTilers, [Smart Data Smart Cities & 3D GeoInfo](#) (Article intégration de multimedia dans une représentation 3D numérique).

Intégration de couches de données 2D

Dans cette idée d'une meilleure compréhension, au-delà d'une représentation 3D, nous nous sommes intéressés à une visualisation 2D du territoire. En effet, les données 2D urbaines apportent une nouvelle vision sur un territoire et donnent plus d'informations sur celui-ci, comme l'accessibilité de certains quartiers grâce au réseau de transport en commun ou bien les zones végétalisées d'un arrondissement.

L'intégration de couches de données se fait via des services [WFS](#) (Web Feature Service) et [WMS](#) (Web Map service) dans la bibliothèque [UD-viz](#). Ce type de données nous permet de représenter différentes géométries dans UD-Viz; cela peut être des polylinéaires, des polygones ou des images PNG/JPEG. Cet ajout de couches de données apporte une nouvelle vision à la représentation numérique et permet de contextualiser les modèles 3D de bâtiments.

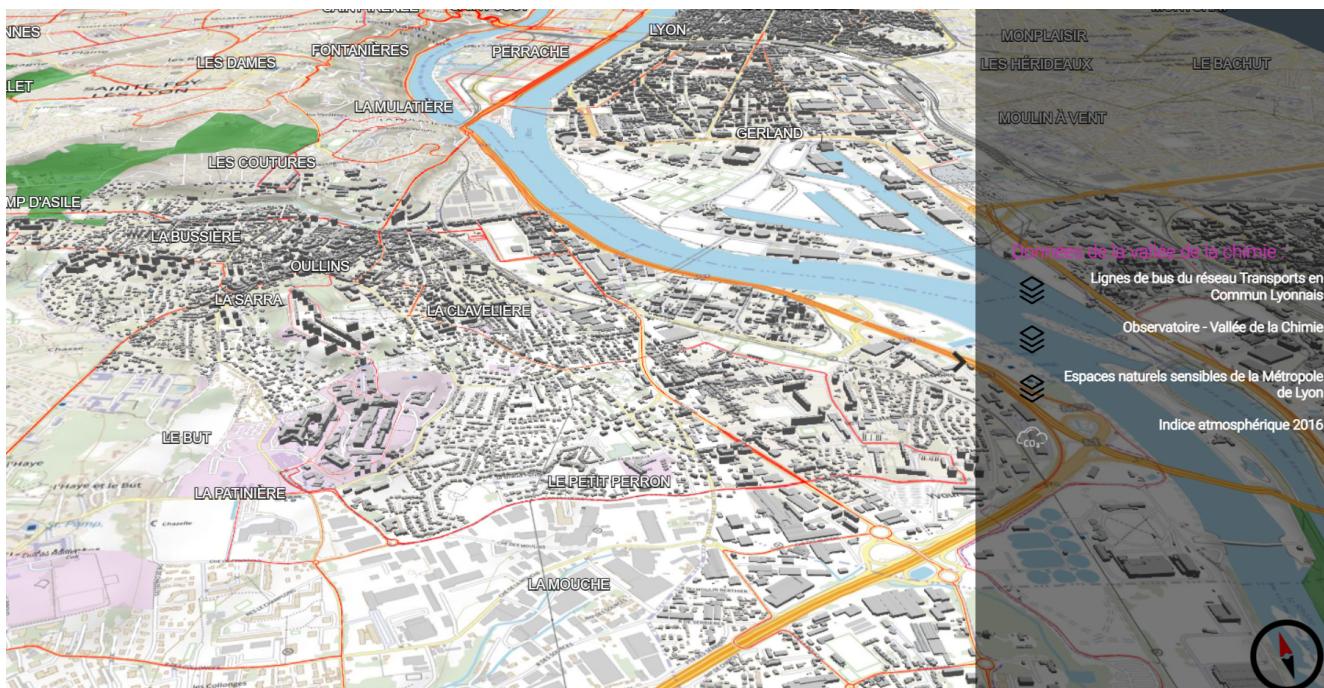


Figure 3 : "Derrière les fumées": intégration de données urbaines 2D tel que lignes de transports en communs (lignes rouges) et les espaces naturels sensibles de la métropole de Lyon (polygones verts) dans la représentation numérique de la vallée de la chimie

Un cas d'exemple de cette intégration peut être retrouvé dans la [démonstration](#) de la vallée de la chimie avec un jeu de données 2D disponible sur le site [open data Grand Lyon](#). Nous avons intégré le réseau de transport en commun pour montrer l'accessibilité à certains sites industriels de la vallée de la chimie. Nous avons également intégré les espaces naturels sensibles pour donner une autre vision de ce territoire.

Démos UD-Viz

[UD-Viz-Template](#) est un template d'application permettant de créer rapidement des démonstrations basées sur la librairie [UD-Viz](#). Ces démonstrations permettent d'illustrer des fonctionnalités et/ou des données particulières. Grâce à ce template, différentes démonstrations ont pu être réalisées afin d'alimenter le projet TIGA. En voici la liste présentée ci-dessous :

Démonstration Py3DTilers

- [Code](#) source de la démonstration ainsi que sa documentation.
- [Docker](#) pour reproduire l'application.

- [Démo en ligne](#)

Cette démo propose un ensemble de 3D Tiles créés avec les outils de Py3DTilers. On y retrouve des 3D Tiles de bâtiments, ponts, relief et cours d'eau, certains texturés et d'autres colorés.

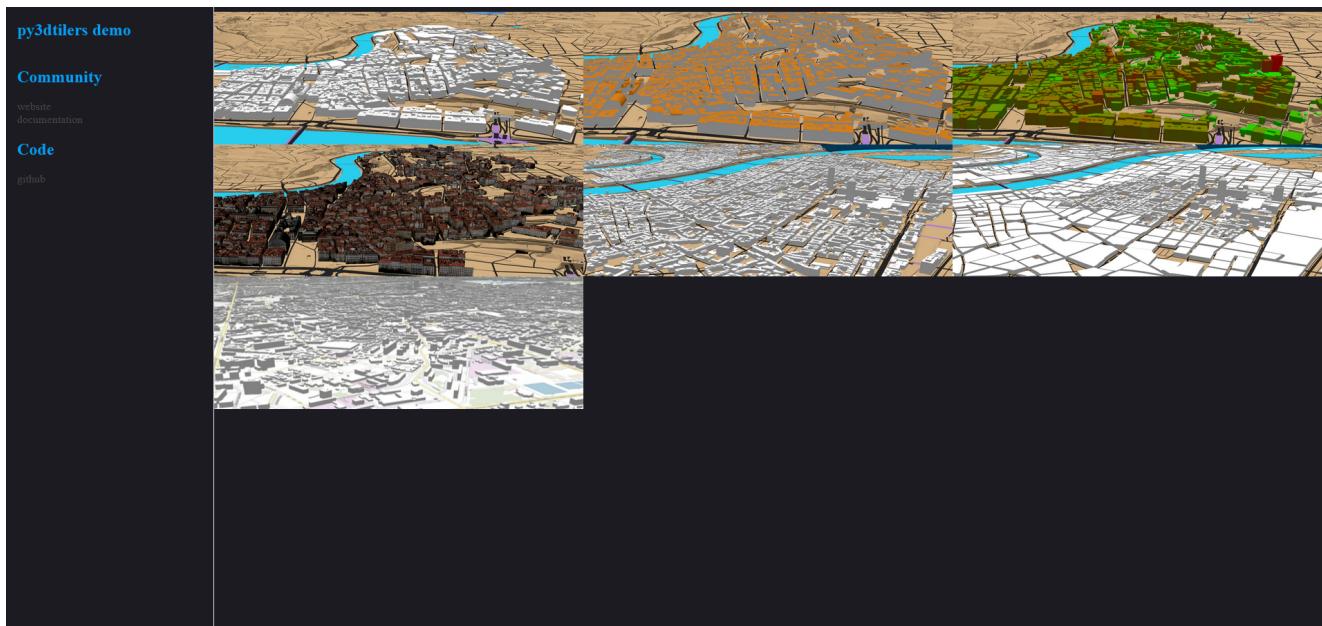


Figure 4 : Mosaïques des différentes démonstrations développées grâce à l'outil Py3tilers. Une démonstration de la ville texturée, une autre avec de la couleur, la ville avec différent niveau de détail..

Les 3D Tiles ont été créés à partir de couches de données publiques issues du site du [Grand Lyon](#) et de l'[IGN](#). Les modèles 3D des ponts et du relief sont créés à partir de la donnée CityGML du Grand Lyon, par l'intermédiaire d'une [base de données 3DCityDB](#). Les fleuves et les routes sont créés à partir des données GeoJSON de l'[IGN](#). Les bâtiments sont quant à eux créés soit à partir des fichiers CityGML soit à partir de fichiers GeoJSON. Une [documentation](#) a été créée pour expliquer plus en détails le processus de création des 3D Tiles depuis de la donnée publique.

La démo introduit aussi de nouvelles modalités de visualisation des 3D Tiles. Elle implémente notamment un nouveau fonctionnement des niveaux de détails des modèles 3D. Par défaut, le niveau de détails se raffine en fonction du zoom de la camera : plus la camera est proche d'un modèle, plus ce dernier est détaillé. Ici, on offre la possibilité d'utiliser la souris comme une "loupe": les modèles proches de la souris de l'utilisateur sont raffinés afin d'obtenir des modèles plus détaillés sans avoir besoin de bouger la caméra.

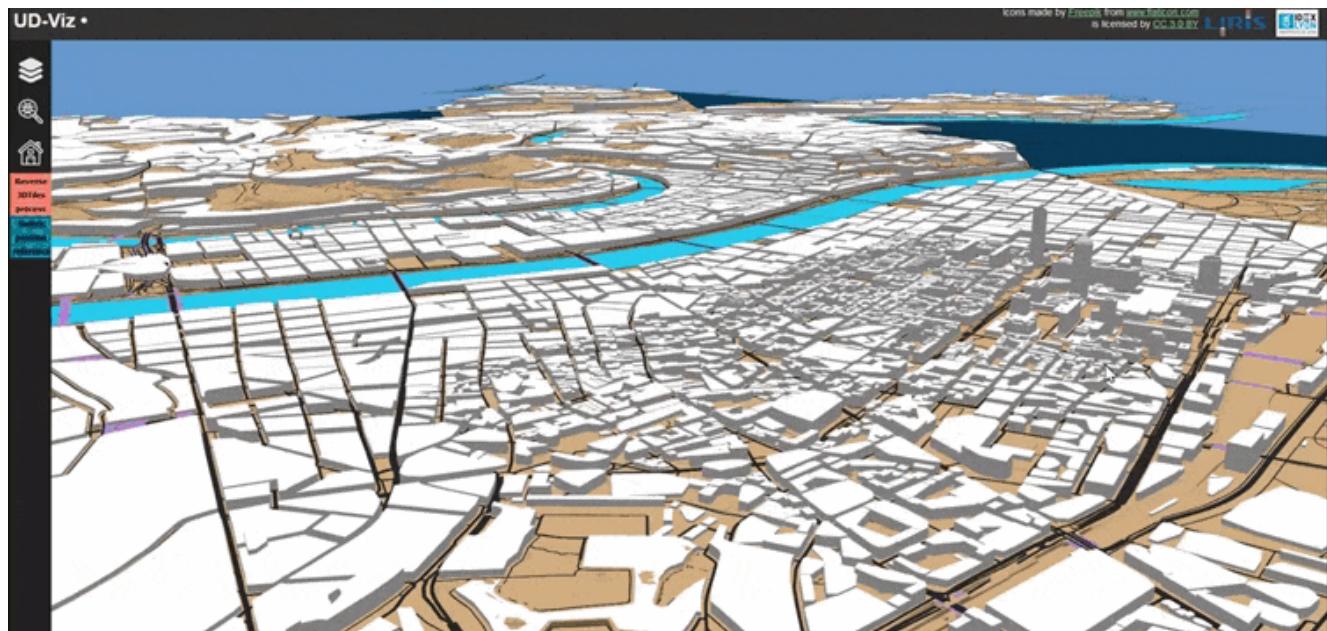


Figure 5 : Démonstration montrant le raffinement du niveau de détail des bâtiments de Lyon en fonction de la position de la souris

Démo UI-driven

- [Code](#) source de la démonstration ainsi que sa documentation.
- [Docker](#) pour reproduire l'application.
- [Démo en ligne](#)

Cette démo permet de calculer la hauteur de routes en les plaçant sur le relief. Pour cela, les routes doivent être contenues dans des fichiers GeoJSON. Ces fichiers peuvent ensuite être glissés/déposés dans la démonstration. Les routes seront affichées au fur et à mesure du calcul. Une fois toutes les routes placées sur le relief, de nouveaux fichiers GeoJSON contenant les routes aux bonnes altitudes sont téléchargés. Le processus pour créer des routes 3D depuis de la donnée de l'IGN est détaillé dans cette [documentation](#).

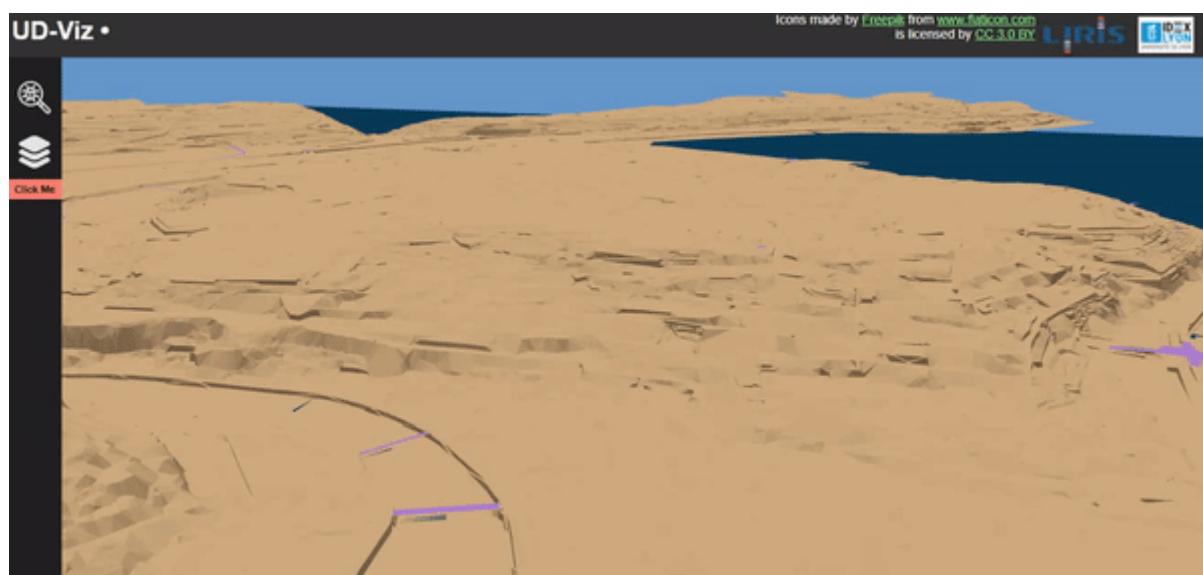


Figure 6 : Démo UD-Viz sur le calcul des hauteurs de routes contenu dans un fichier GeoJSON à l'aide d'un glissé/déposé dans l'application web UD-Viz

Démo Vallée de la chimie

- [Code](#) source de la démonstration ainsi que sa documentation.
- [Docker](#) pour reproduire l'application.
- [Démo en ligne](#)

La démo [Vallée de la chimie](#) a été développé dans le cadre d'un web-documentaire en collaboration avec [Interfora](#), un pôle de formation au métiers de la chimie. Ce projet "Derrière les fumées" a pour but de déconstruire les idées préconcues que peuvent avoir les citoyens de la métropole de Lyon sur ce territoire. Dans cette problématique, nous nous sommes orientés vers une représentation numérique de la vallée de la chimie pour mieux la comprendre et y disposer du contenu multi-média afin d'apporter plus d'informations. Grâce à l'outil d'intégration de multi-média dans la bibliothèque UD-Viz, nous avons pu disposer des d'interviews d'acteurs de ce territoire et des données urbaines dans la maquette numérique. Ces différents éléments sont disposés à des points d'intérêt en lien avec le thème de l'industrie.



Figure 7 : Capture du web-dcoumentaire "Derrière les fumées", avec les deux panneaux d'affichages, les différentes interviews d'acteurs de la vallée de la chimie et les données urbaines récupérées sur l'open data Gradn Lyon

Pour produire cette démonstration nous avons récupéré les données CityGML disponibles sur l'open data Grand Lyon pour ensuite les traiter avec Py3DTilers et générer les 3D Tiles ([3D Tiles générés avec Py3DTilers](#) : bâtiments, relief, routes, ponts, fleuves) de la vallée de la chimie. En complément de ces modèles 3D, nous avons intégré les [photos de l'observatoire de la vallée de la chimie](#) produites par Florent PERROUD et les images d'archives de la catastrophe de Feyzin en libre accès sur le site la bibliothèque municipale de Lyon ([lien des photos](#)).

Docker

Docker est un outil permettant de lancer des applications dans un contexte déterminé et isolé. Les applications ne sont ainsi pas exécutées directement sur la machine hôte, mais dans un contexte maîtrisé. Une

application contenue dans un Docker sera toujours exécutée de la même manière, les versions des différents composants sont figées. Cela permet de s'assurer que l'application pourra être utilisé sur n'importe quelle machine, sans soucis d'installation et sans erreur de version des logiciels. L'utilisation de Docker permet d'éviter qu'une application fonctionnelle ne devienne inutilisable après quelques temps à cause de mise à jours de la machine hôte ou de l'application elle-même.

C'est pourquoi toutes les applications développées lors du projet possèdent des versions contenues dans des Dockers. Ainsi, on s'assure de la pérennité dans le temps des applications en plus d'être certains qu'elles pourront être lancées sur toutes les machines.

- [UD-Viz-Docker](#)
- [UD-Demo-TIGA-Webdoc-ChemistryValley-Docker](#)
- [UD-Demo-vcity-py3dtilers-lyon-Docker](#)
- [UD-Demo-VCity-UI-driven-data-computation-Lyon-Docker](#)

Chaque docker listé est une application des différents outils développés dans le cadre du projet TIGA.

Conclusion

La veille nous a permis de nous orienter vers ces premiers outils de médiation dans le but de rapprocher les différentes acteurs du territoire de la métropole de Lyon. Ces différents outils ont permis la création de nouvelles représentations numériques 3D. Grâce aux améliorations apportées sur cette visualisation nous avons cherché à améliorer la compréhension du citoyen de son territoire. A l'aide de py3dtilers et de l'intégration de document dans un visualisation 3D nous avons développé un web-documentaire pour découvrir le couloir de la vallée de la chimie et des industries qui s'y trouvent. Le tout dans une approche réproductible et open source afin que ces socles techniques puissent être réutilisable et sur différentes thématiques. Les prochaines étapes de ces outils de médiation sont d'être mis à l'épreuve face au citoyen et industriel à travers des ateliers de médiations. Une prochaine étape est d'améliorer la représentation numérique 3D en apportant une maquette tangible d'un quartier et d'y projeter de la donnée sur celle-ci. Nous pourrions animer des ateliers autour de cette maquette augmentée afin d'avoir des retours sur l'efficacité de ces outils de médiation pour répondre au mieux à l'objectif du projet TIGA sur la reconnexion industrie, territoire et citoyen.