## Livrable 2022 : Développement des outils pour la médiation industrielle

### Note aux lecteurs

Toutes les [productions](https://github.com/VCityTeam/TIGA/edit/master/doc/livrable2022.md) sont mises à disposition en libre accès. Le code a été libéré. Il est diffusé en Open Sources [LGPL](https://www.gnu.org/licenses/lgpl-3.0.fr.html) sur ces mêmes pages. Ce livrable est aussi proposé dans une version pdf contenant l’ensemble des informations. La première partie du document contient cette page décrivant le livrable 2022. Les contenus décrivant les composants les plus pertinents sont proposés en annexe du document pdf, mais aussi disponible sur [github](https://github.com/VCityTeam/).

### Introduction

La première étape de ce travail a consisté à proposer une veille des outils de médiation. Dans le [premier livrable](https://github.com/VCityTeam/TIGA/blob/master/doc/livrable2020.md), des expérimentations inspirantes pour le projet TIGA ont été listées et analysées. Ce deuxième livrable est dédié à des propositions d’éléments de médiations liant numérique, territoire et jeu. Ces dispositifs sont pensés afin de favoriser l’interaction entre acteurs. Ils sont aussi conçus comme un ensemble de briques composables et réutilisables pour d’autres cas d’utilisations ou d’autres territoires.

Proposer de nouveaux outils innovants ne peut se faire que par une recherche “tirée par l’aval”; il s’agit de définir avec les partenaires un ensemble de besoins méthodologiques, scientifiques et techniques qui permettent de répondre à des cas d’utilisations présents dans TIGA. Les propositions faites ci-dessous permettent de nourrir en particulier deux besoins. Le premier besoin est de pouvoir mieux appréhender ce qui se passe dans une zone industrielle, en faisant ressortir la ville productive, mais aussi la vie des usagers du même territtoire. Ce travail a été mené avec [la mission vallée de la chimie](https://lyonvalleedelachimie.fr/la-vallee/la-mission/), le centre de formation [Interfora AFAIP](https://www.interfora-ifaip.fr/) et [TUBA : Tube à expérimentation urbaine Lyon](https://www.tuba-lyon.com/). Le deuxième besoin porte sur la nécessaire mobilisation de données (en particulier 3D) dans un ensemble d’outils que nous mettons à disposition de la métropole de Lyon, en particulier de leur [Laboratoire des Usages](https://www.erasme.org/-UrbanLab-). Il s’agit de proposer des briques permettant une meilleure appropriation des données urbaines afin d’en faciliter l’utilisation par les usagers. Ces briques logicielles, issues du laboratoire [Liris](https://projet.liris.cnrs.fr/vcity/), sont présentées ci-dessous. Elle sont ensuite illustrées grâce à des cas d’utilisation.

Pour répondre à ce besoin d’une meilleure compréhension du secteur industriel et du territoire par le citoyen, nous nous sommes donc orientés vers des représentations 3D numériques de la ville. La visualisation 3D d’un environnement nous permet de mieux l’appréhender et ainsi mieux discerner son organisation, ses activités, etc. Cette représentation 3D de la ville peut être améliorée en allant au-delà de la géométrie des bâtiments. Nous voulons apporter plus de couleur et de réalisme à cette représentation pour rendre l’utilisateur plus à l’aise dans sa déambulation. De plus, afin de contextualiser un quartier ou un immeuble, nous cherchons à le lier à des connaissances, en particulier des contenus multi-médias (photo, vidéo, texte, etc.) et apporter une nouvelle dimension la 3D, allant au délà d’une simple déambulation dans l’espace 3D. Pour ce faire, nous avons conçus différents outils utiles pour la conception de jumeaux numériques avec comme point d’entrée des données urbaines. Dans un premier temps, nous développerons les socles techniques qui ont permis la construction de représentations 3D tel que la géométrie des bâtiments. Puis dans un second temps, nous détaillerons les démonstrations produites grâce aux outils développés et qui ont pu être mis en pratique. Ces outils ont été pensés de manière reproductible avec une documentation en continue afin que ceux-ci puissent être utilisables de manière autonome et également sous différentes thématiques. Le type de Licence [LGPL](https://www.gnu.org/licenses/lgpl-3.0.fr.html) facilite aussi la réutilisation des composants et outils développés.

### Sommaire :

* [Développement](#développement)
  + [Modèles 3D](#modèles-3D)
  + [Py3DTiles](#py3dtiles)
  + [Py3DTilers](#py3dtilers)
  + [UD-Viz](#ud-viz)
    - [Couleurs et textures](#couleurs-et-textures)
    - [Intégration de contenus multi-médias](#intégration-de-contenus-multi-médias)
    - [Intégration de couches de données 2D](#intégration-de-couches-de-données-2D)
  + [Démos UD-Viz](#démos-ud-viz)
    - [Démo Py3DTilers](#démo-py3dtilers)
    - [Démo UI-driven](#démo-ui-driven)
  + [Démo Vallée de la chimie](#démo-vallée-de-la-chimie)
  + [Docker](#docker)
* [Conclusion](#conclusion)

### Développement

#### **Modèles 3D**

L’intéraction 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 de ces objets avec de la donnée sémantique (entendons par sémantique les données additionnelles qui viennent enrichir les données géométriques 3D; il peut s’agir d’attributs ou de façon plus général de corpus documentaires liés). L’approche que nous proposons consiste à n’appuyer au maximum sur des standards issus de [L’isso TC/ 211](https://www.iso.org/fr/committee/54904.html) et de l’[Open Geospatial Consortium](https://www.ogc.org/). Cela facilite l’utilisation de composants déjà existants, mais aussi la réutilisation de nos propres composants. En ce sens, nous avons par exemple fait le choix d’utiliser le standard [3D Tiles](https://github.com/CesiumGS/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 associées 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 (distance par rapport au point de vue) 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 représentant 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 avec cet objectif qu’ont été développés [Py3DTiles](#py3dtiles) et [Py3DTilers](#py3dtilers).

#### **Py3DTiles**

[Py3DTiles](https://github.com/VCityTeam/py3dtiles/tree/Tiler) est une librairie libre permettant de produire des [3D Tiles](#modèles-3D). Elle offre un ensemble de fonctionnalités pour écrire des 3D Tiles depuis de la donnée. Originellement développé par [Oslandia](https://gitlab.com/Oslandia/py3dtiles), et le [Liris](https://projet.liris.cnrs.fr/vcity/), 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 Py3DIiles puissent être manipulés par tous les autres outils suivant aussi le standard.

#### **Py3DTilers**

[Py3DTilers](https://github.com/VCityTeam/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](https://en.wikipedia.org/wiki/Wavefront_.obj_file), [GeoJSON](https://en.wikipedia.org/wiki/GeoJSON), [IFC](https://en.wikipedia.org/wiki/Industry_Foundation_Classes) et [CityGML](https://en.wikipedia.org/wiki/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 standard [CityGML](https://www.ogc.org/standards/citygml) permet de représenter les géométries des bâtiments et du relief. Le standard [IFC: Industry foundation classes](https://www.buildingsmart.org/standards/bsi-standards/industry-foundation-classes/) décrit quant à lui plus en détails l’intérieur des bâtiments.

Py3DTilers se base sur la librairie [Py3DTiles](#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](https://fr.wikipedia.org/wiki/Level_of_detail) 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 des objets dans la scène. Ces fonctionnalités permettent aussi de placer de la donnée non géo-référencée dans un contexte géospatial, 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. Cet outil est maintenant à disposition de la communauté TIGA et permet une utilisation facilitée des modèles 3D représentant les territoires cibles de TIGA.

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](https://conference.unsw.edu.au/en/sdsc-3dgeoinfo)) 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](../Livrables/article_py3dtilers.pdf)). Cet article a été accepté pour une présentation dans la conférence internationale [3DGeoinfo](https://conference.unsw.edu.au/en/sdsc-3dgeoinfo) et sera publié dans le journal international [ISPR Annals](https://www.isprs.org/publications/annals.aspx).

Les 3D Tiles générés avec l’outil Py3DTilers peuvent être [visualisés avec différents logiciels](https://github.com/VCityTeam/UD-SV/blob/master/ImplementationKnowHow/Visualize3DTiles.md). Néanmoins, nous utilisons dans la majorité des cas le visualisateur [UD-Viz](https://projet.liris.cnrs.fr/vcity/tools/ud-viz/), un logiciel libre qui a été en partie développé au cours du projet TIGA.

#### **UD-Viz**

[UD-Viz](https://github.com/VCityTeam/UD-Viz) est une librairie JavaScript permettant de visualiser de la donnée urbaine et d’intéragir avec cette donnée via navigateur Web. UD-Viz se base sur [iTowns](https://github.com/itowns/itowns), développé par l’[IGN](https://www.ign.fr/), le LIRIS étant partenaire de ces développements. iTowns permet de charger et afficher des couches de données géographiques ainsi que des modèles [3D Tiles](#modèles-3D).

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 maintenant 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 une plus grande diversité des styles, puisque ces derniers ne sont plus limités à une couleur unique. De plus, les différentes options de [Py3DTilers](#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.

| texture_refine |
| --- |
| *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 données multi-médias additionnelles. Ces multi-médias peuvent être des images d’archives, des vidéos d’acteurs du territoire ou des photos d’un observatoire photographique(voir par exemple le travail sur l’[Observatoire photographique des paysages de la Vallée de la chimie](https://www.caue69.fr/1/page/10622/Observatoire_photographique_du_paysage_sur_le_territoire_de_la_Vallee_de_la_Chimie) mené par la mission vallée de la chimie. 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 libraire [UD-viz](https://github.com/VCityTeam/UD-Viz). L’intégration est découpée en deux parties :

* [Configuration du fichier JSON](https://github.com/VCityTeam/UD-Demo-TIGA-Webdoc-ChemistryValley/blob/main/doc/PinsDoc.md" \l "configepisodesjson-) : documentation sur la configuration du fichier JSON afin de lier contenus multi-medias et positions géospatiales.
* [Episode visualizer object](https://github.com/VCityTeam/UD-Demo-TIGA-Webdoc-ChemistryValley/blob/main/doc/PinsDoc.md) : programme d’intégration des multi-médias avec comme point d’entrée le fichier de configuration JSON.

| contenue |
| --- |
| *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 web-documentaire du projet “Derrière les fumées” (voir ci-après). 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 multi-média dans une représentation 3D numérique soumis à la même conférence internationale que Py3dTilers, [Smart Data Smart Cities & 3D GeoInfo](https://conference.unsw.edu.au/en/sdsc-3dgeoinfo) ([Article intégration de multimedia dans une représentation 3D numérique](../Livrable/Integrating_multimedia_documents_for_augmented_models_to_a_better_understanding_of_its_territory.pdf)). Cet article a été accepté pour une présentation dans la conférence internationale [3DGeoinfo](https://conference.unsw.edu.au/en/sdsc-3dgeoinfo) et sera publié dans le journal international [ISPR Annals](https://www.isprs.org/publications/annals.aspx).

##### **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](https://www.geolittoral.developpement-durable.gouv.fr/IMG/pdf/note_explicative_wms_wfs_geolittoral.pdf) (Web Feature Service) et [WMS](https://www.geolittoral.developpement-durable.gouv.fr/IMG/pdf/note_explicative_wms_wfs_geolittoral.pdf) (Web Map service) dans la bibliothèque [UD-viz](https://github.com/VCityTeam/UD-Viz). Ce type de données nous permet de représenter différentes géométries dans UD-Viz; cela peut être des polylignes, 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.

| Capture |
| --- |
| *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 [démo vallée de la chimie](#démo-vallée-de-la-chimie) avec un jeu de données 2D disponible sur le site [open data Grand Lyon](https://data.grandlyon.com/). 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](https://github.com/VCityTeam/UD-Viz-Template) est un template d’application permettant de créer rapidement des démonstrations basées sur la librairie [UD-Viz](#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 :

##### **Démo Py3DTilers**

* [Code](https://github.com/VCityTeam/UD-Demo-vcity-py3dtilers-lyon) source de la démonstration ainsi que sa documentation.
* [Docker](https://github.com/VCityTeam/UD-Demo-vcity-py3dtilers-lyon-docker) pour reproduire l’application.
* [Démo en ligne](https://py3dtilers-demo.vcityliris.data.alpha.grandlyon.com/)

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.

| image |
| --- |
| *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 différents niveaux de détail..* |

Les 3D Tiles ont été créés à partir de couches de données publiques issues du site du [Grand Lyon](https://data.grandlyon.com/) et de l’[IGN](https://geoservices.ign.fr/telechargement). 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](https://github.com/VCityTeam/UD-SV/blob/master/ImplementationKnowHow/PostgreSQL_for_cityGML.md). 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](https://github.com/VCityTeam/UD-Reproducibility/blob/master/Computations/3DTiles/Lyon_Relief_Roads_Buildings_Water/Compute_Lyon_3DTiles.md) 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 (position par rapport à la caméra) : plus la caméra 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.

| mouse_refine |
| --- |
| *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](https://github.com/VCityTeam/UD-Demo-VCity-UI-driven-data-computation-Lyon) source de la démonstration ainsi que sa documentation.
* [Docker](https://github.com/VCityTeam/UD-Demo-VCity-UI-driven-data-computation-Lyon-docker) pour reproduire l’application.
* [Démo en ligne](https://ui-driven-data-lyon.vcityliris.data.alpha.grandlyon.com/)

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 à partir de donnée de l’IGN est détaillé dans cette [documentation](https://github.com/VCityTeam/UD-Reproducibility/blob/master/Computations/3DTiles/Lyon_Relief_Roads_Buildings_Water/Roads_from_relief.md).

| roads_drag_and_drop |
| --- |
| *Figure 6 : Démo UD-Viz sur le calcul des hauteurs de routes contenu dans un fichier GeoJSON à l’aide d’un glisser/déposer dans l’application web UD-Viz* |

#### **Démo Vallée de la chimie**

Une expérimentation de ces outils aidant à la représentation numérique d’un territoire et la démo [Vallée de la chimie](https://fr.wikipedia.org/wiki/Vallée_de_la_chimie) qui a été développé dans le cadre d’un web-documentaire en collaboration avec [la missions vallée de la chimie](https://lyonvalleedelachimie.fr/la-vallee/la-mission/), le centre de formation [Interfora AFAIP](https://www.interfora-ifaip.fr/) et [TUBA : Tube à expérimentation urbaine Lyon](https://www.tuba-lyon.com/). Ce projet “Derrière les fumées” a pour but de déconstruire les idées préconçues 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.

* [Code](https://github.com/VCityTeam/UD-Demo-TIGA-Webdoc-ChemistryValley) source de la démonstration ainsi que sa documentation.
* [Docker](https://github.com/VCityTeam/UD-Demo-TIGA-Webdoc-ChemistryValley-docker) pour reproduire l’application.
* [Démo en ligne](https://www.derrierelesfumees.com/_Contenusdlf/Carte/index.html)

| chemistryvalley |
| --- |
| *Figure 7 : Capture du web-documentaire “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 Grand 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](https://dataset-dl.liris.cnrs.fr/three-d-tiles-lyon-metropolis/Demo/UD-Demo-vcity-py3dtilers-lyon/) : 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](https://umap.openstreetmap.fr/fr/map/vallee-de-la-chimie-observatoire-photographique_233823" \l "18/45.68050/4.86110) 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](https://numelyo.bm-lyon.fr/BML:BML_01ICO001015c33b77d0036c?&collection_pid=BML:BML_01ICO00101&luckyStrike=1&query%5B%5D=isubjectgeographic:"Feyzin (Rhône)"&hitPageSize=1&hitTotal=62&hitStart=24)).

#### **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 utilisée sur n’importe quelle machine, sans souci d’installation et sans erreur de version des logiciels. L’utilisation de Docker permet d’éviter qu’une application fonctionnelle ne devienne inutilisable après quelque temps à cause de mise à jour 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](https://github.com/VCityTeam/UD-Viz-docker)
* [UD-Demo-TIGA-Webdoc-ChemistryValley-Docker](https://github.com/VCityTeam/UD-Demo-TIGA-Webdoc-ChemistryValley-docker)
* [UD-Demo-vcity-py3dtilers-lyon-Docker](https://github.com/VCityTeam/UD-Demo-vcity-py3dtilers-lyon-docker)
* [UD-Demo-VCity-UI-driven-data-computation-Lyon-Docker](https://github.com/VCityTeam/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

Depuis le début du projet TIGA, l’action 14 “THINK & DO TANK Sciences, Sociétés et Industrie” permet de favoriser les échanges entre partenaires. Après un [travail de veille](https://github.com/VCityTeam/TIGA/blob/master/doc/livrable2020.md) qui a donné lieu au premier livrable en tout début de projet TIGA, et une phase d’échange au sein de cette action, il s’est agit de proposer un ensemble de composants permettant le développement de nouvelles modalités de médiation pour aider à la reconnexion entre Industrie, citoyen et territoire. La veille nous a permis de nous orienter vers ces premiers outils de médiation dans le but de rapprocher les différents acteurs du territoire de la métropole de Lyon.

Dans ce deuxième livrable, nous avons cherché à voir comment des représentations 3D pouvaient venir en interface entre le territoires et les usagers. Les différents outils ont permis la création de nouvelles représentations numériques 3D en y apportant une nouvelle dimension grâce aux multi-médias et ainsi de les éprouver sur des problématiques en lien avec l’industrie. Chaque application est développée grâce à un ensemble de composants atomiques mis à disposition des acteurs TIGA, mais aussi plus généralement d’une communauté plus large afin d’en assurer une meilleure dissémination. Au delà du code, il s’est agit de proposer une documentation liée, ainsi que des dockers afin que tout le monde puisse utiliser les composants sur son propre environnement de travail et répondre au besoin de reproductibilité. Les composants sont conçus de façon atomique afin de pouvoir les composer en fonction des besoins et assurer une meilleur réutilisabilité, mais aussi réplicabilité.

Ces composants ont été mis en place afin de proposer une nouvelle modalité d’accès aux contenus via le territoire dans le cadre du web doc “Derrière les fumées” mis en place en collaboration avec [la missions vallée de la chimie](https://lyonvalleedelachimie.fr/la-vallee/la-mission/), le centre de formation [Interfora AFAIP](https://www.interfora-ifaip.fr/) et [TUBA : Tube à expérimentation urbaine Lyon](https://www.tuba-lyon.com/).

Une autre application de ces composants est faite sur les “maquettes tangibles”, élément d’hybridation entre maquette physique du territoire et contenu numérique. Elle sera décrite dans un prochain livrable. Grâce à l’intégration d’images d’archives dans un jumeau numérique de ville nous avons abordé l’évolution temporelle de la ville. Nous voulons aller plus loin et développer plus en profondeur ce côté évolutif pour mieux comprendre comment un territoire est devenu ce qu’il est actuellement. Ce projet Morphogenèse se veut focalisé sur l’évolution du travail de 1950 à aujourd’hui et est le prochain objectif du LIRIS dans le contexte du projet TIGA, en collaboration avec le laboratoire [Environnement Ville et Société](https://umr5600.cnrs.fr/fr/accueil/). Il viendra aussi compléter le prochain livrable que le LIRIS aura à produire dans les prochains mois dans le cadre du projet TIGA.