

# Comment implémenter l'automatisation au sein de votre organisation

**David LICONA**

BIM Implementation Consultant | [@David Licona](#)





# A propos du présentateur

David LICONA

- Ingénieur en Génie Civil
- 5+ années d'expérience dans la phase de conception, réalisation et opération des infrastructures linéaires
- Aujourd'hui, consultant implémentation pour Autodesk
- 1ère participation en AU

# Objectifs d'apprentissage



## IDENTIFIER LES OPPORTUNITÉS D'AUTOMATISATION

- Pour quoi l'automatisation ?
- Identification des cas d'usage
- Classement par priorité



## DÉVELOPPER DES SOLUTIONS INDUSTRIALISABLES

- Dynamo Player
- Documentation du fonctionnement
- Modularisation des solutions



## PENSER ANALYTIQUEMENT

- Décrire la problématique
- Créer un « POC » rapidement
- Concepts clés de Dynamo



## IMPLÉMENTER L'AUTOMATISATION

- « Best practices »
- Communication des succès
- Ressources pour réussir





Photo par [Frame Harirak](#) - [Unsplash](#)

## S'attendre à :

- Conseils, bonnes pratiques, outils, méthodes pour réussir à implémenter l'automatisation
- Cas d'étude : modélisation des terrassements comme des objets (BIM 4D, BIM 5D)
- Focalisé sur Dynamo pour Civil 3D



Photo par [Lindsay Henwood](#) - [Unsplash](#)

## Et pas à ...

- Une démonstration pas à pas construction des graphes Dynamo
- Une formation de Dynamo pour Civil 3D



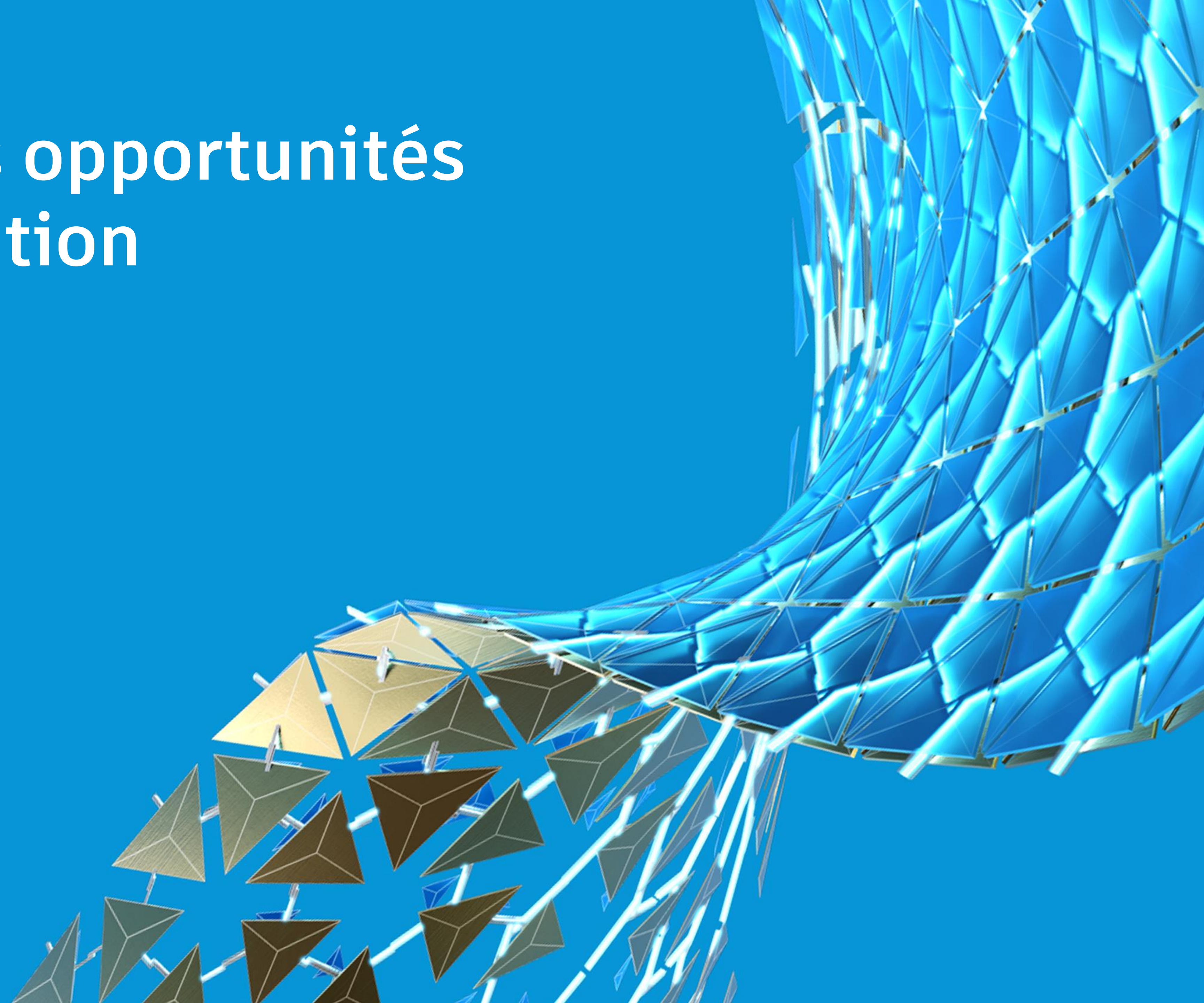
# Dynamo Foundation

- **Offre d'Autodesk Consulting**
  - résoudre les cas d'usage les plus complexes
  - améliorer la collaboration
  - créer une communauté autour de Dynamo à travers les bureaux et les disciplines





# Identifier les opportunités d'automatisation





# C'est quoi l'automatisation

- Réduire l'intervention humaine pour réaliser des tâches
- Ce n'est pas le futur, c'est le **présent**





# Programmation Visuelle

- Un moyen pour les agents d'étude
  - Accessibilité à tous pour créer l'automatisation
  - Libération de la créativité



Photo par [Daniel Tuttle](#) - [Unsplash](#)

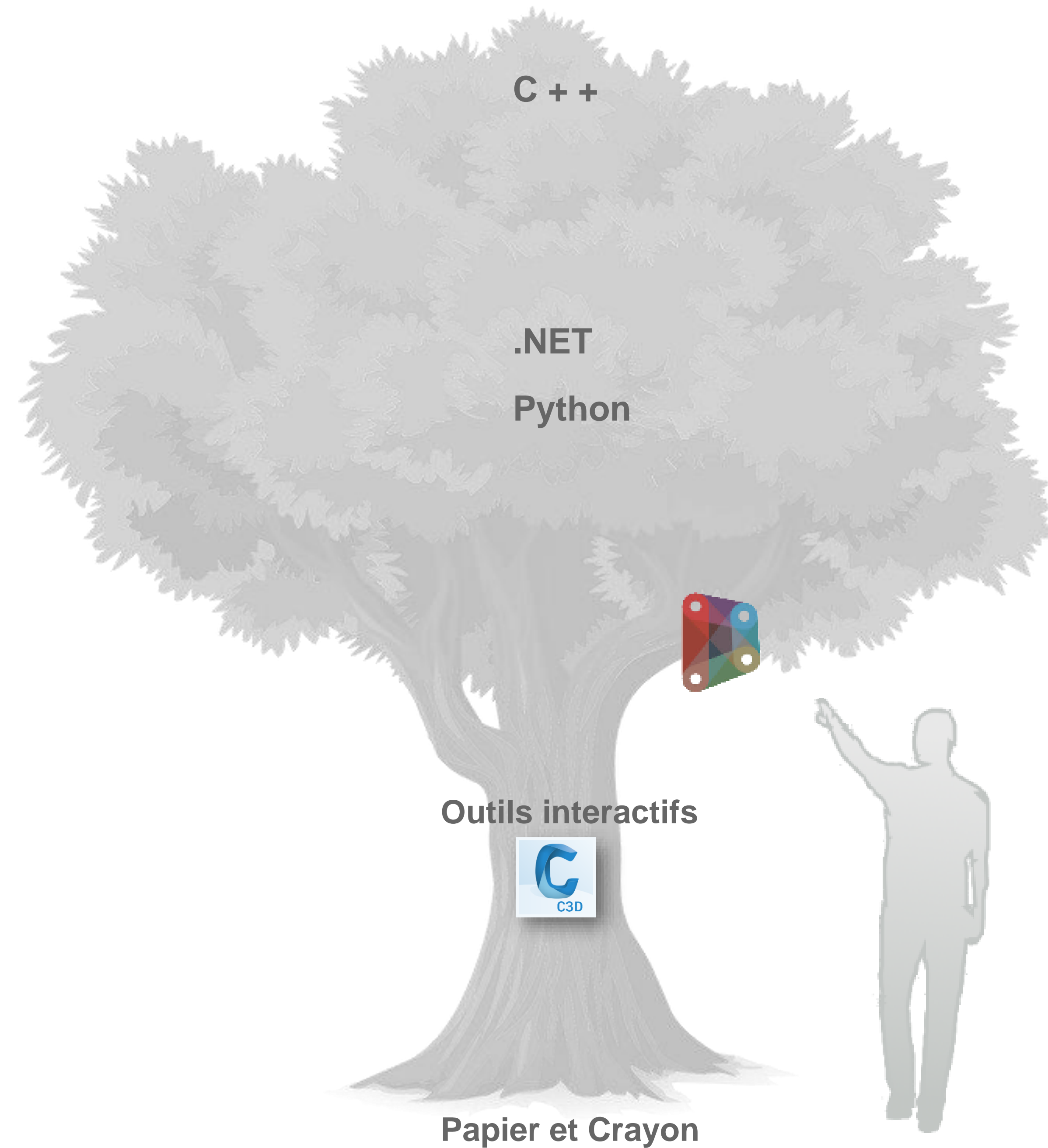


Photo par [Vitae London](#) - [Unsplash](#)



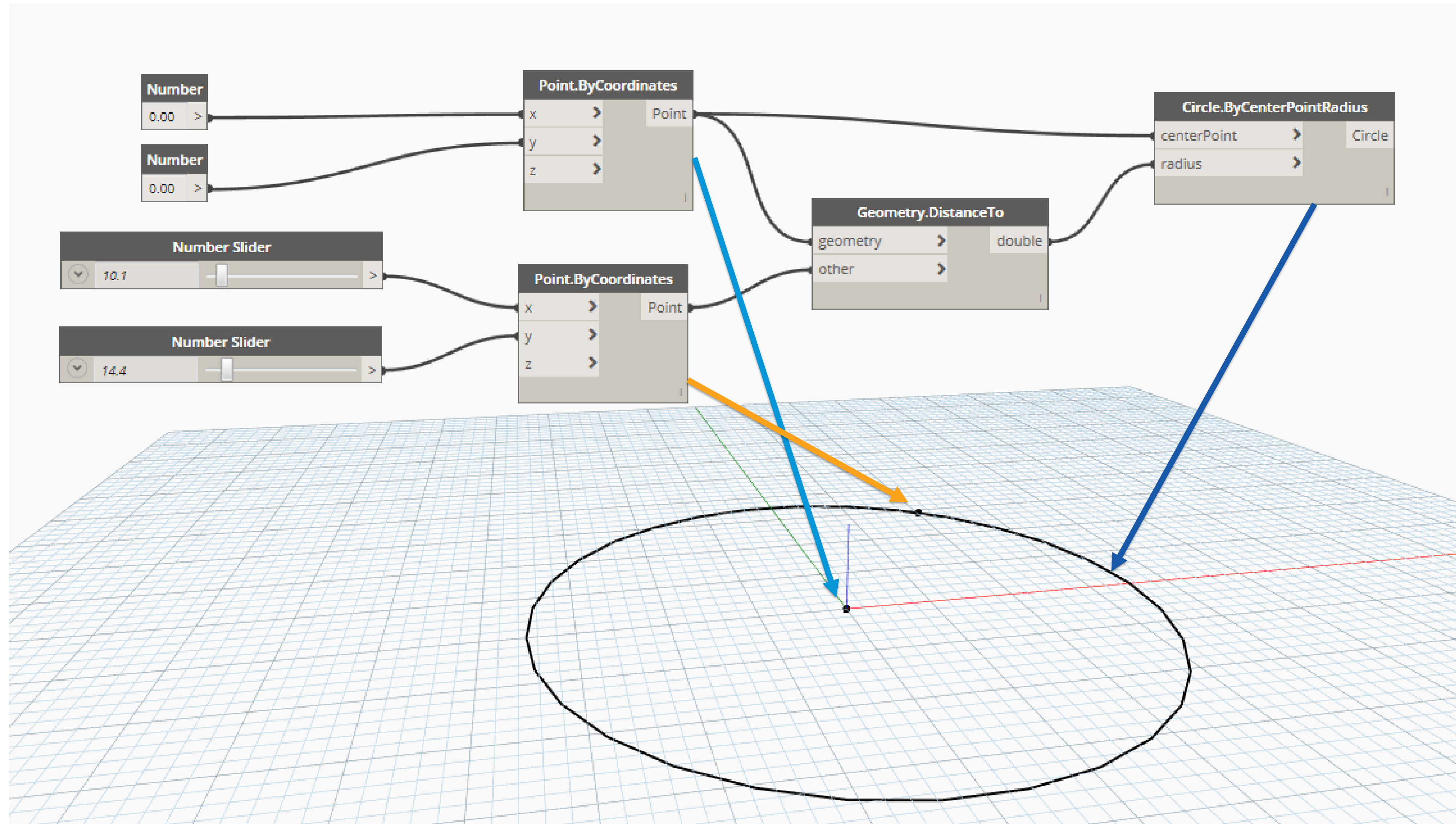
# Dynamo pour Civil 3D

- Dynamo est un outil de programmation visuelle
- Intégré dans Civil 3D





# Programmation Visuelle





# Pour faire quoi ?

- Des cas **PLUS** complexes, modèles **PLUS** détaillés
- **MIEUX** : éviter les erreurs humaines, **MEILLEURE** qualité
- Avec **MOINS** :
  - ↓ ressources
  - ↓ temps
  - ↓ reprises
  - ↓ travail répétitif





# Pour faire quoi ?

- Automatiser des tâches répétitives
- Créer des workflows complexes, connectant plusieurs sources des données
- Améliorer l'interopérabilité des outils
- Explorer des options de conception rapidement
- Accéder aux données du modèle, souvent obscures
- Créer et analyser rapidement des formes géométriques complexes





# Pour faire quoi ?

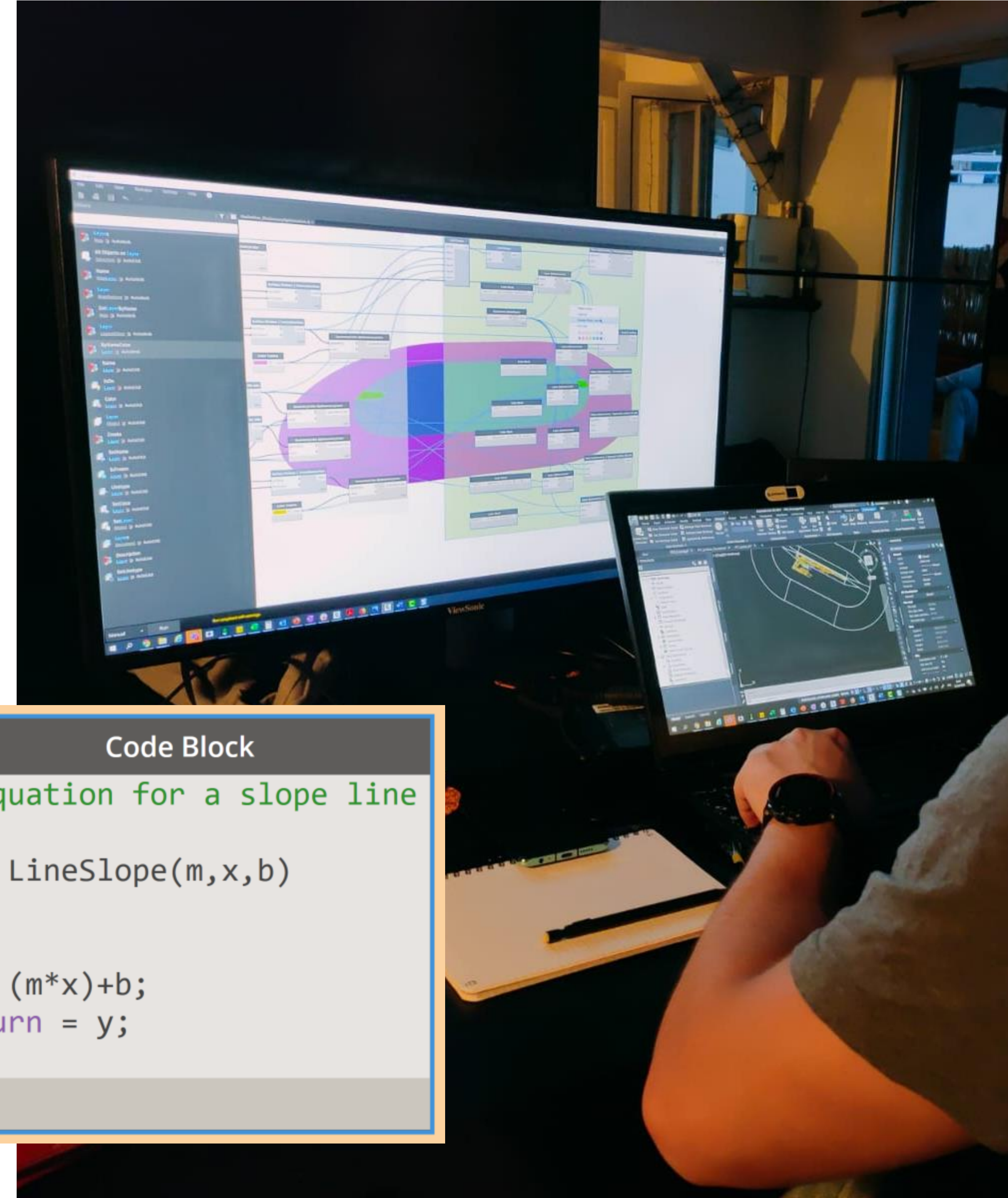
- Automatiser des tâches répétitives
- Créer des workflows complexes, connectant plusieurs sources des données
- Améliorer l'interopérabilité entre les outils
- Explorer des options de conception rapidement
- Accéder aux données du model, souvent obscures
- **Créer et analyser rapidement des formes géométriques complexes**





# Pour faire quoi ?

- Automatiser des tâches répétitives
- Créer des workflows complexes, connectant plusieurs sources des données
- Améliorer l'interopérabilité entre les outils
- Explorer des options de conception rapidement
- Accéder aux données du model, souvent obscures
- **Créer et analyser rapidement des formes géométriques complexes**



## Code Block

```
//Equation for a slope line
```

```
def LineSlope(m,x,b)
```

```
{
```

```
y = (m*x)+b;
```

```
return y;
```

```
};
```



# Valeur pour tout type d'organisation

- **Petites**
  - Concurrencer les plus grands grâce à l'automatisation
- **Moyennes**
  - Réduire des coûts en interne pour réinvestir sur la recherche et innovation
- **Grandes**
  - Déployer les meilleures pratiques de manière plus efficace



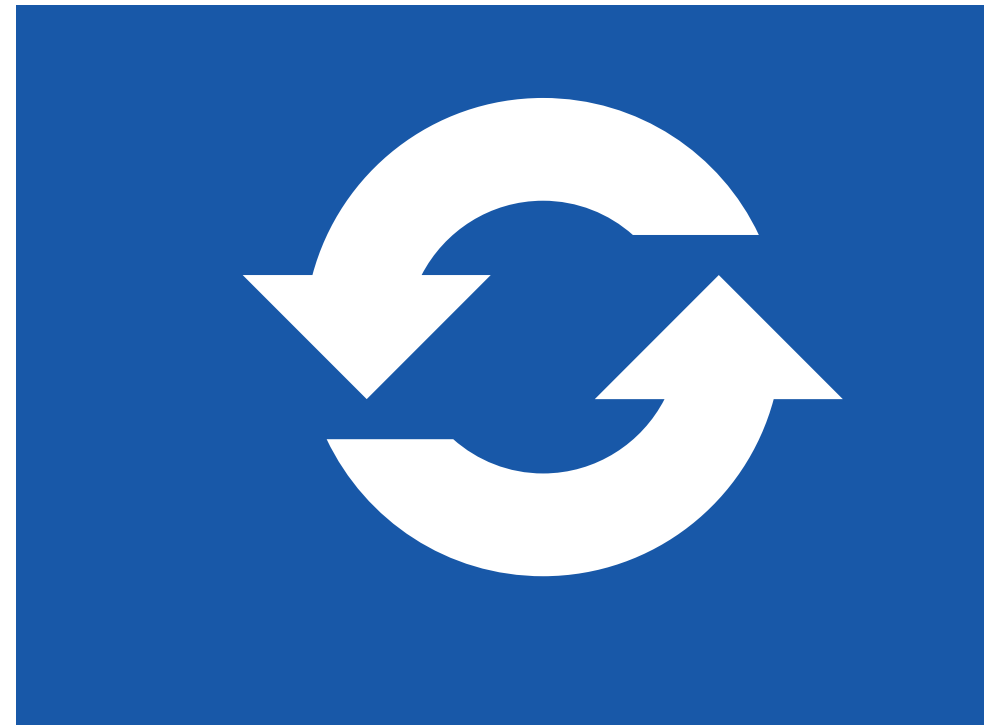


# Pour identifier les opportunités

Est-ce que c'est une  
tâche répétitive ?



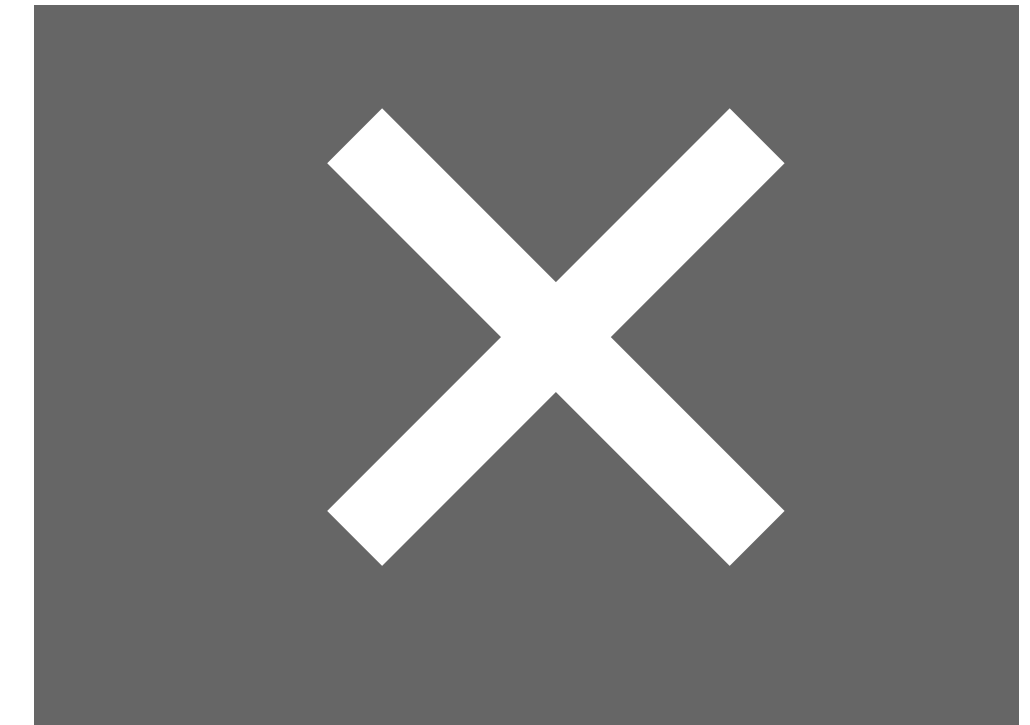
Est-ce que c'est une  
tâche systématique?



Est-ce que cela prend  
trop de temps ?



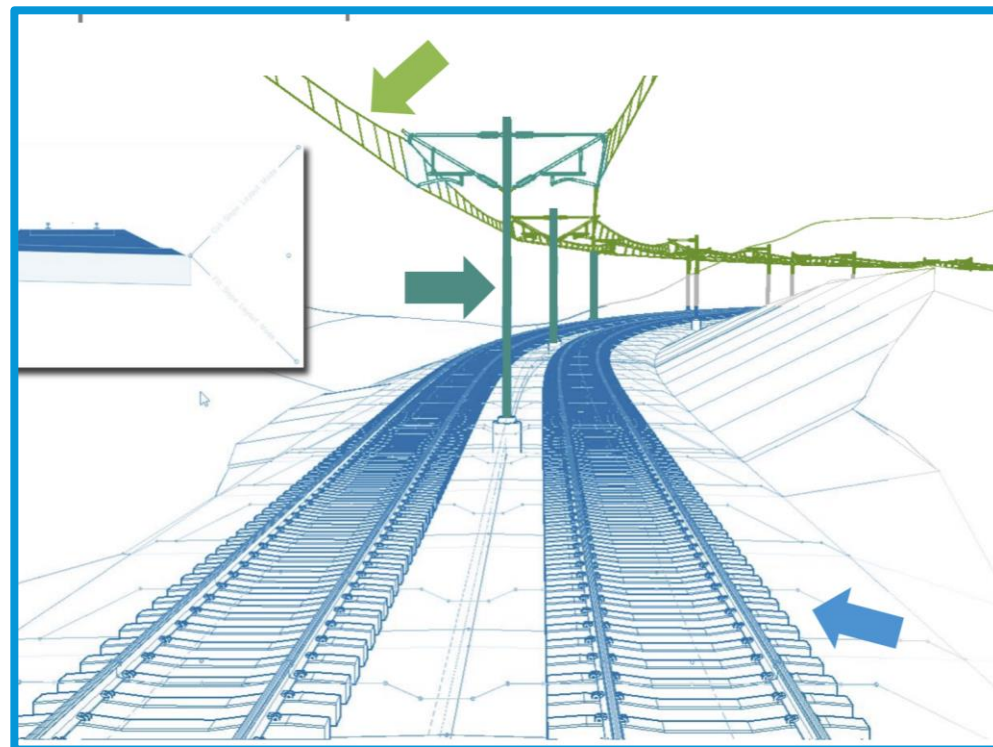
Est-ce que c'est sensible à  
l'erreur humaine ?





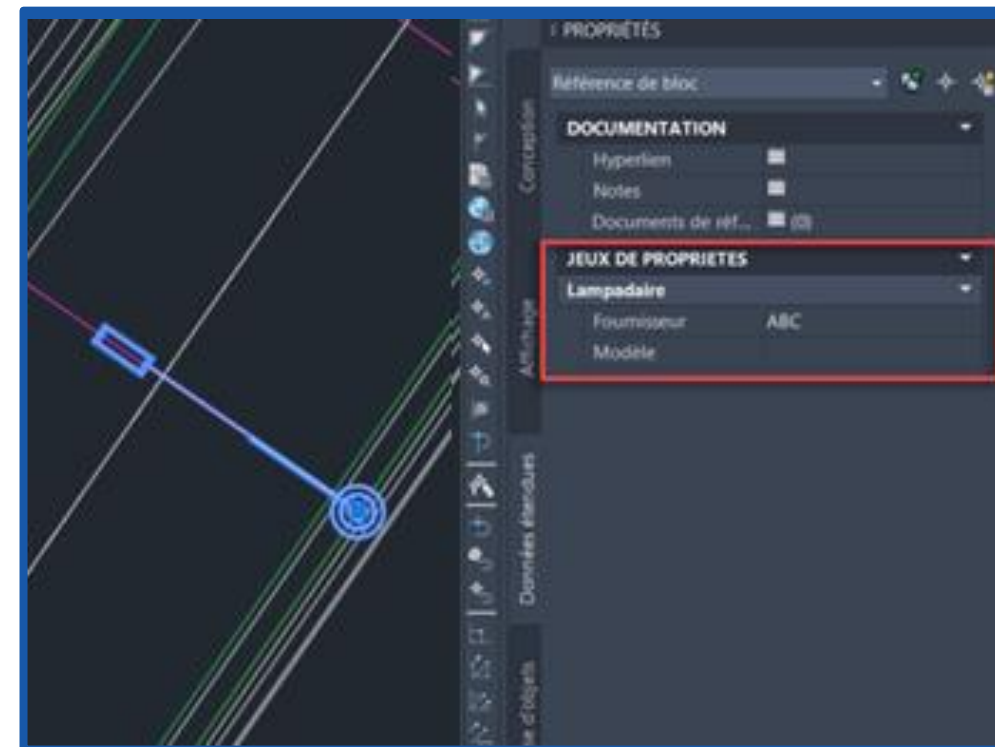
# Cas d'usage métier

Est-ce que c'est une tâche répétitive ?



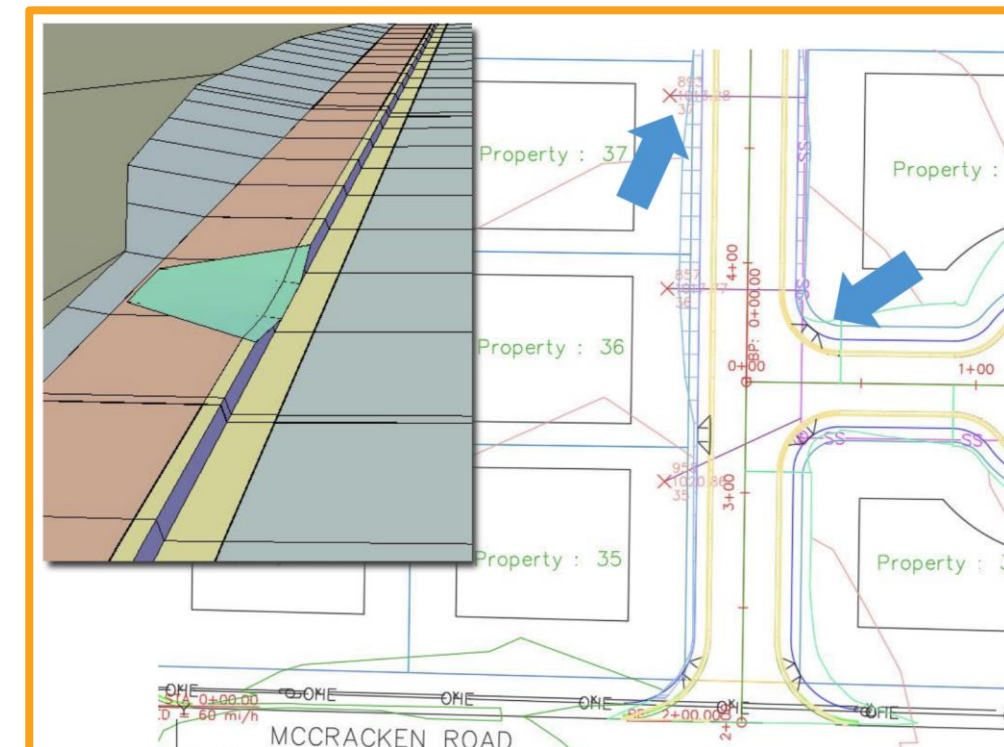
Placement des blocs le long d'un projet  
(traverses, lampadaires poteaux caténaires)

Est-ce que c'est une tâche systématique?



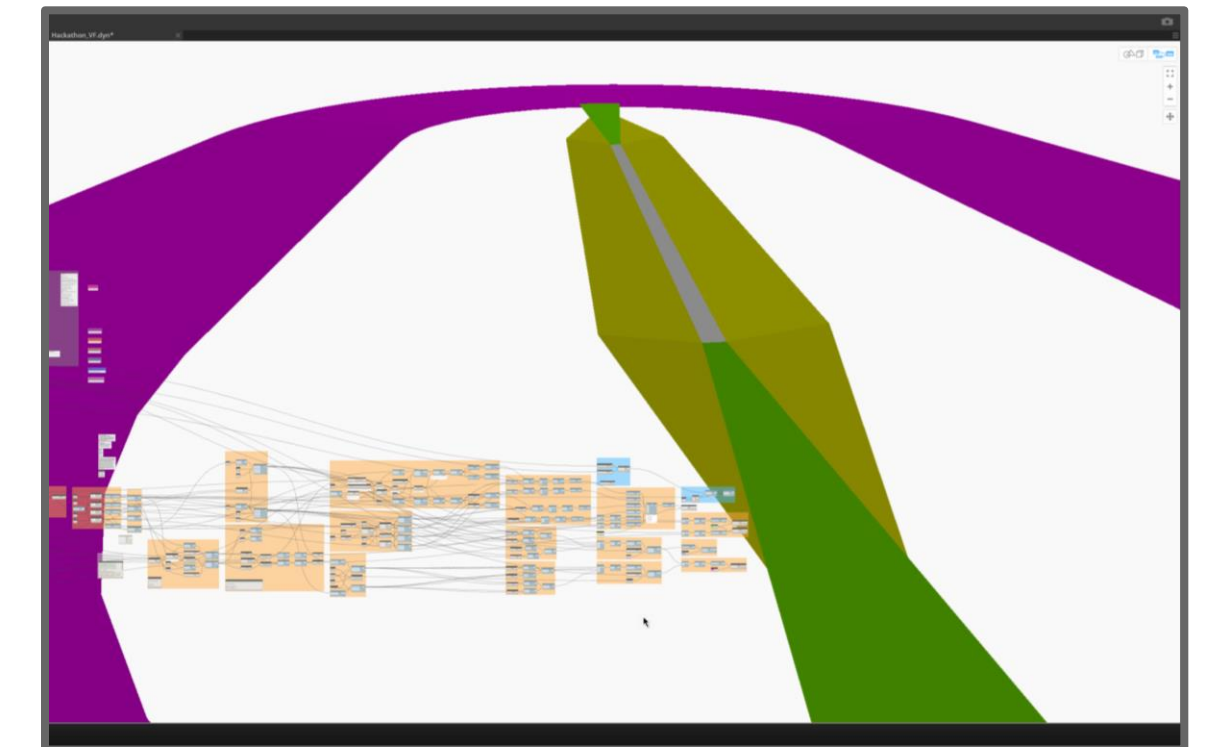
Application des jeux de propriétés

Est-ce que cela prend trop de temps ?



Modélisation des rampes pour des trottoirs

Est-ce que c'est sensible à l'erreur humaine ?



Géométrie complexe

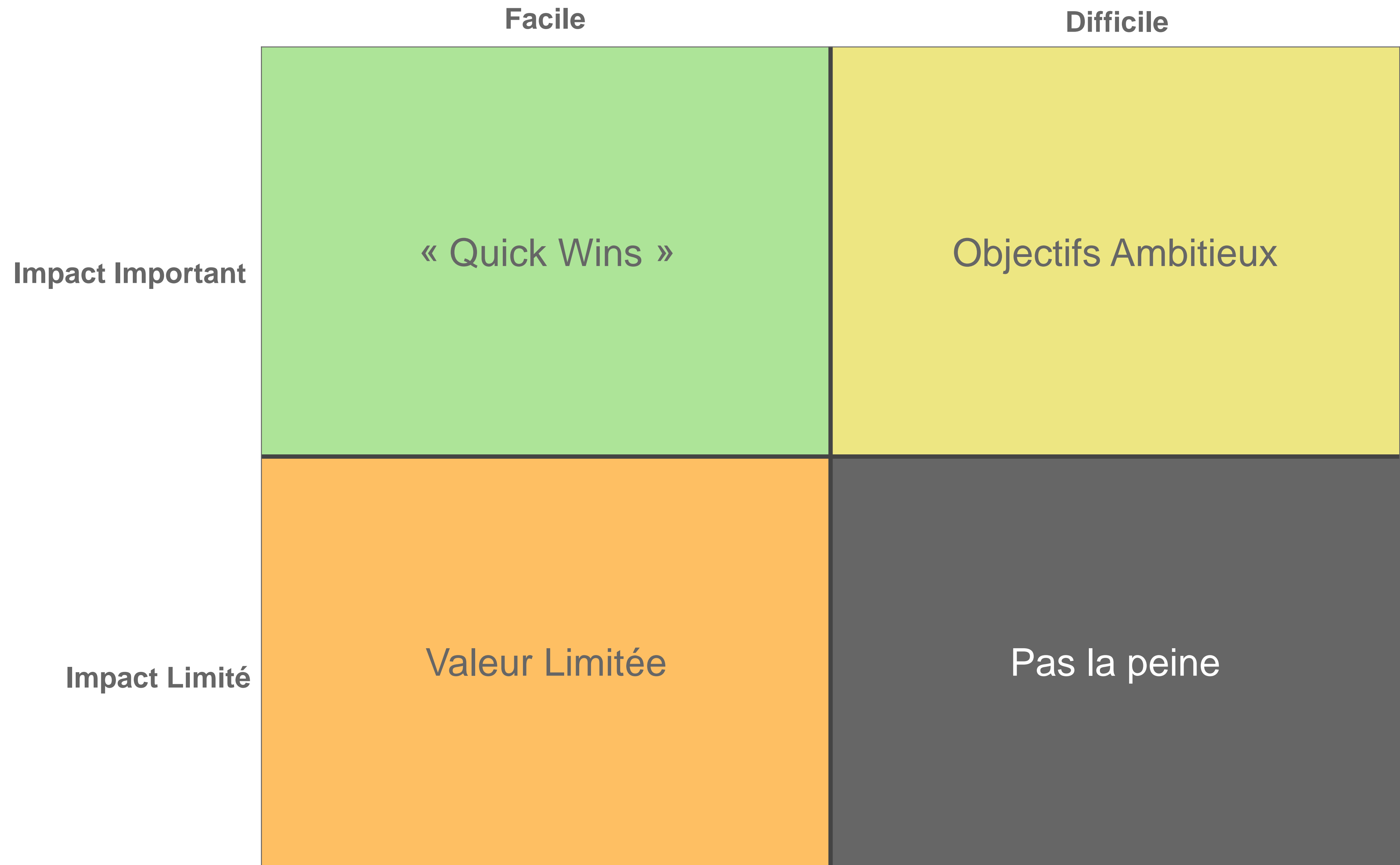


# Prioriser vos idées

- Se concentrer sur les « quick wins »
- Impact visible avec « minimum » d'effort







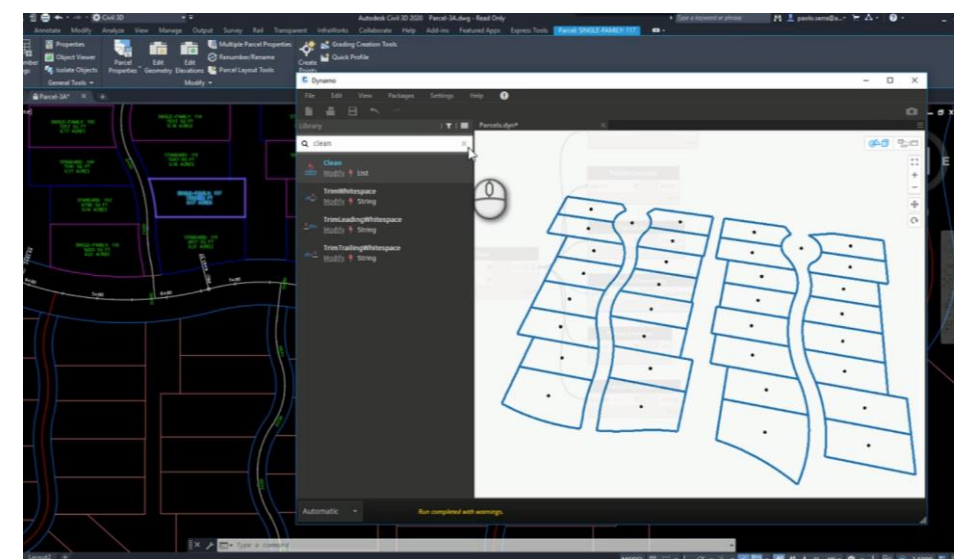


# La base pour aller plus loin

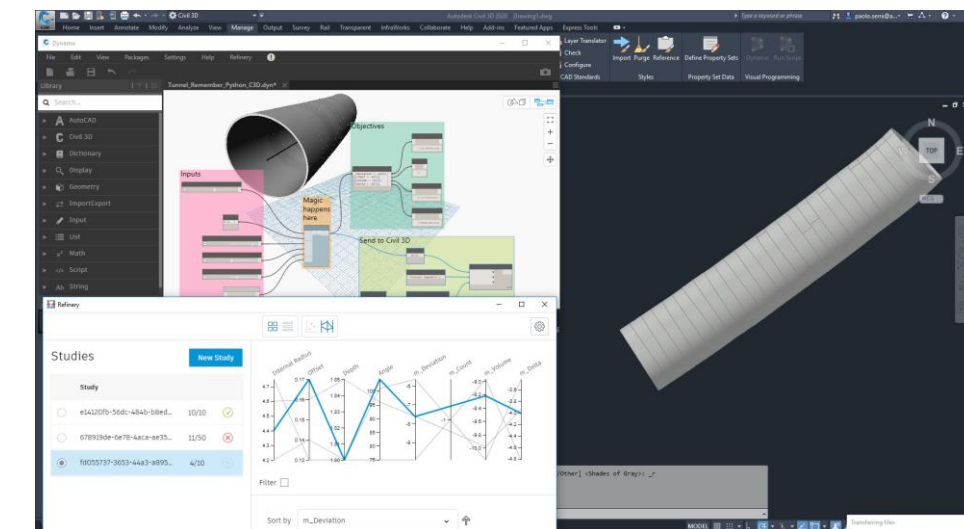
## CONCEPTION PARAMÉTRIQUE



## PROGRAMMATION VISUELLE



## GENERATIVE DESIGN



## MACHINE LEARNING

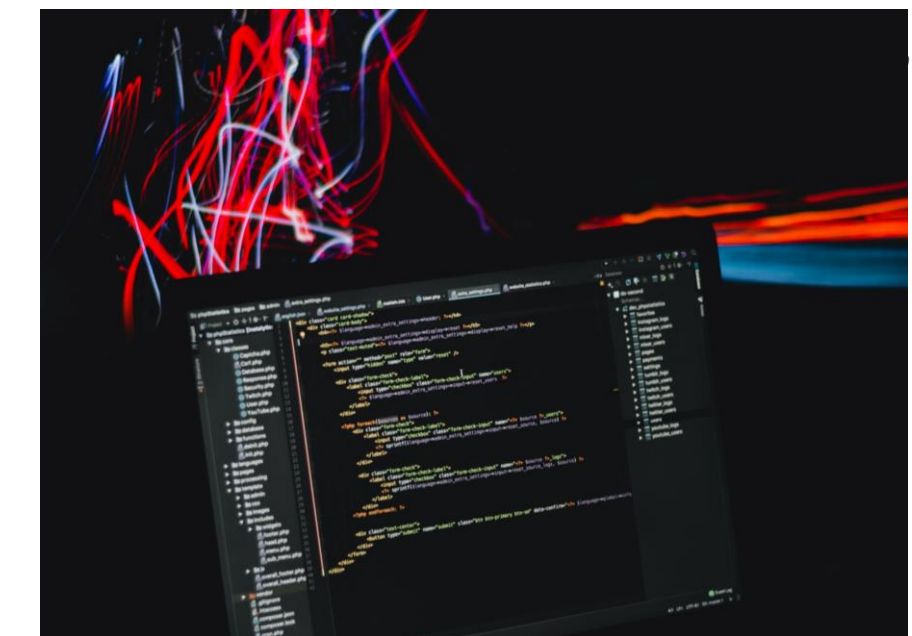


Photo par [AltumCode](#) - [Unsplash](#)



# Objectifs d'apprentissage



## IDENTIFIER LES OPPORTUNITÉS D'AUTOMATISATION

- ✓ L'automatisation pour faire plus, mieux et avec moins de ressources
- ✓ Cas d'usage : tâches répétitives, tâches avec peu de valeur ajoutée, processus itératifs, etc.
- ✓ Se focaliser sur les « quick wins »



## DÉVELOPPER DES SOLUTIONS INDUSTRIALISABLES



## PENSER ANALYTIQUEMENT

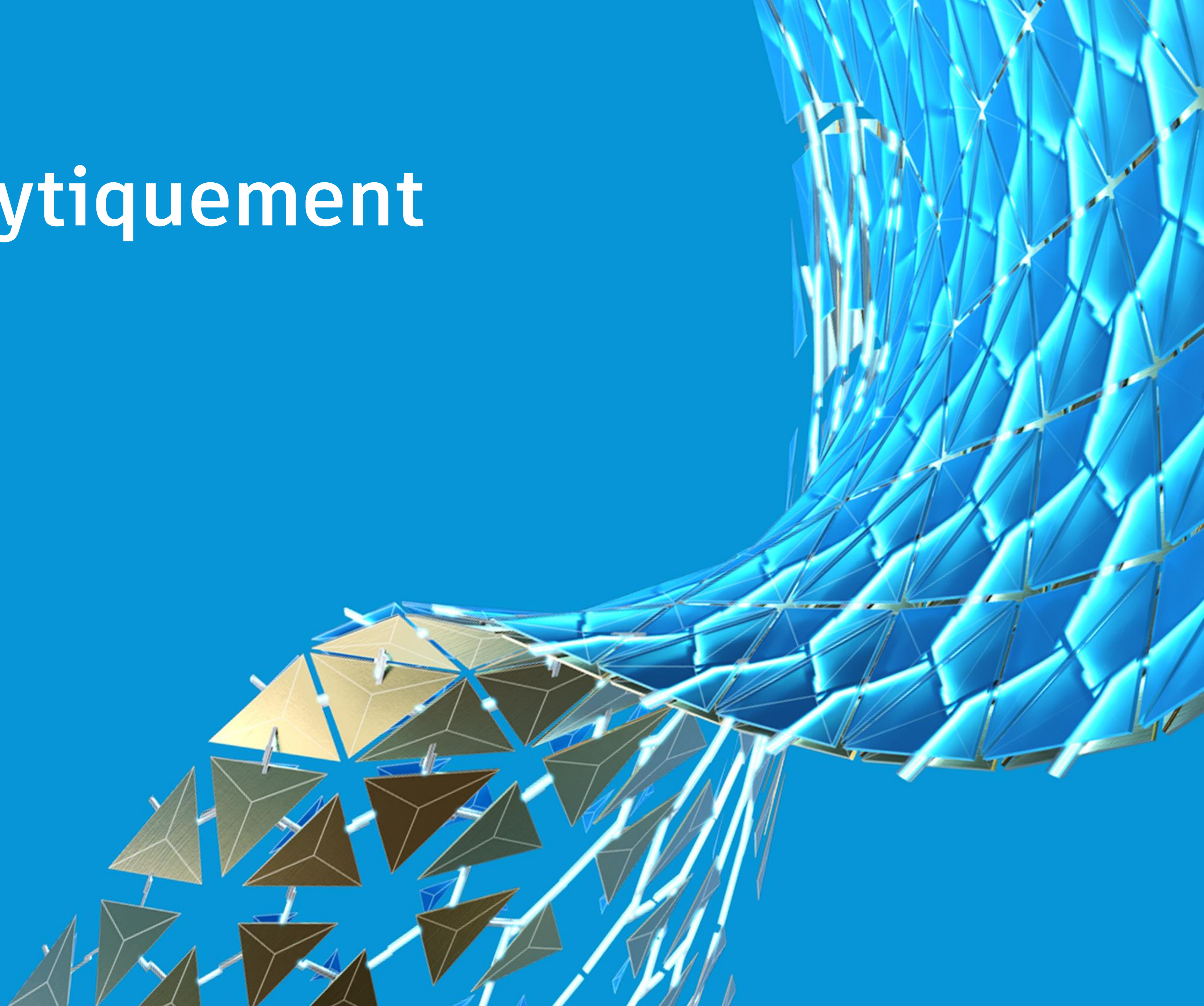
- Décrire la problématique
- Créer un « POC » rapidement
- Concepts clés de Dynamo



## METTRE EN PLACE L'AUTOMATISATION



# Penser analytiquement



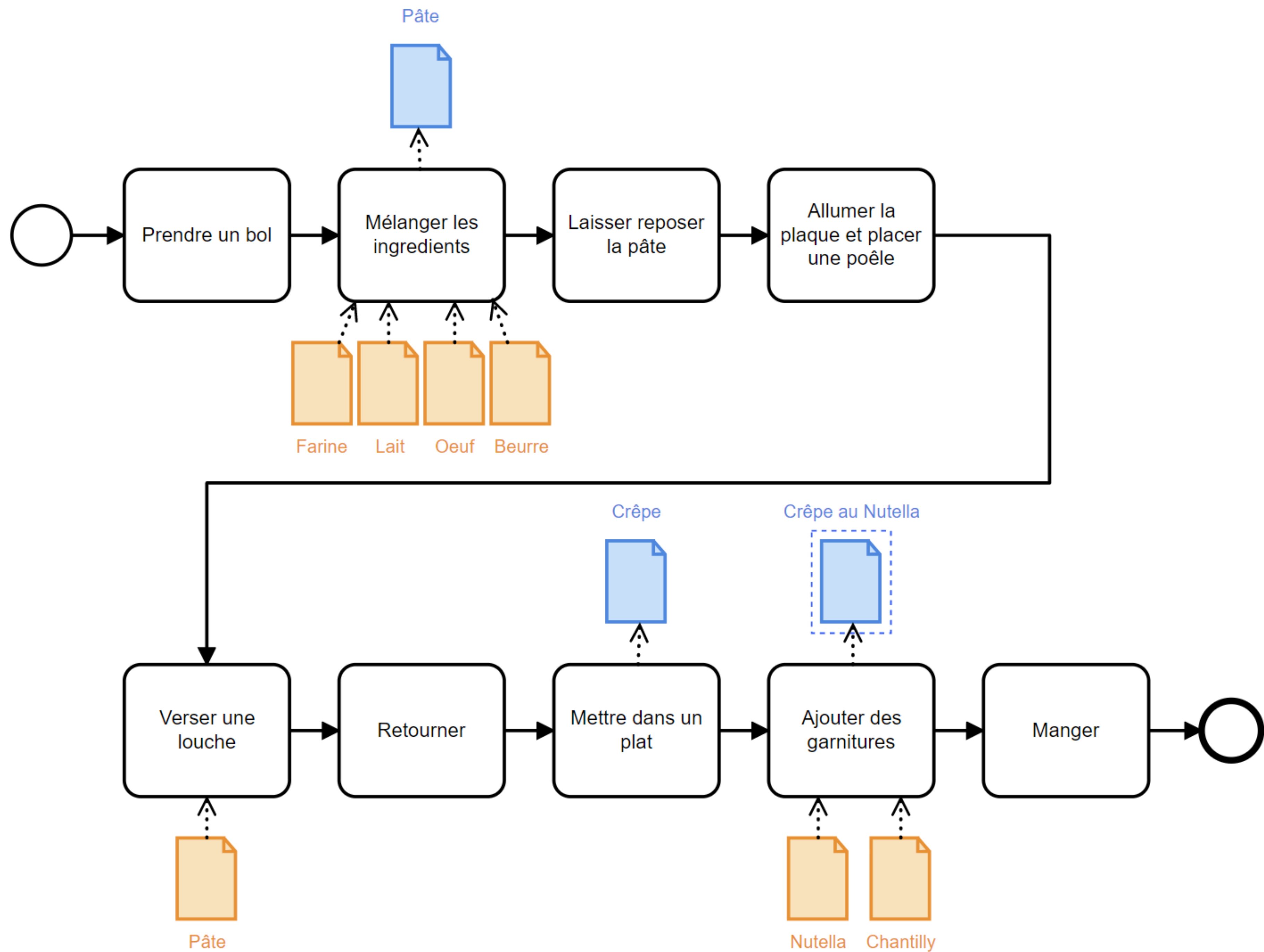


# Ecrivez-le

- Être capable de résumer la problématique
  - Mettre les tâches dans une séquence
  - Visualiser le flux de travail / processus
- BPMN
  - Outil : cawemo

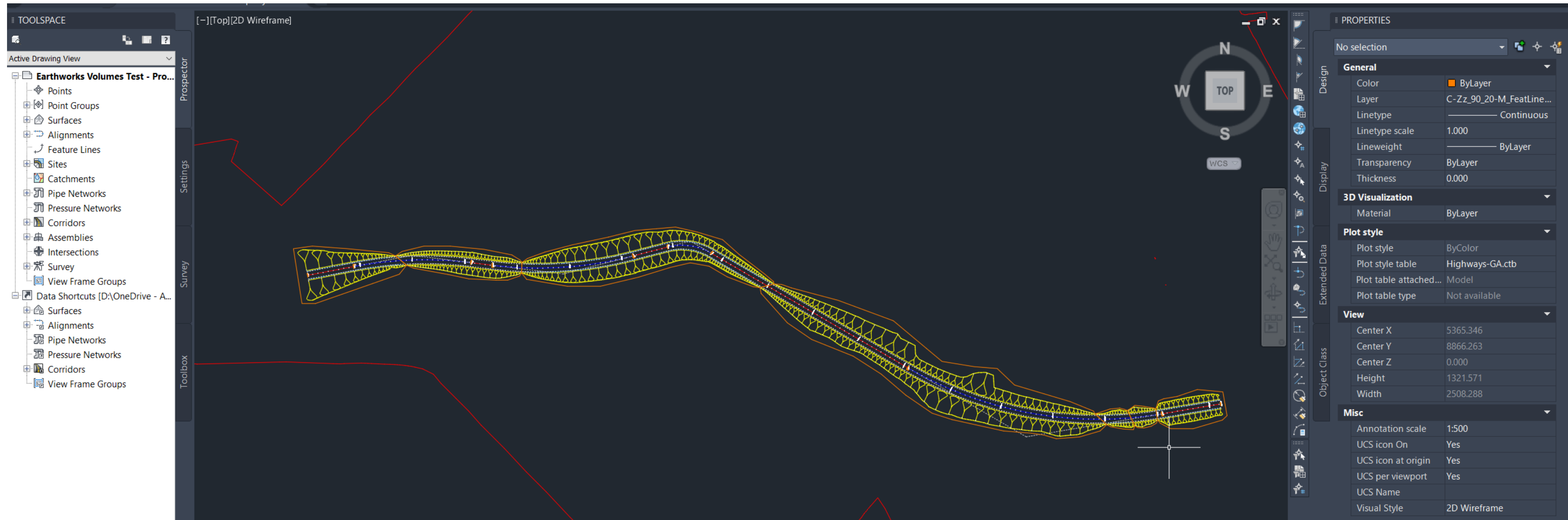








# Cas d'étude : Modélisation des terrassements



Un script Dynamo pour :

- Modéliser des solides 3D représentant les terrassements pour n'importe quel projet 3D
- Diviser les solides
  - Déblai/Remblai
  - Par la ligne de base
  - Par corps de terrassement
- Ajouter des jeux de propriétés et des propriétés :
  - Hauteur maximum
  - Hauteur moyenne



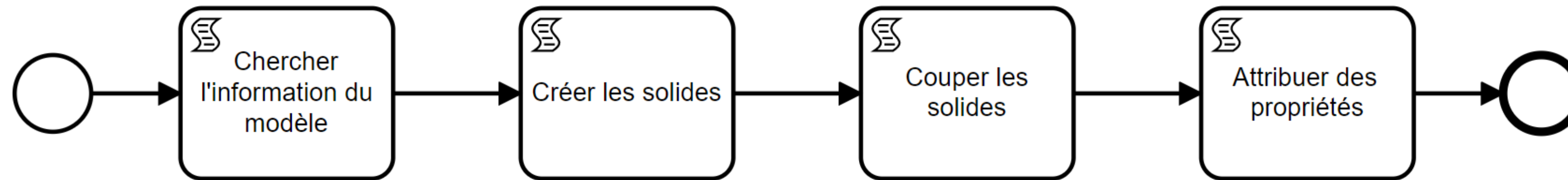
# Modélisation des terrassements

- BIM4D & 5D
- 4 jours pour tout réaliser



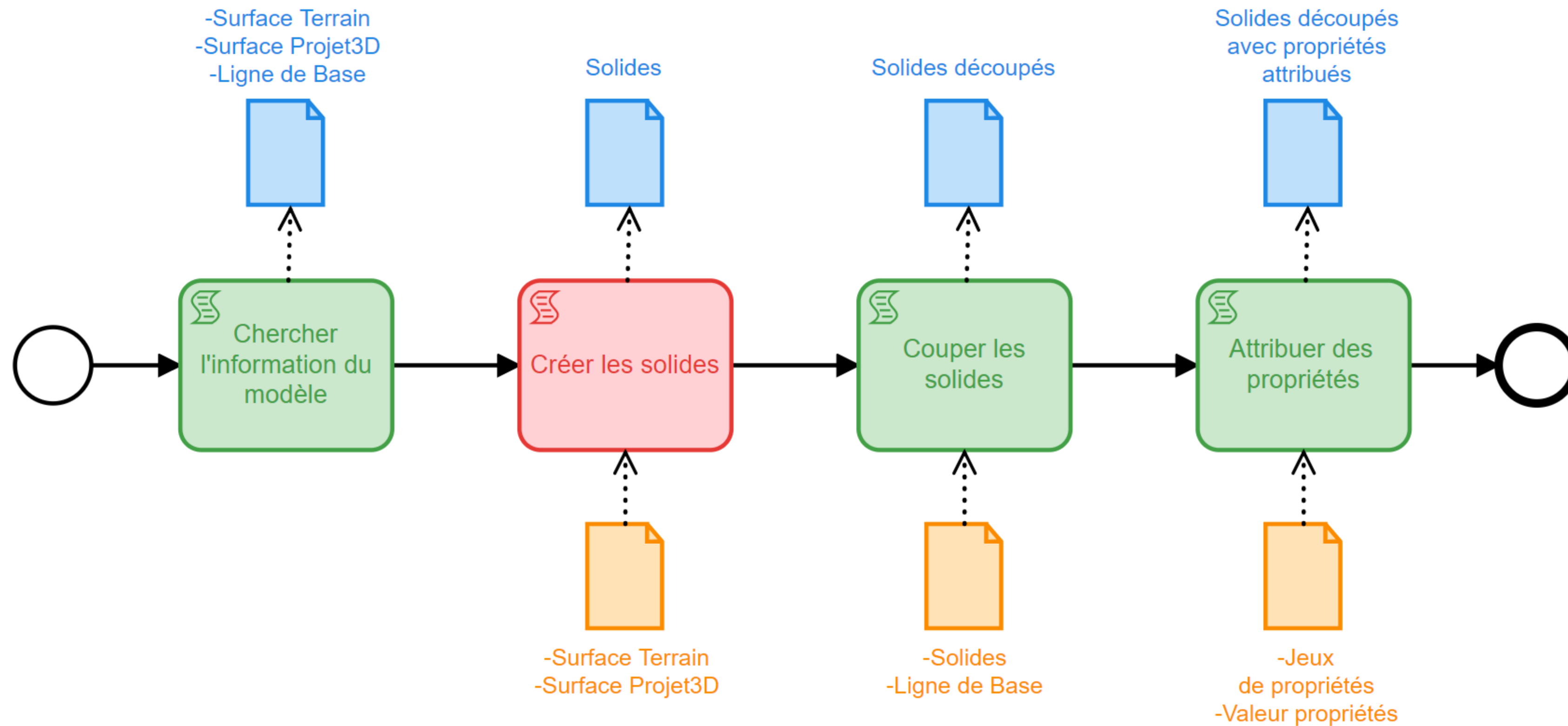


# 1 – Quel workflow ?



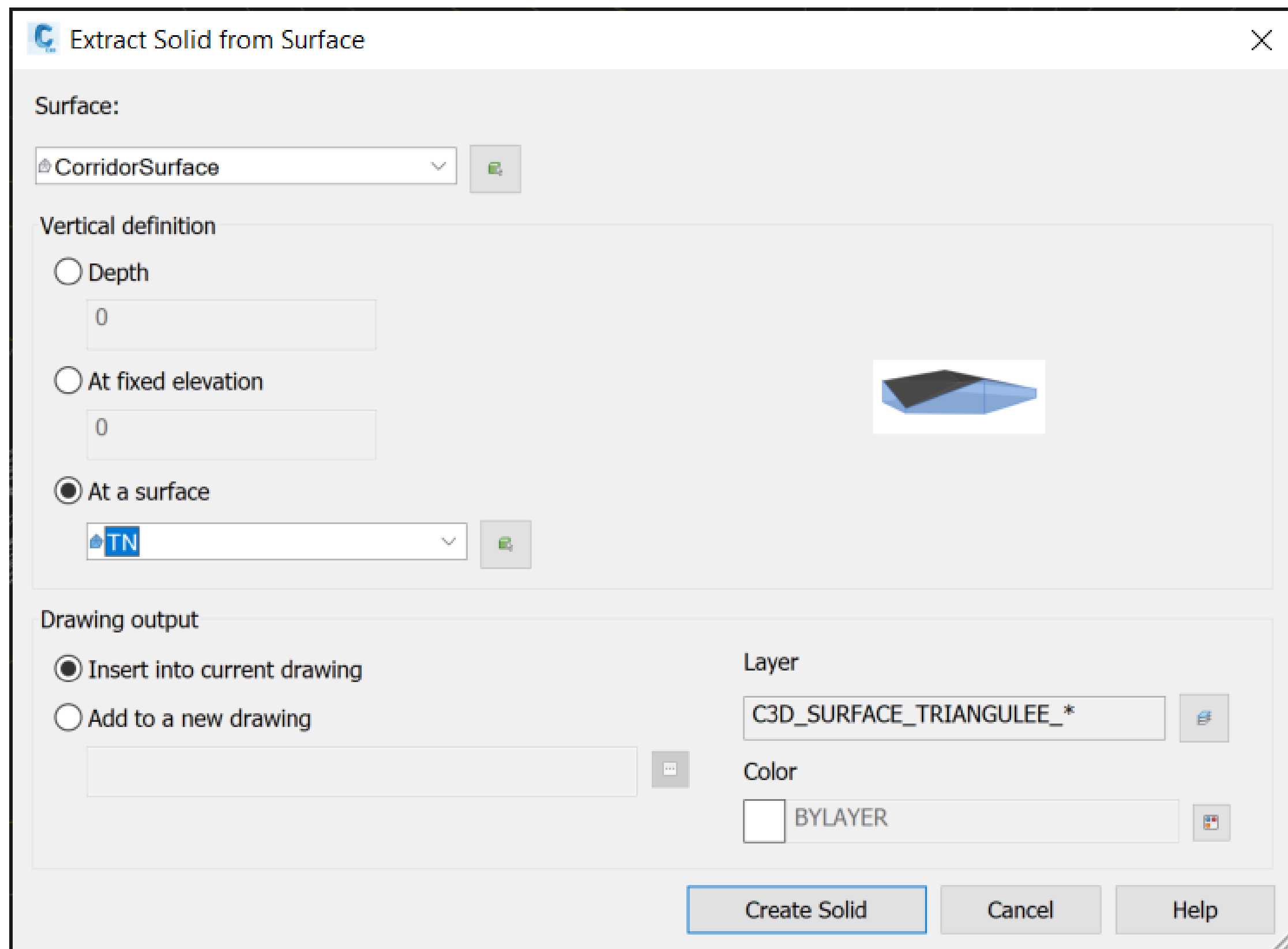


## 2 – Identifier entrées/sorties et capacités





# S'appuyer sur des ressources spécialisées



Extracts the solids between top and bottom Tin Surfaces.

`TinSurfaceExtensions.CreateSolidsAtSurface (surface: TinSurface, bottomSurface: TinSurface, layer: string = "0", path: string = ""): Object[]`

## TinSurfaceExtensions.CreateSolidsAtSurface

surface	>	Object[]
bottomSurface	>	
layer	>	
path	>	

AUTO

Function



# Construction des graphes

Quels types de graphes ?

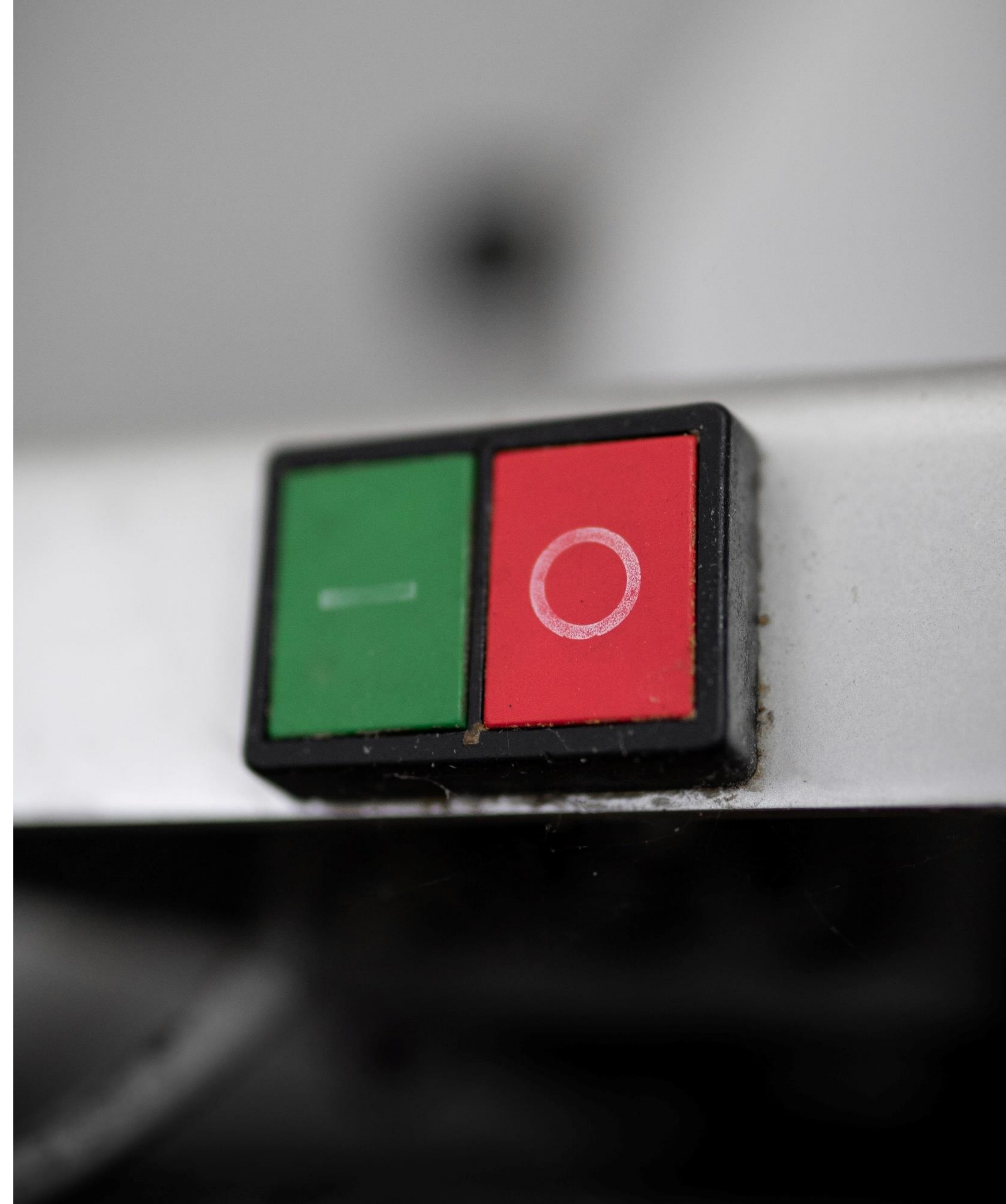
- Performantes
- Génériques
- Extensibles





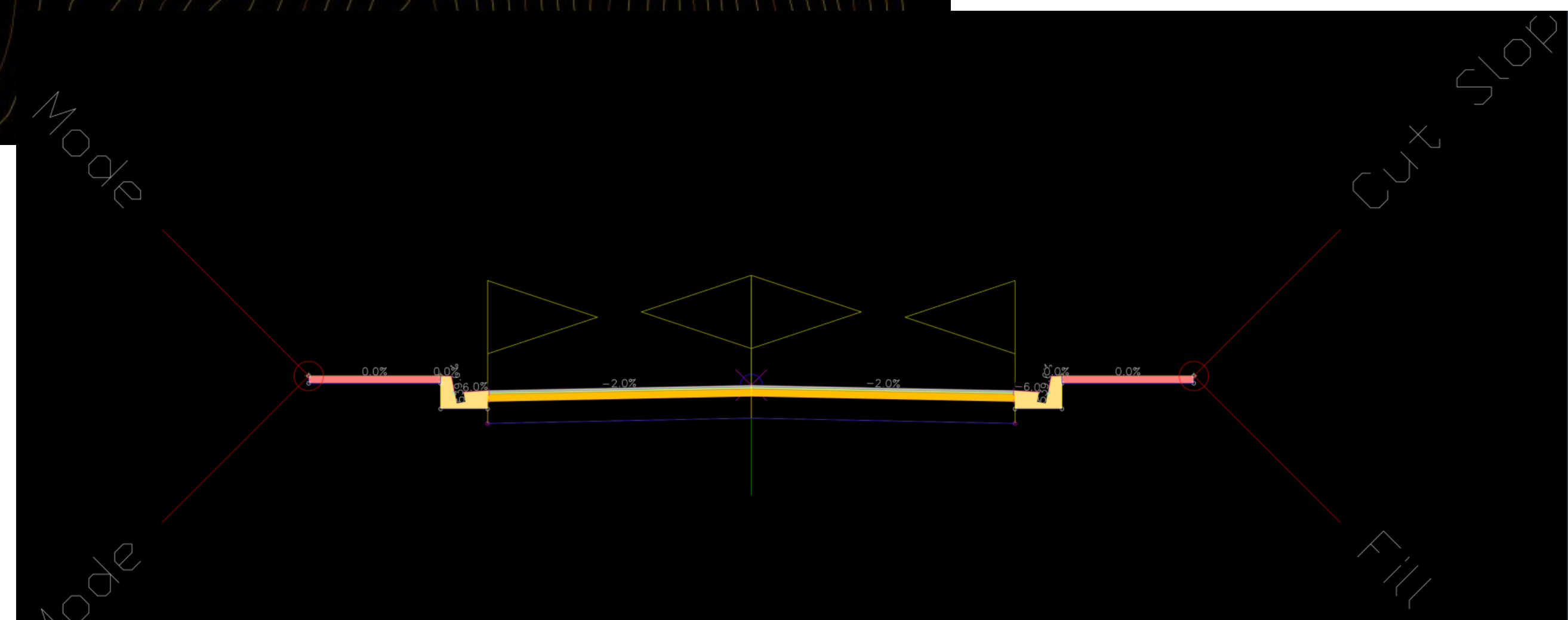
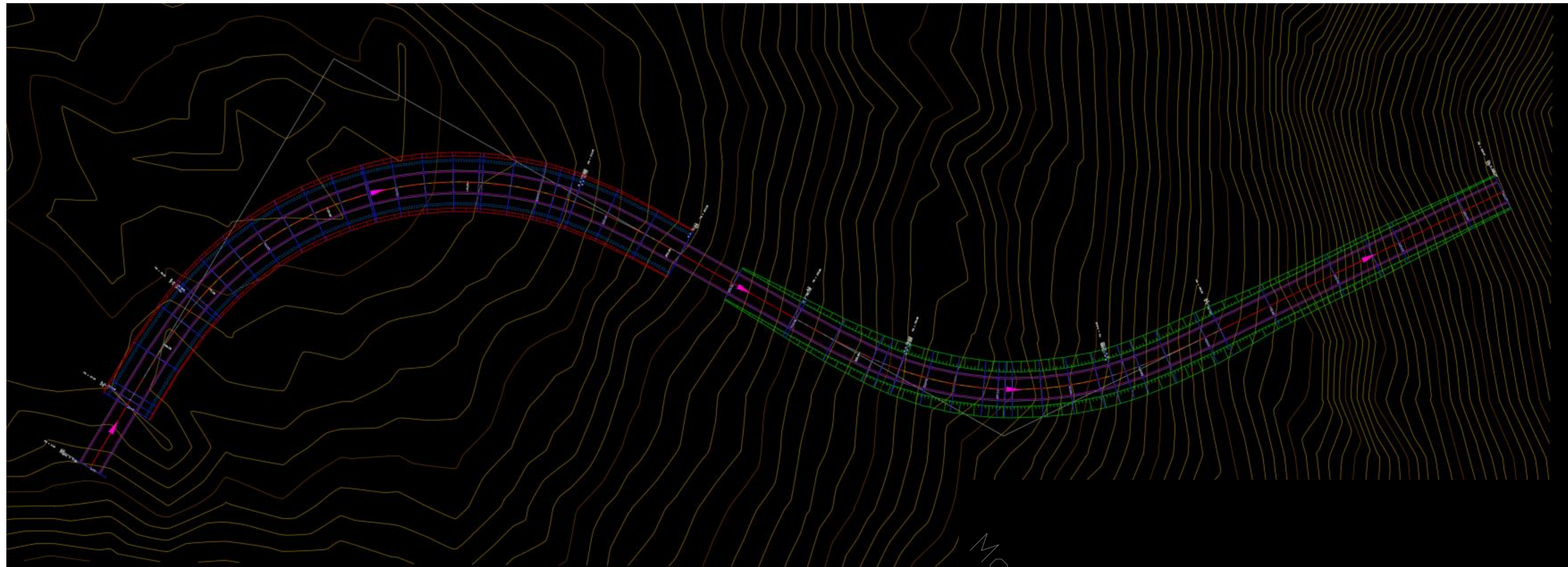
# Un POC avant des graphes optimisés

- Réalisation de tests rapidement :
  - Est-ce que la logique suivie fonctionne ?



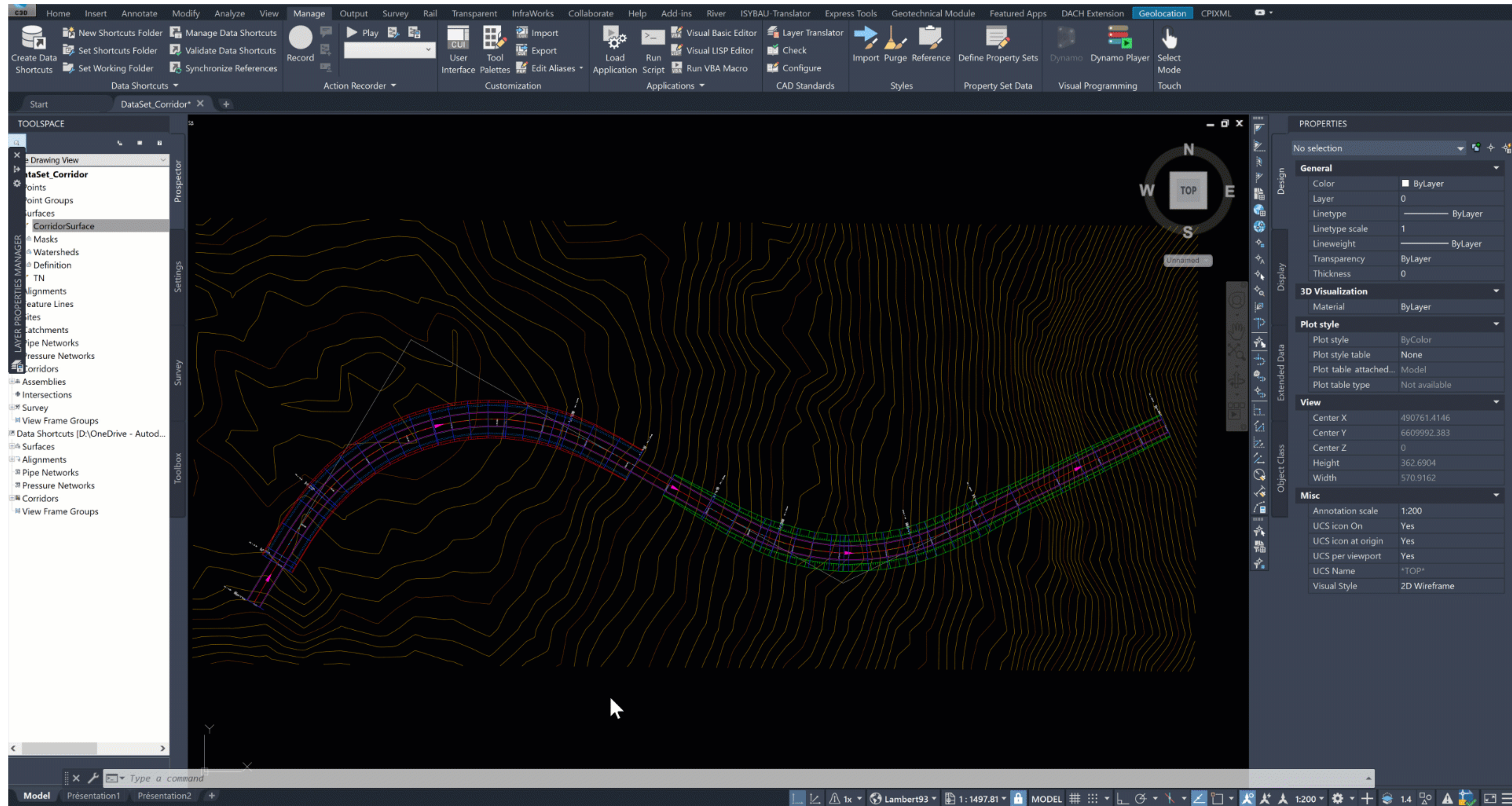


# Dataset pour tester





# Et dans Dynamo ?





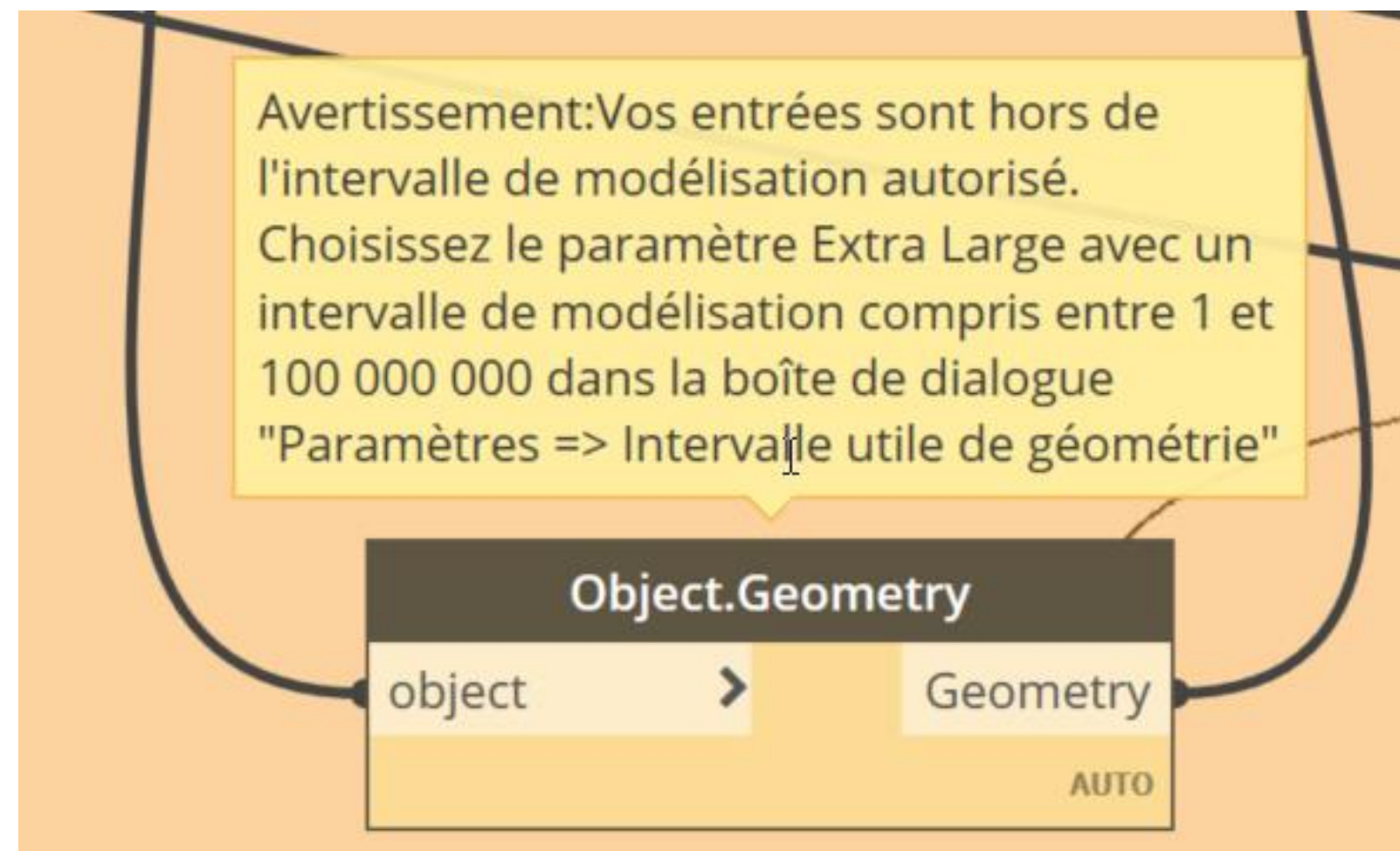
# Quelques concepts clés sur Dynamo ...

A tenir en compte pendant la construction des graphes



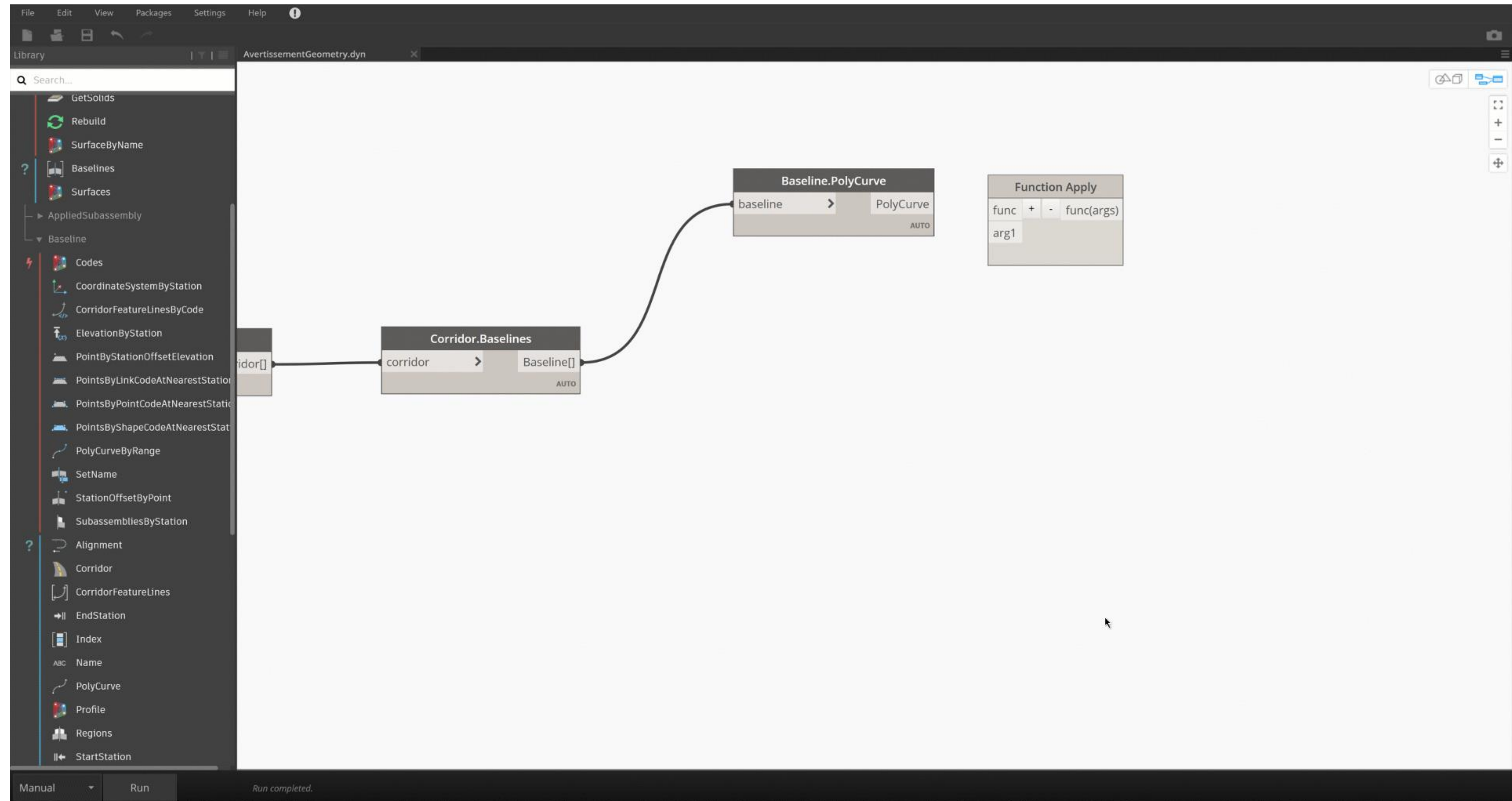
# Intervalle de géométrie

- Avertissement très commun
- Apparaisse quand on est loin du 0,0,0 alors chaque fois qu'on a un modèle géoréférencé
- **JAMAIS** changer l'intervalle de géométrie :
  - Induira des erreurs dans la modélisation





# Intervalle de géométrie





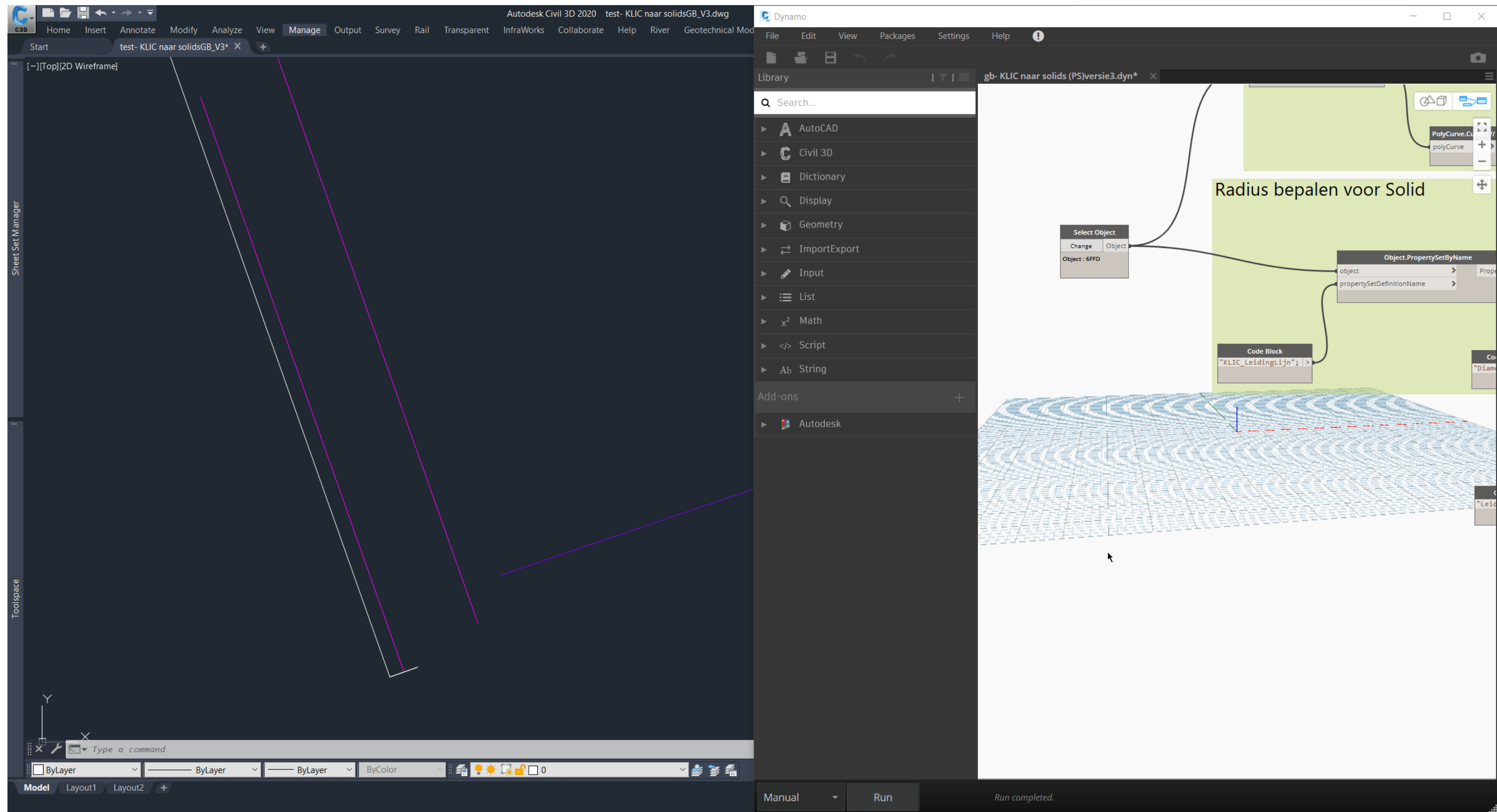
# « Binding »

- Très important à comprendre
- Liaison de fichier avec les éléments créés
- Problématique pour le placement des blocs



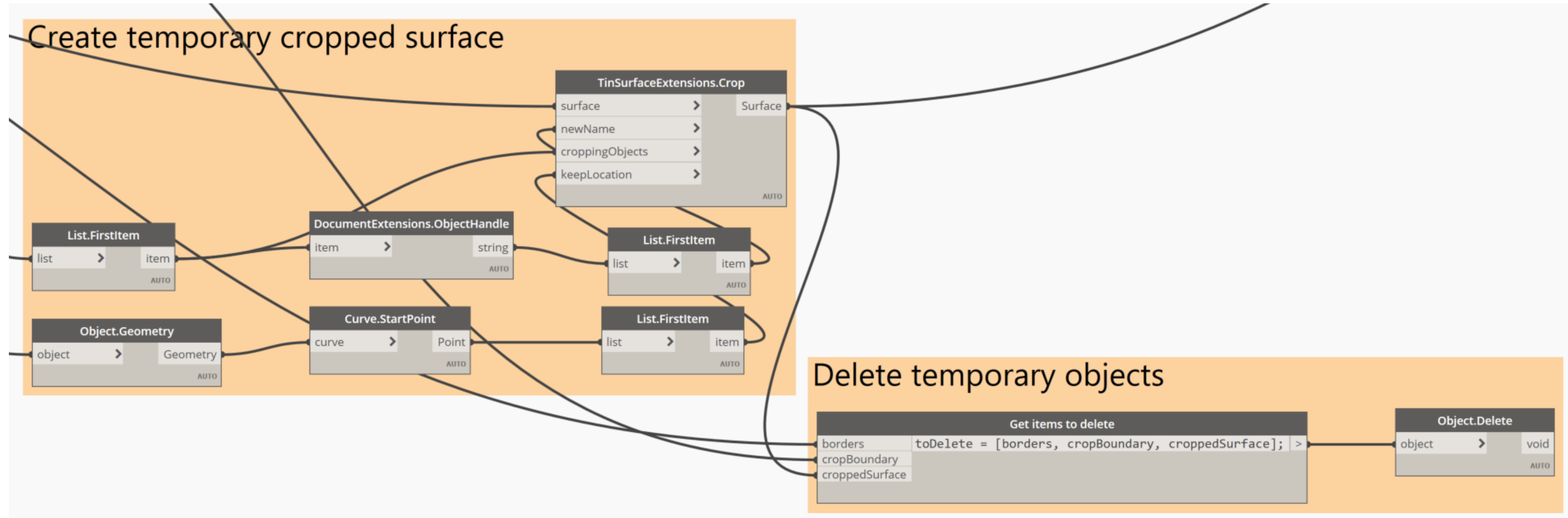


# Problèmes avec le binding





# Supprimer les objets temporaires créés sur Civil 3D



- Certains nœuds ou « actions » ne prennent que des objets AutoCAD (et pas Dynamo)
- Transparent pour l'utilisateur final



# Objectifs d'apprentissage



## IDENTIFIER LES OPPORTUNITÉS D'AUTOMATISATION

- ✓ L'automatisation pour faire plus, mieux et avec moins de ressources
- ✓ Cas d'usage : tâches répétitives, tâches avec peu de valeur ajoutée, processus itératifs, etc.
- ✓ Se focaliser sur les « quick wins »



## DÉVELOPPER DES SOLUTIONS INDUSTRIALISABLES

- Dynamo Player
- Documentation du fonctionnement
- Modularisation des solutions



## PENSER ANALYTIQUEMENT

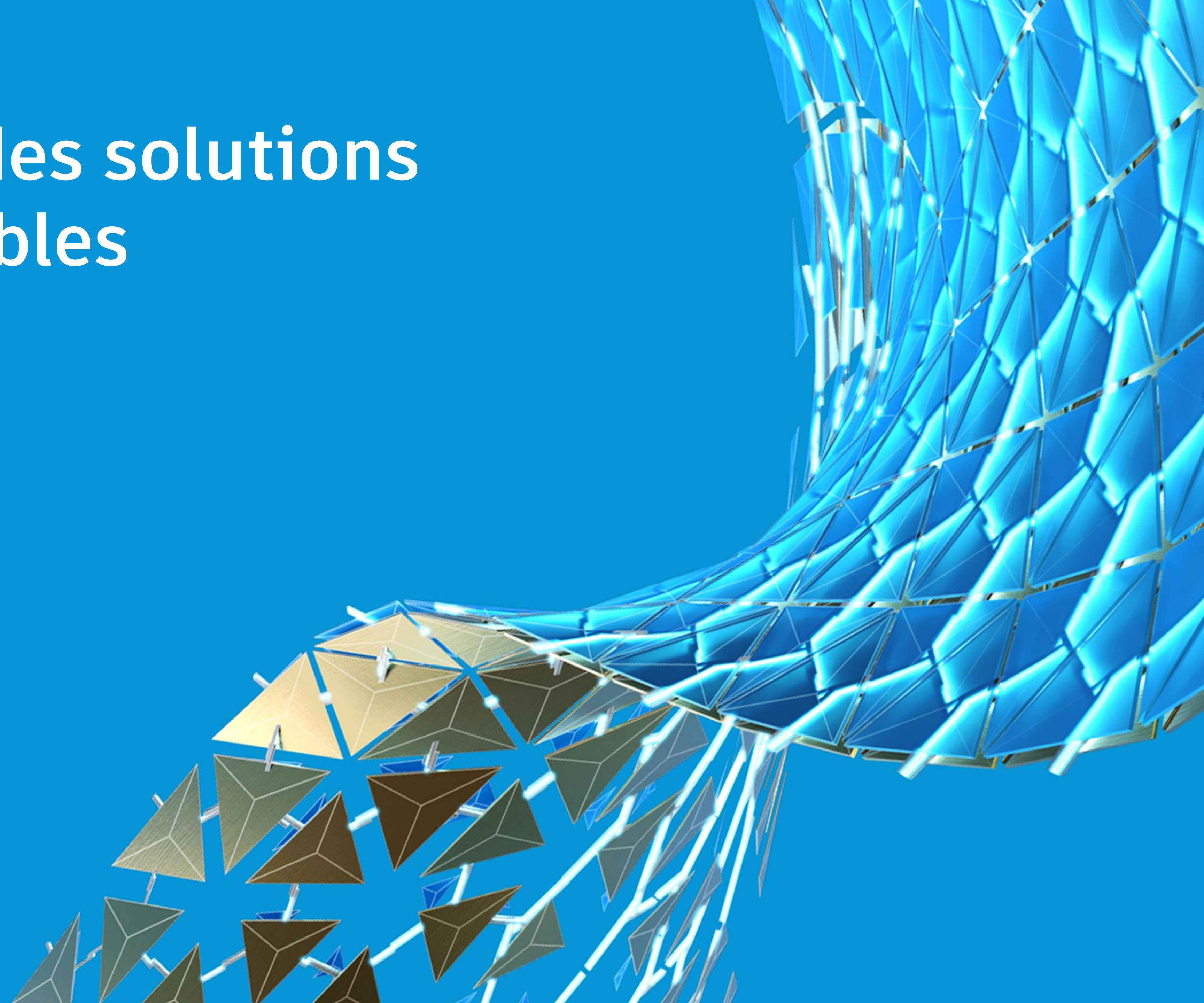
- ✓ Décrire la séquence logique pour résoudre la problématique
- ✓ Créer un « POC » rapidement, utiliser de datasets génériques
- ✓ Ne pas changer l'intervalle de géométrie et avoir en tête le binding et l'utilisation des objets temporaires



## IMPLÉMENTER L'AUTOMATISATION



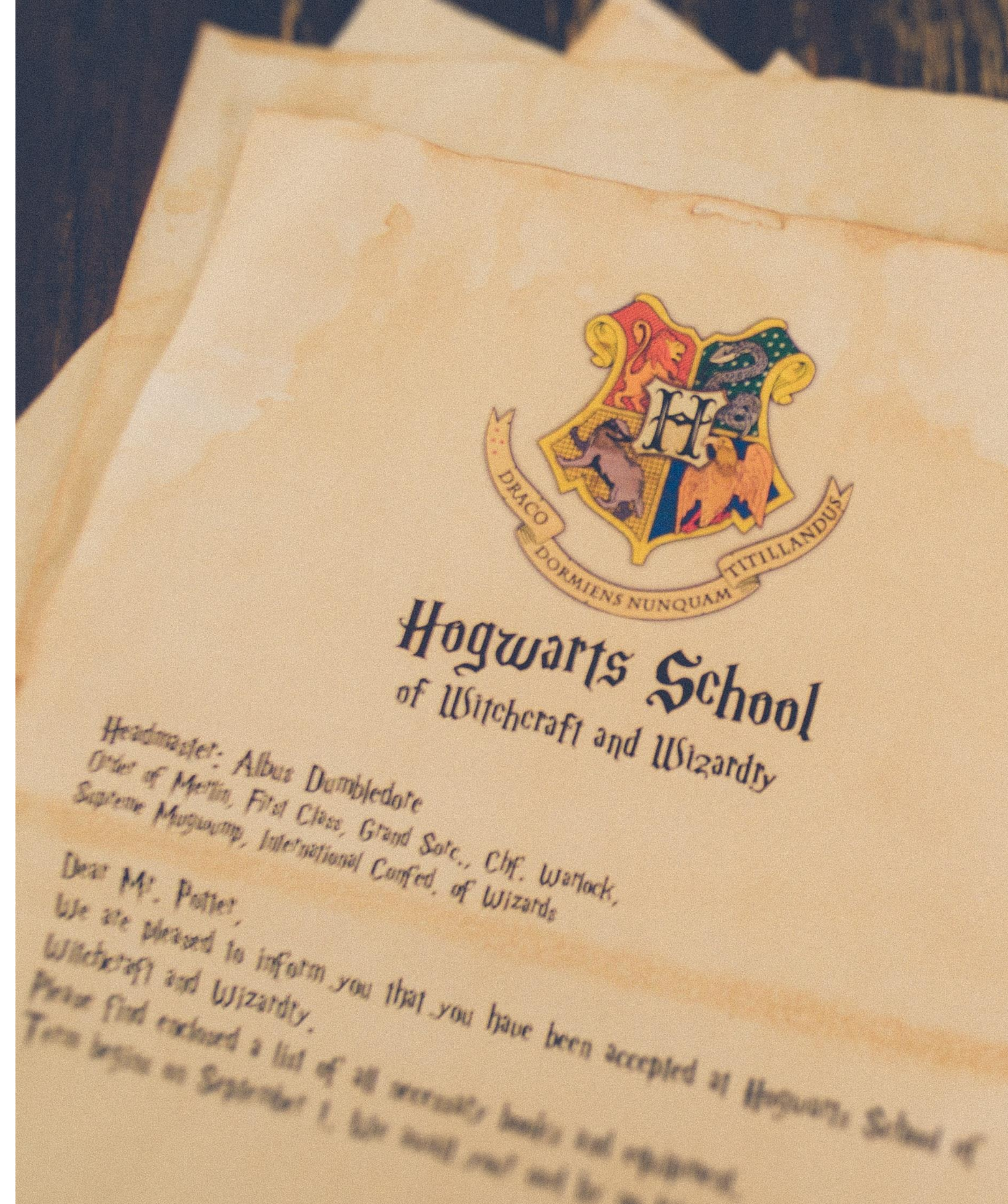
Développer des solutions  
industrialisables





# Dynamo Player

- Directement depuis Civil 3D
- Aucune saisie manuelle dans Dynamo
- Dans quelques clics / bouton magique










# Graphes Paramétriques

- Dynamo Player permet de changer les données d'entrée
- Utilisateur peut tester différentes configurations rapidement :
  - Exemple : lampadaire tout le 20/25/30 m

Dynamo Player




 WIP\_TIN  
 Ready


 File Path | Only if solids are created in a new document :  

Browse...


D:\OneDrive - Autodesk\O\_Projects\Jacobs\TIN\_Test\Solids\_Result.dwg


 String | Property Set Name  
Name is case sensitive  
Property Set must already exist in the document :  

Jacobs\_Test


 Choose EG surface  
Surfaces are ordered as in your prospector, starting from 1.  
So your first surface in the prospector, corresponds to 1.  
If you have more than 10 surfaces, then you need to go into Dynamo to change this number slider in th  


1



 Choose corridor surface for the corridor for which you want to create the solids.  
(this is used to crop the EG surface).  
Surfaces are ordered as in your prospector, starting from 1.  
So your first surface in the prospector, corresponds to 1  
If you have more than 10 surfaces, then you need to go into Dynamo to change this number slider in th  


2



 Create solids in a new document ?  
True = Yes  
False = No :  

☐

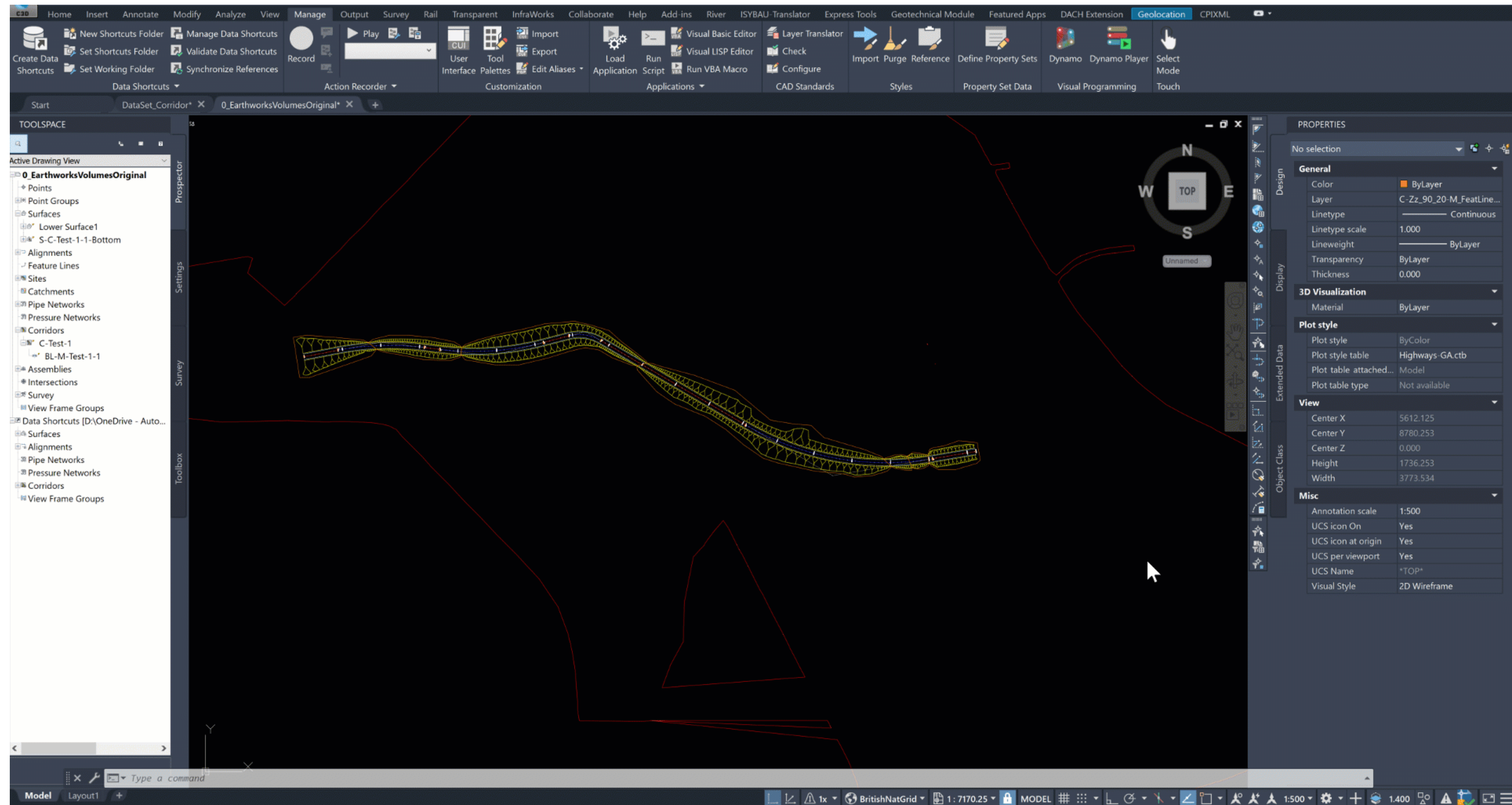
False

 String | Define layer to place solids  
If the layer doesn't exist, it will be created :  

Solids



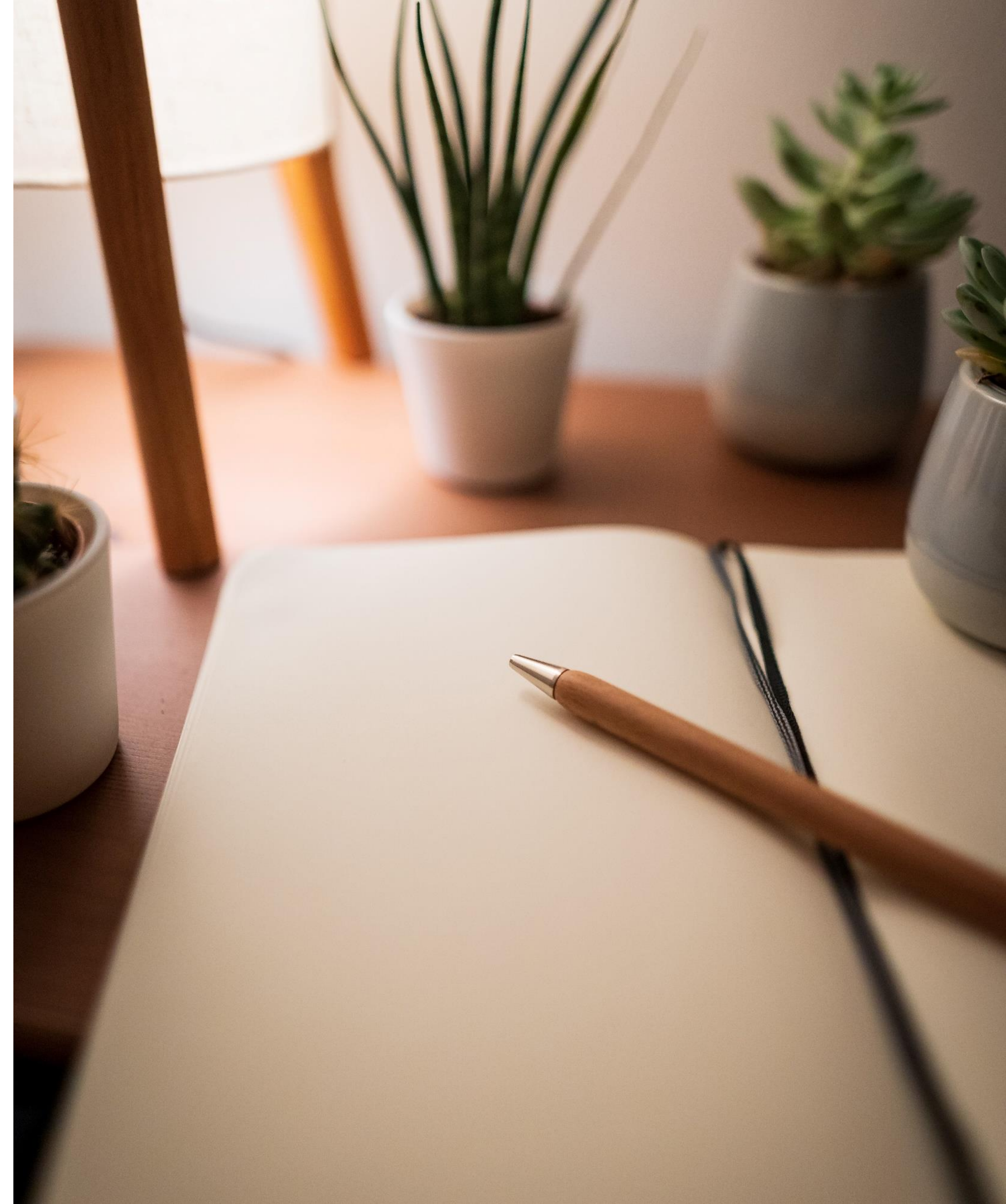
# Et dans Dynamo ?





# Instructions

- **Instructions intégrées dans le graphe :**
  - Le but du graphe
  - Les hypothèses prises
  - Les données d'entrée
  - Les optimisations possibles
  - Les problématiques identifiées
  - Les tests qui ont été réalisés
  - ....





# Exemple d'instructions

## NOTES

### GRAPH INFO

Copyright 2020 Autodesk, Inc. All rights reserved.  
Company: Jacobs  
Office: <office>

Version: 1.0.0  
Author: David Licon  
david.licon@autodesk.com

Keywords: Earthworks Solids,

Tested on:  
Dynamo: 2.4.1  
Civil 3D: 2020.4  
Civil 3D Toolkit Package : 1.1.9

### Useful Links

Your code goes here

### INSTRUCTIONS

The purpose of this graph is to create earthwork solids.

To achieve this goal, the Dynamo graph is executing the following operations :

- Crop existing ground surface using the corridor surface as the cropping limit. This is due to a limitation of Civil 3D: being able to create solids between tin surfaces made up of less than 100K triangles.

- Create earthwork solids. As with the Civil 3D UI, the solids are created between two tin surfaces:
  - o The corridor surface,
  - o The existing ground surface (bottom surface).

Two solids will be initially created, one for the fill and one for the cut. Tests show that the first solid in the resulting list is always the fill solid (more tests may be required as this was tested in 3 different documents only).

- Split the solids to have separate solids for the two sides of the carriageway. Solids are split along the corridor baseline.

- The resulting fill and cut solids are sent into Civil 3D on user-defined layers

#### User inputs

- Choose parametrically the existing ground surface (surfaces are sorted alphabetically as in the prospector – index 1 corresponds to the first surface in alphabetical order) [slider],

- Choose parametrically the corridor surface (surfaces are sorted alphabetically as in the prospector – index 1 corresponds to the first surface in alphabetical order) [slider] ,

- Choose parametrically the corridor (corridors are

### KNOWN ISSUES AND LIMITATIONS

The following issues have been identified:

- Solids will not be created if any surface exceeds 100K triangles
  - o The graph will crop the surface using the corridor surface as the object to crop

- Cropping will only work if the corridor surface style is set to "\_No Display". It may occur that depending on the country kit used, the graph will not work also with the "no display" style. Tests were done with Client provided datasets and no issues were found.

- If the corridor is long enough, it may happen that the surface created from the crop exceeds the 100K triangles, today the graph will not be able to handle this

- The graph will not select corridors who have a feature line as a baseline

- In a file with multiple corridors, only one corridor can be created at a time

- Created solids will only be split by type (cut or fill) and by the baseline. This means that solids may consist of different earthworks.

- Solids are created by lofting the EG surface and the corridor surface, this means that solids will be created taking into account every point (on the X and Y plane) where the corridor surface intersects the EG, even when variations are really small.

- Errors may occur because of the offset distance used to create the cropping object and based on the corridor surface. If this happens, try lowering the value of the Offset (for cropping purposes) node in the Input section

### GUIDELINES

Read the instructions.

Add Notes and Comments to the graph.

Use Node Groups and the Standard Color Coding.

Rename Nodes: <OriginalName> | <Description>.

Write Input and Output Notes for Python Scripts.

Prefer repeatable simple node structures.

Simple is better than complex.

Complex is better than complicated.

Readability counts.

Special cases aren't special enough to break the rules.

Although practicality beats purity.

Errors should never pass silently.

Unless explicitly silenced.

In the face of ambiguity, do not guess.

There should be one obvious way to do it.

Now is better than never.

Although never is often better than \*right\* now.



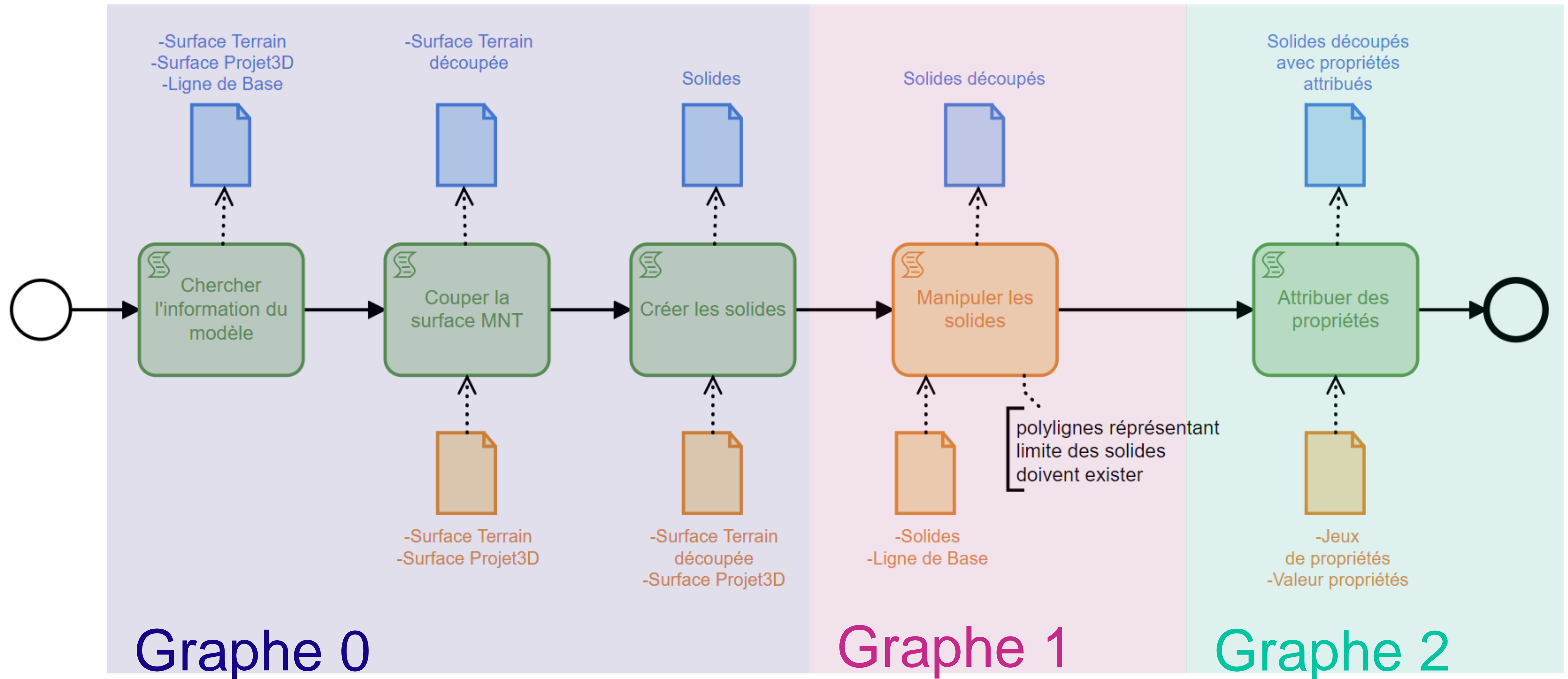
# Modulariser

- Plus facile à tester
- Plus facile à comprendre
- Plus facile à débbugger



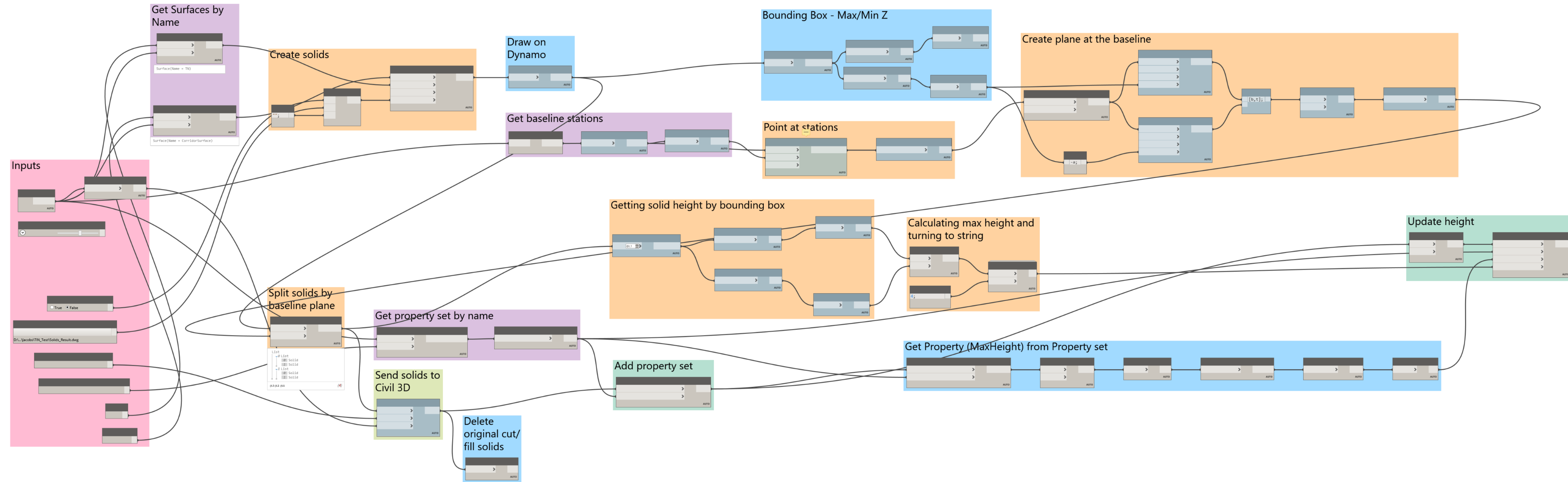


# Résultat final workflow



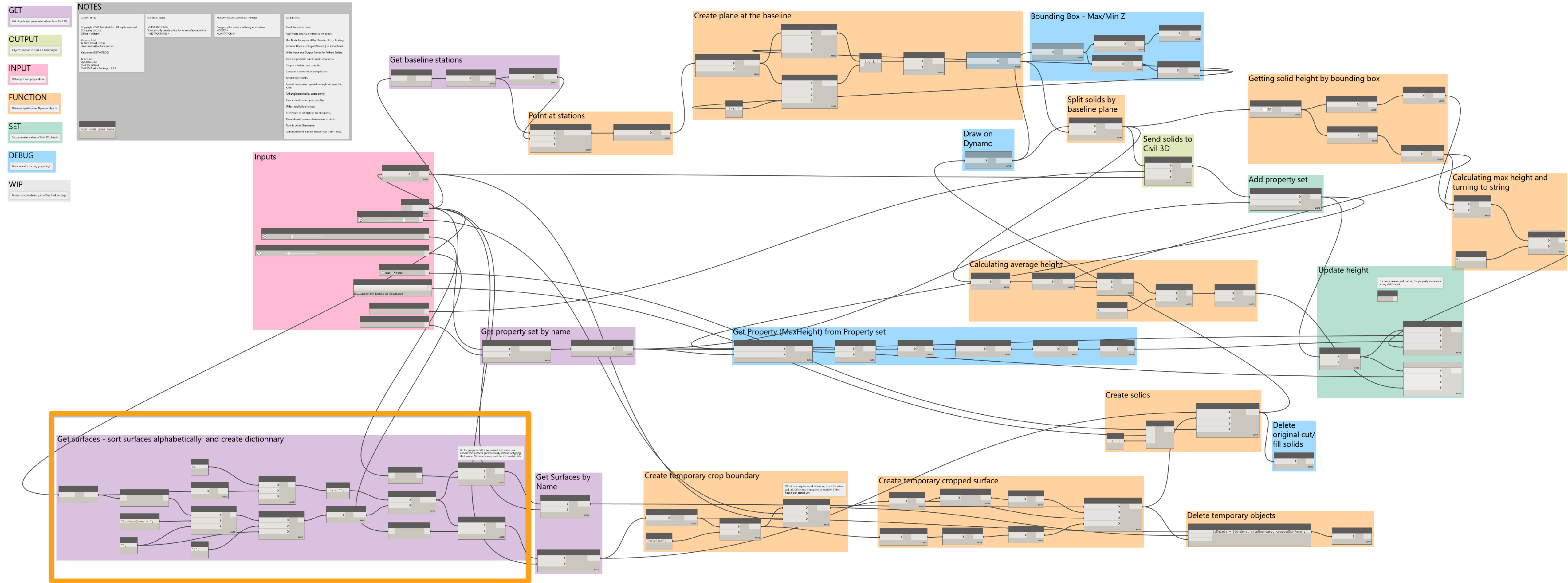


# POC





# Paramétrique





# Graphe 0 : Créer des solides

- GET
  - Get objects and parameter values from Civil 3D
- OUTPUT
  - Object Creation in Civil 3D, final output
- INPUT
  - Data input and preparation
- FUNCTION
  - Data manipulation on Dynamo objects
- SET
  - Set parameter values of Civil 3D objects
- DEBUG
  - Nodes used to debug graph logic
- WIP
  - Notes not considered part of the final package

GRAPH INFO

Copyright 2020 Autodesk, Inc. All rights reserved.  
Company: Jacobs  
Office: "surface"  
  
Version: 1.0.0  
Author: David Lironi  
david.lironi@autodesk.com  
  
Keywords: Earthworks Solids,  
  
Tested on:  
Dynamo 2.11  
Civil 3D 2024A  
Civil 3D Toolset Package 1.1.0  
  
Your code goes here

INSTRUCTIONS

The purpose of this graph is to create earthwork solids.  
  
To achieve this goal, the Dynamo graph is executing the following operation:  

- Crop existing ground surface using the corridor surface as the cropping link. This is due to a limitation of Civil 3D being able to create solids between two surfaces made up of less than 100K triangles.
- Create earthwork solids. As with the Civil 3D UI, the solids are created between two surfaces:
  - o The corridor surface
  - o The existing ground surface (bottom surface).
- Two solids will be initially created, one for the fill and one for the cut. Tests show that the first solid in the resulting list always the fill solid (even tools may be required as this was tested in 3 different documents only).
- Split the solids to have separate solids for the two sides of the canopyway. Solids are split along the corridor baseline.
- The resulting fill and cut solids are sent into Civil 3D as user-defined types.

User inputs:

- Choose parametrically the existing ground surface (surface or corridor) alphabetically as in the prospector - index 1 corresponds to the first surface in alphabetical order (solid).
- Choose parametrically the corridor surface (surface or corridor) alphabetically as in the prospector - index 1 corresponds to the first surface in alphabetical order (solid).
- Choose parametrically the baseline (baseline) will be shown as created - normally, the fill baseline will correspond to the middle of the canopyway (solid).

-Defining the layer on which the fill solid objects will be created (text).  
  
-Defining the layer on which the cut solid objects will be created (text).  
  
-Choose if the solids are to be created in a new document or not (Boolean).

- o Give the file path to the new document (file path).

Output nodes on the player  
  
Watch nodes are provided on the player to indicate to the user the chosen surfaces, corridor and baseline

KNOWN ISSUES AND LIMITATIONS

- The following issues have been identified
- Solids will not be created if any surface exceeds 100K triangles
  - o The graph will crop the surface using the corridor surface as the object to crop
- Cropping will only work if the corridor surface style is set to "No Display". It may occur that depending on the styling list used, the graph will not work with the "No display" style. Tests were done with Client provided datasets and no issue were found.
- If the corridor is long enough, it may happen that the surface created from the crop exceeds the 100K triangles, today the graph will not be able to handle this.
- The graph will not select corridors who have a feature line as a baseline
- In a file with multiple corridors, only one corridor can be created at a time
- Created solids will only be split by type (cut or fill) and by the baseline. This means that solids may consist of different earthworks
- Solids are created by lifting the EG surface and the corridor surface, this means that solids will be created taking into account every point (on the X and Y plane) where the corridor surface intersects the EG, even when variations are really small.
- Errors may occur because of the offset distance used to create the cropping object and based on the value of the Offset for cropping (reported node in the input section)

GUIDELINES

- Read the instructions.
- Add Notes and Comments to the graph.
- Use Node Groups and the Standard Color Coding.
- Rename Nodes: <OriginalName> - <Description>.
- Write Input and Output Notes for Python Scripts.
- Prefer repeatable simple node structures.
- Simple is better than complex.
- Complex is better than complicated.
- Readability counts.
- Special cases aren't special enough to break the rules.
- Although practicality beats purity.
- Errors should never pass silently.
- Unless explicitly silenced.
- In the face of ambiguity, do not guess.
- There should be one obvious way to do it.
- Now is better than never.
- Although never is often better than "right" now.

Get corridors - sort alphabetically and create dictionary

Get baselines - Create dictionary

Get surfaces - sort surfaces alphabetically and create dictionary

Get baseline stations

Get Surfaces by Name

Inputs

Point at stations

Create plane at the baseline

Split solids by baseline plane

Send Fill solids to Civil 3D

Send Cut solids to Civil 3D

Bounding Box - Max/Min Z

Draw on Dynamo

Create temporary crop boundary

Create temporary cropped surface

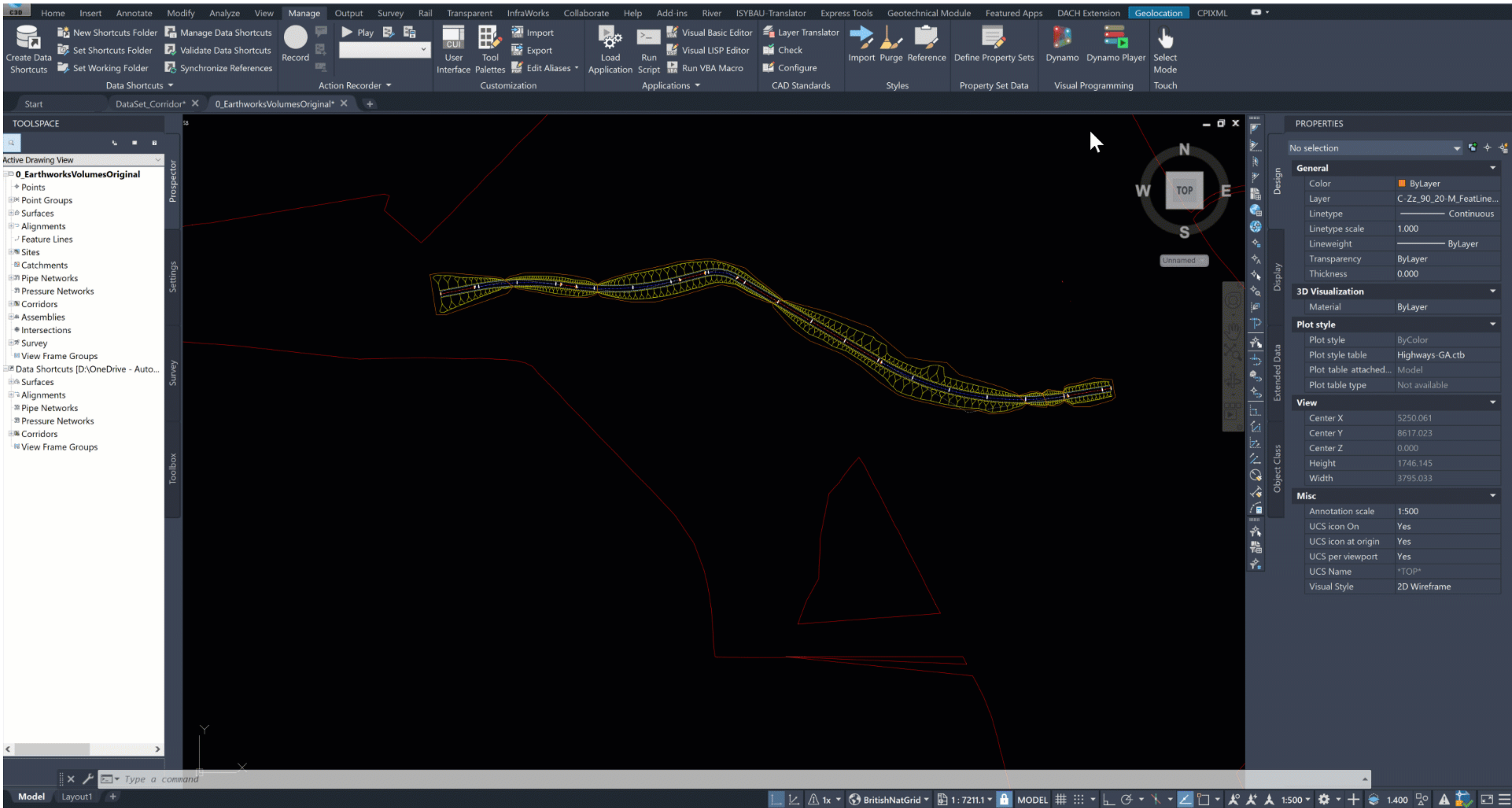
Delete temporary objects

Visualization/Debugging only - Separate Fill and Cut and Draw on Dynamo

Create solids



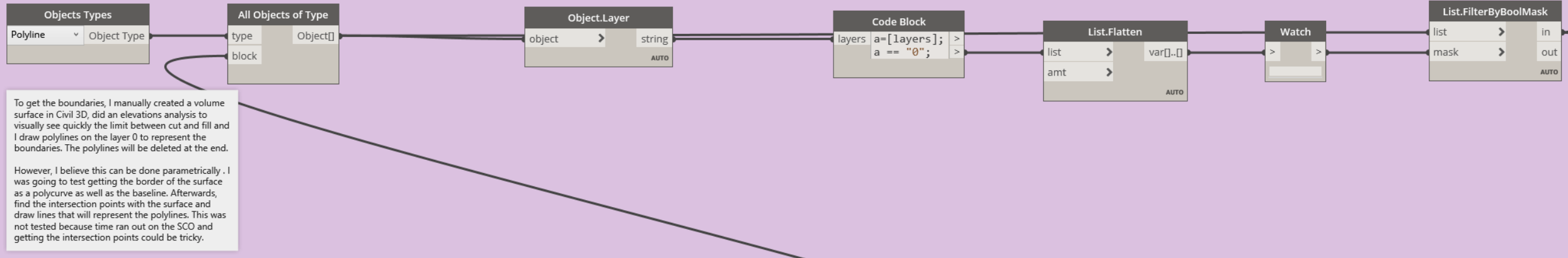
# Graphe 0 : Créer des solides



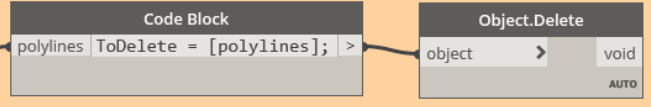


# Graphe 1 : Manipuler les solides

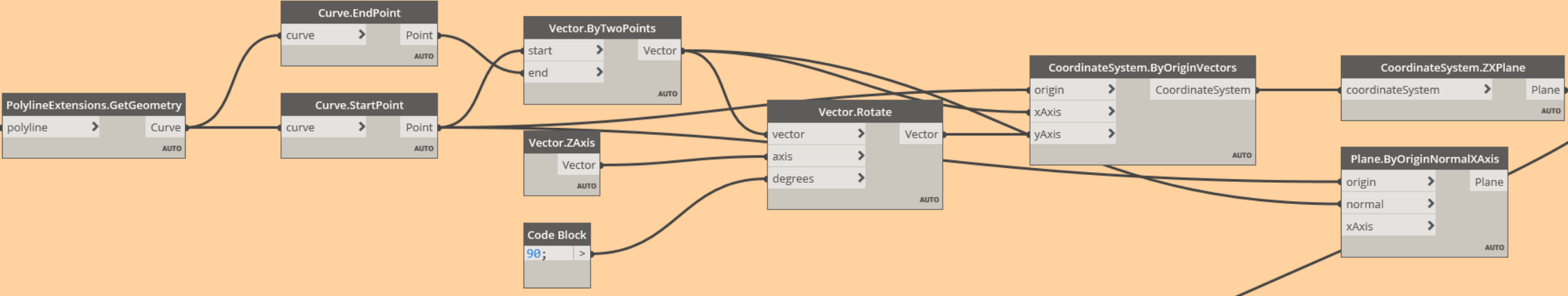
## Get boundaries between earthworks



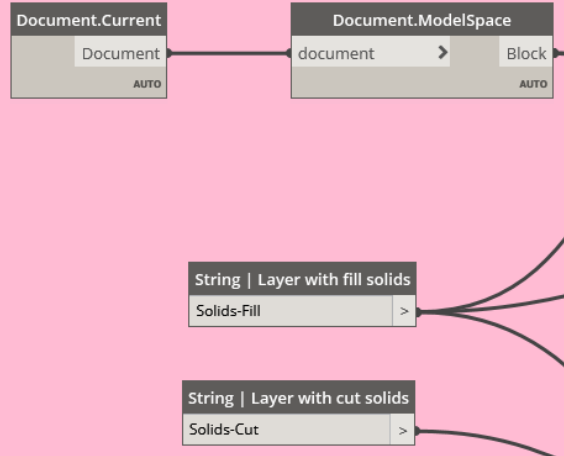
## Delete polylines



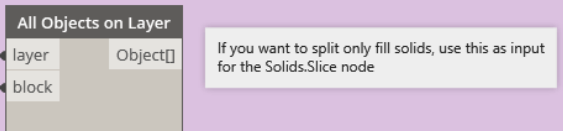
## Getting ZX plane of polylines



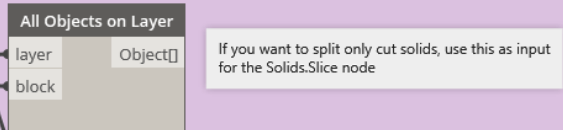
## Inputs



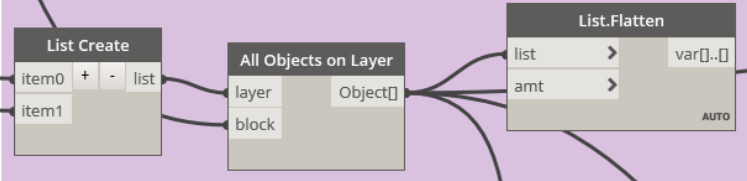
## Get Fill solids



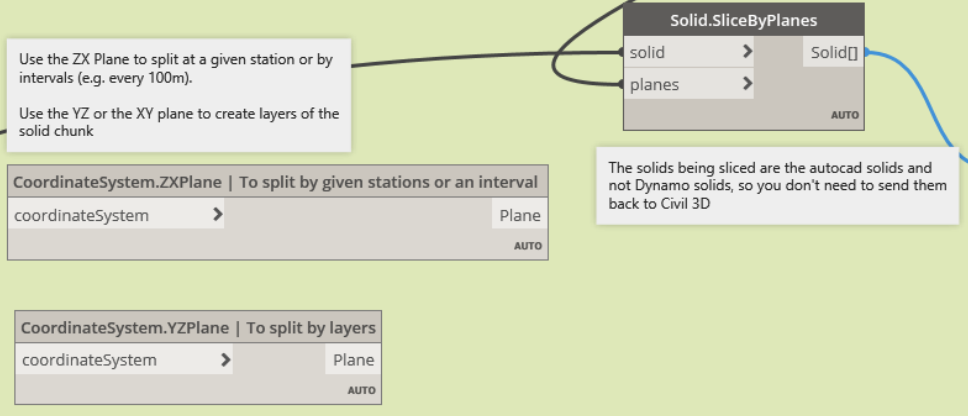
## Get Cut solids



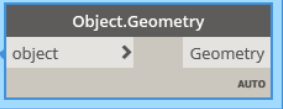
## Get all solids



## Slicing solids by planes



## Draw Geometry on Dynamo



## GET

Get objects and parameter values from Civil 3D

## OUTPUT

Object Creation in Civil 3D, final output

## INPUT

Data input and preparation

## FUNCTION

Data manipulation on Dynamo objects

## SET

Set parameter values of Civil 3D objects

## DEBUG

Nodes used to debug graph logic

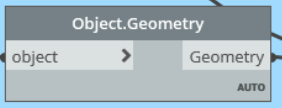
## WIP

Notes not considered part of the final package

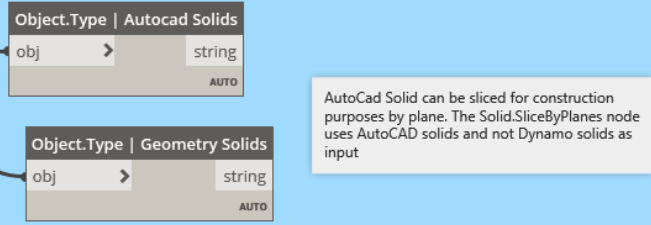
## NOTES

GRAPH INFO	INSTRUCTIONS	KNOWN ISSUES AND LIMITATIONS	GUIDELINES
<p>Copyright 2020 Autodesk, Inc. All rights reserved. Company: Jacobs Office: &lt;office&gt; Version: 1.0.0 Author: David Licona david.licona@autodesk.com Keywords: [KEYWORDS] Tested on: Dynamo: 2.4.1 Civil 3D: 2020.4 Civil 3D Toolkit Package: 1.1.9</p>	<p>The purpose of this graph is to manipulate earthwork solids created by the Get Solids Graph.</p> <p>The following manipulation is being done:</p> <p>Slicing solids by earth work.</p> <p>To achieve this goal, the Dynamo graph is executing the following operations:</p> <ul style="list-style-type: none"><li>-Looking for all the polylines in the graph</li><li>-Filtering the polylines in order to get only polylines on layer 0</li><li>-Determining the ZX plane of this polylines</li><li>-Selecting solids from fill and cut layers</li><li>-Slicing the selected solids by using the ZX plane of the polylines</li><li>-Deleting the polylines used for slicing (polylines on layer 0). If you don't want to delete the polylines used for the slicing, then freeze the delete node</li></ul> <p>User inputs</p> <ul style="list-style-type: none"><li>-User must create manually the polylines that indicate where the slicing is going to occur. These polylines must be on layer 0</li><li>-Defining the layer on which the fill solid objects are [text]</li><li>-Defining the layer on which the cut solid objects are [text]</li></ul> <p>Notes are provided along the groups to clarify or explain nodes used as well as possibilities.</p>	<ul style="list-style-type: none"><li>-For the moment, polylines will be needed to be created manually. The graph needs to be modified in such a way to identify the earthwork change.</li><li>-Graph will only work with polylines on layer 0</li><li>-Only run the graph once in a file. If the graph is run more than one time on the same file, it will delete previously created solids.</li></ul>	<p>Read the instructions.</p> <p>Add Notes and Comments to the graph.</p> <p>Use Node Groups and the Standard Color Coding.</p> <p>Rename Nodes: &lt;OriginalName&gt;   &lt;Description&gt;.</p> <p>Write Input and Output Notes for Python Scripts.</p> <p>Prefer repeatable simple node structures.</p> <p>Simple is better than complex.</p> <p>Complex is better than complicated.</p> <p>Readability counts.</p> <p>Special cases aren't special enough to break the rules.</p> <p>Although practicality beats purity.</p> <p>Errors should never pass silently.</p> <p>Unless explicitly silenced.</p> <p>In the face of ambiguity, do not guess.</p> <p>There should be one obvious way to do it.</p> <p>Now is better than never.</p> <p>Although never is often better than "right" now.</p>

## Draw on Dynamo

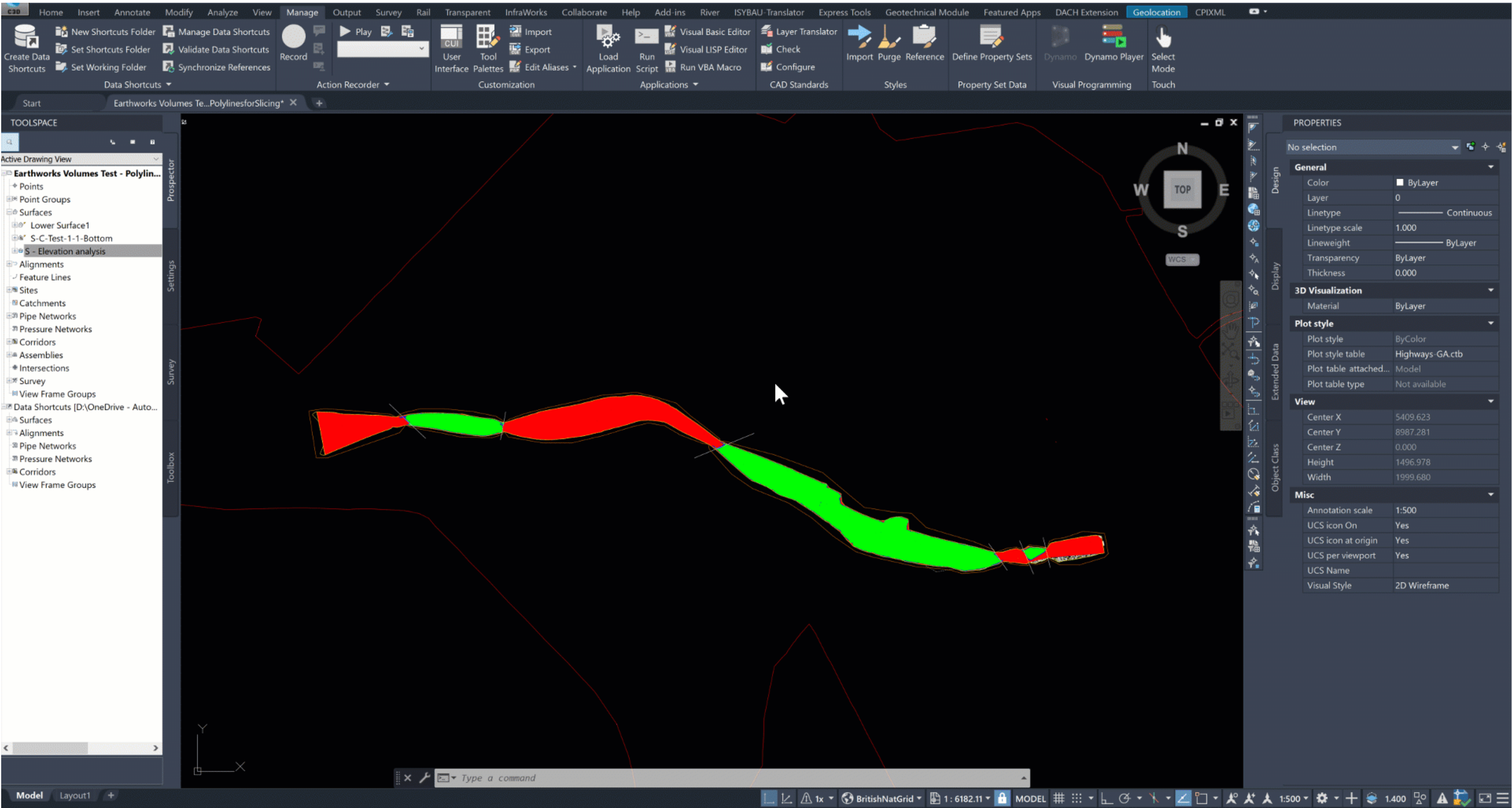


## For context > difference between ACAD solids and Geometry solids



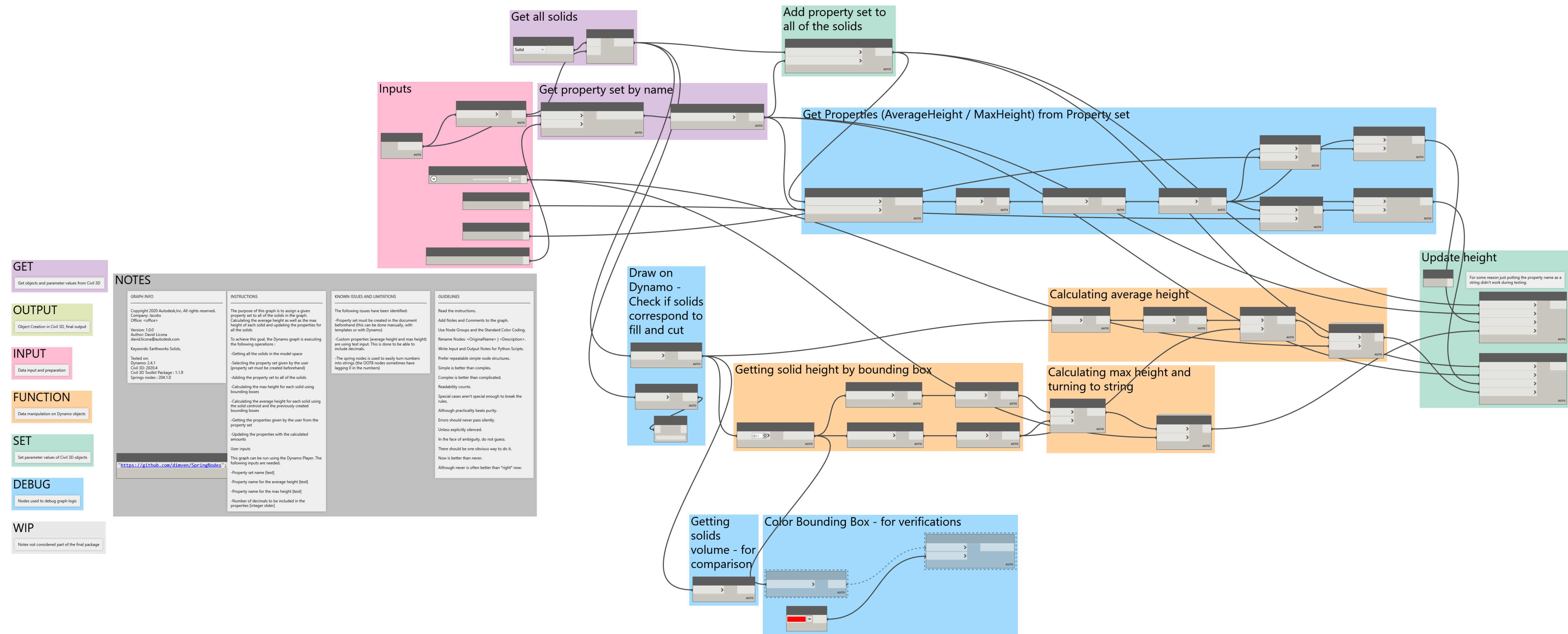


# Graphe 1 : Manipuler les solides



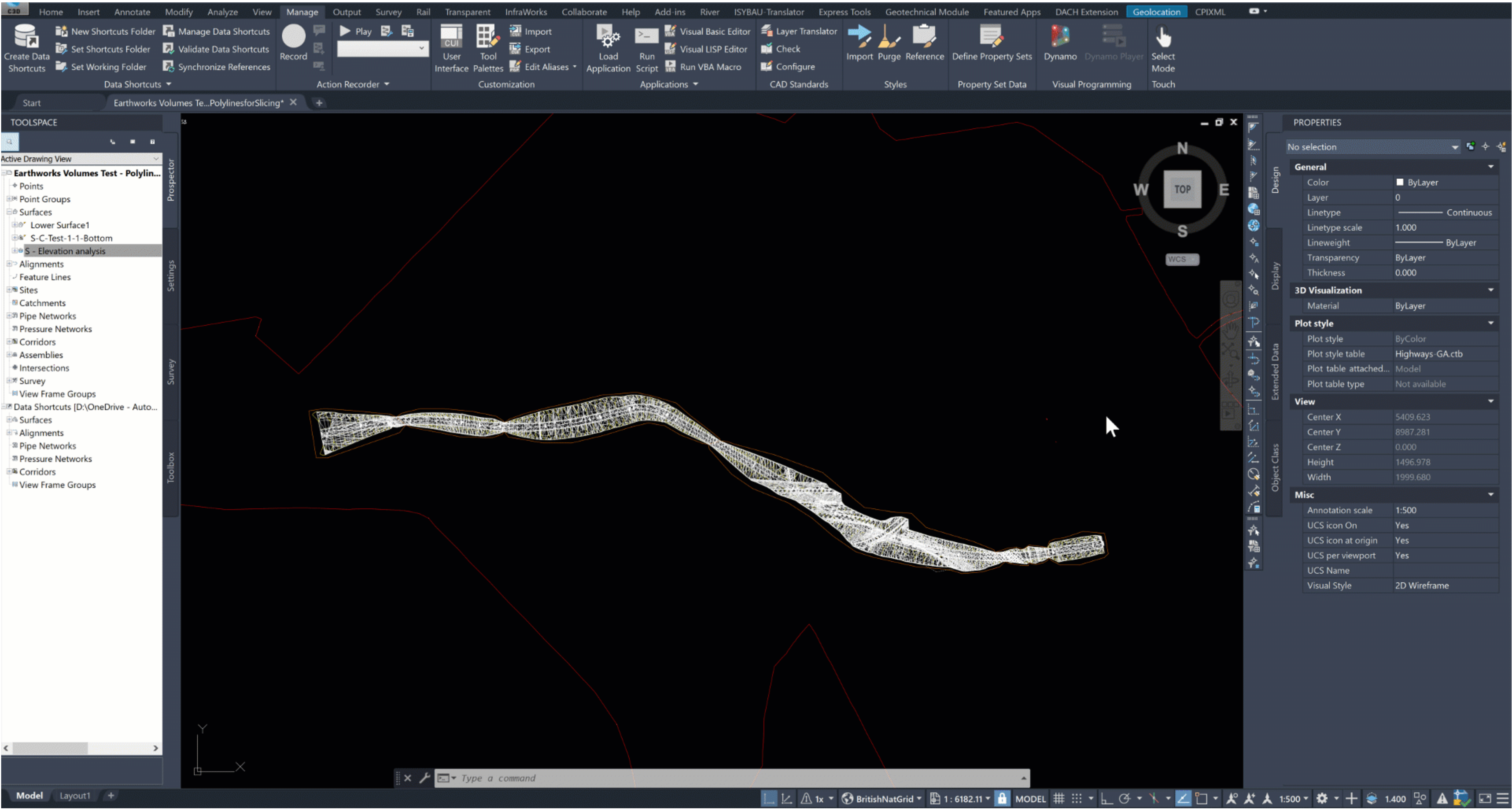


# Graphe 2 : Jeu de propriétés





# Graphe 2 : Jeu de propriétés





# Objectifs d'apprentissage



## IDENTIFIER LES OPPORTUNITÉS D'AUTOMATISATION

- L'automatisation pour faire plus, mieux et avec moins de ressources
- Cas d'usage : tâches répétitives, tâches avec peu de valeur ajoutée, processus itératifs, etc.
- Se focaliser sur les « quick wins »



## DÉVELOPPER DES SOLUTIONS INDUSTRIALISABLES

- Faire de solutions que puissent être utilisées avec le Dynamo Player
- Imbriquer les graphes avec des notes
- Modulariser les solutions pour tester et débbugger plus facilement



## PENSER ANALYTIQUEMENT

- Décrire la séquence logique pour résoudre la problématique
- Créer un « POC » rapidement, utiliser de datasets génériques
- Ne pas changer l'intervalle de géométrie et avoir en tête le binding et l'utilisation des objets temporaires



## IMPLÉMENTER L'AUTOMATISATION

- « Best practices »
- Communication des succès
- Ressources pour réussir



# Adoption





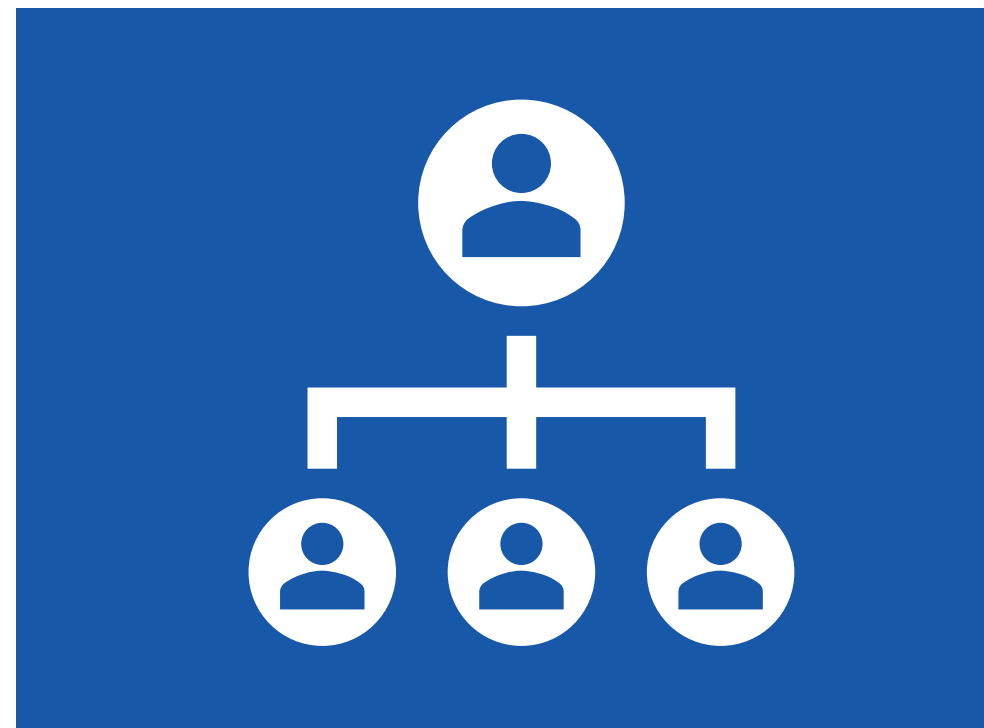
# Mettre en oeuvre un changement

## PERSONNES



Formation ou  
embauches pour avoir  
les compétences

## ORGANISATION



Nouvelles rôles et  
responsabilités

## PROCESSUS



Stratégie de  
déploiement et  
coordination de tâches

## TECHNOLOGIES

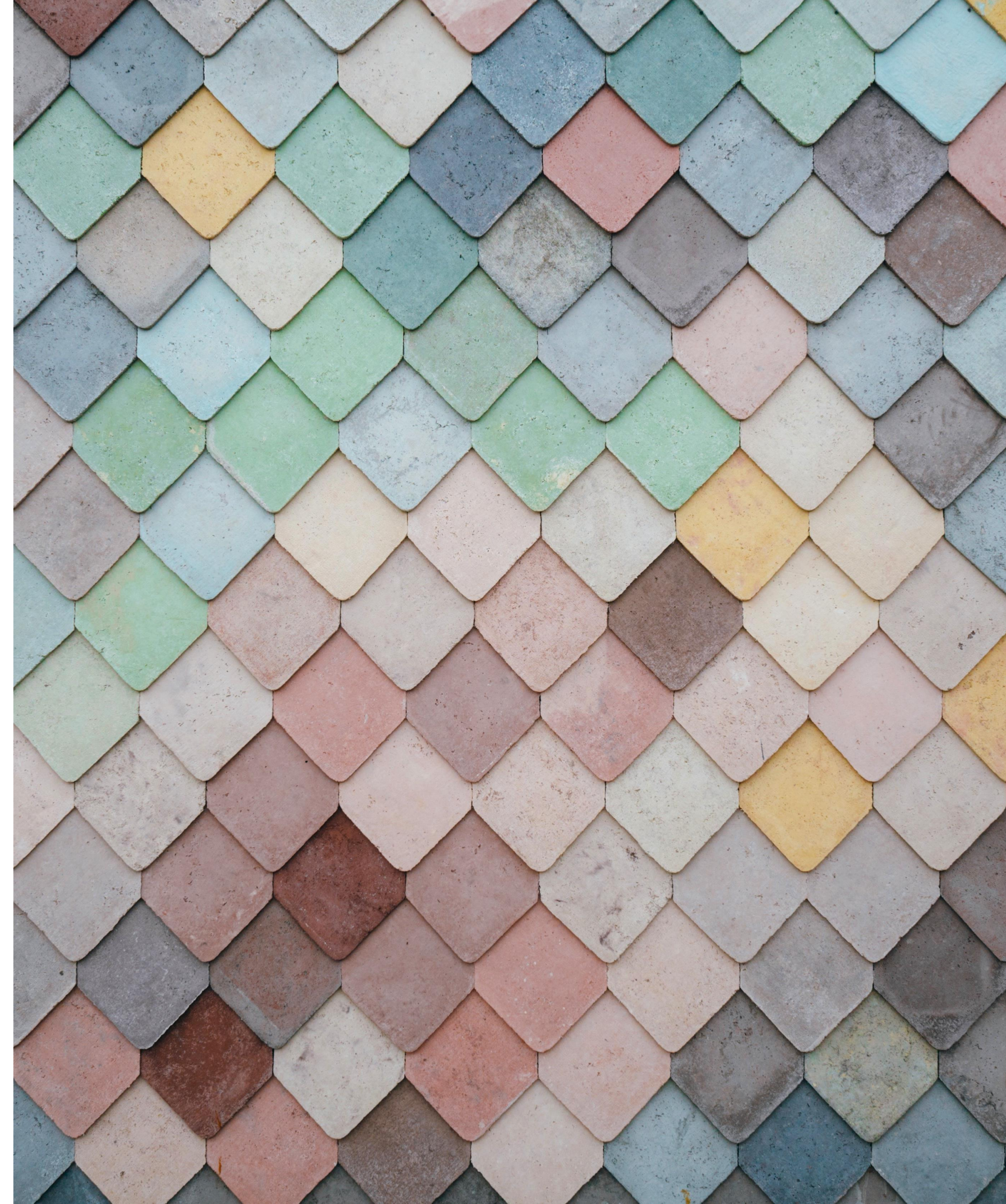


Outils et  
environnement  
technologique



# Convention de dénomination de fichiers

- Clarifier le contenu
- Structurer la gestion des fichiers
- Rendre facile la recherche
- Exemple :
  - Logiciel\_Métier\_UsageBIM\_But\_Version
  - « C3D\_VRD\_Documentation\_CreationProfilsTravers\_V1 »





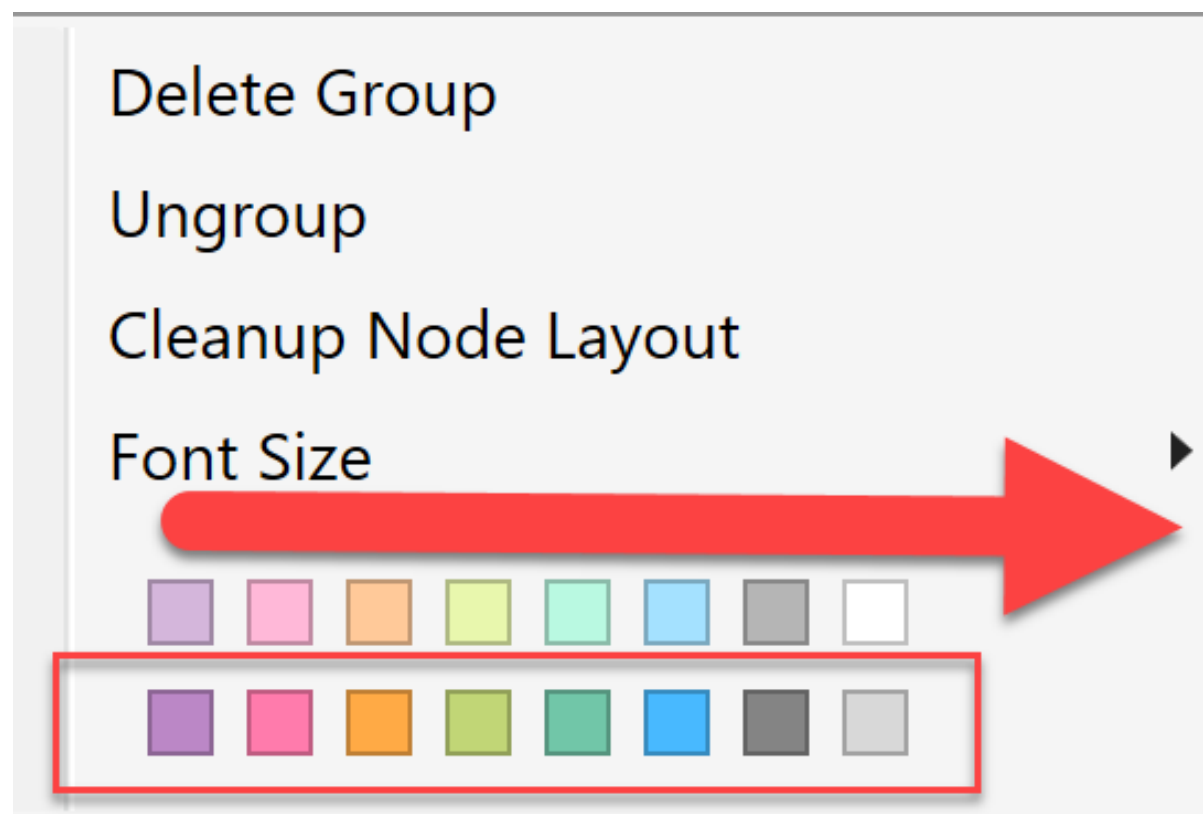
# Gabarit de référence

- Information générale
- Instructions
- Limitations





# Codage par couleur



## GET

Obtention des valeurs de paramètres d'objets Civil 3D

## INPUT

Données d'entrée et préparation

## FUNCTION

Manipulation des données sur des objets Dynamo

## OUTPUT

Création d'objets Civil 3D, résultat final

## SET

Définition des valeurs de paramètres sur des objets Civil 3D

## DEBUG

Noeuds utilisés pour débbugger la macro

## WIP

Notes non considérées comme faisant partie de la version finale



# NOTES

## INFORMATIONS

Copyright 2017 Autodesk, Inc. Tous droit réservés.  
Société : <société>  
Bureaux : <bureau>

Version : 1.0.0  
Auteur : <auteur>  
paolo.serra@autodesk.com

Mots-clés: [MOTS\_CLES]

Testé pour :  
Dynamo : X.X.X  
Civil3D : 20XX

## INSTRUCTIONS

<DESCRIPTIONS>

<INSTRUCTIONS>

## PROBLEMES CONNUS ET LIMITES

<PROBLEMES>

<LIMITES>

## GUIDE

Lisez les instructions ci-dessous.

Ajouter des Notes et Commentaires dans le graphe.

Utiliser les Groupes de Noeuds et leur Code Couleur.

Renommer les noeuds : <NomOriginal> |  
<Description>.

Définir des note pour les Entrées (IN) et Résultats (OUT) des noeuds Python.

Favoriser l'utilisation des structures de noeuds simples et répétables.  
Une macro Simple est meilleure qu'une macro Complexe.

Une macro Complexe est meilleure qu'une macro Compliquée.

La Lisibilité est importante.

Aucun cas n'est suffisamment spécifique pour justifier de ne pas respecter les règles.

Il ne faut pas sacrifier l'aspect Pratique à la Pureté.

Aucune Erreur ne doit être transparente.

Ou alors cela doit être fait consciemment.

Il ne faut jamais présumer face à une ambiguïté.

Il y a toujours une manière évidente de faire les choses.

Maintenant ou jamais.

Mais plutôt jamais que tout de suite.

Evaluation: XX

### Liens Utiles | CTRL + Cliquez sur les liens ci-dessous

"[http://dynamoprimer.com/en/10\\_Packages/10-Packages.html](http://dynamoprimer.com/en/10_Packages/10-Packages.html)"; >  
"[http://dynamoprimer.com/en/12\\_Best-Practice/12-1\\_Introduction.html](http://dynamoprimer.com/en/12_Best-Practice/12-1_Introduction.html)"; >

This Python Script creates a CSV file to track the usage of the Dynamo file.

### Python Script | Tracker

+ - OUT

## GET

Obtention des valeurs de paramètres d'objets Civil 3D

## INPUT

Données d'entrée et préparation

## FUNCTION

Manipulation des données sur des objets Dynamo

## OUTPUT

Création d'objets Civil 3D, résultat final

## SET

Définition des valeurs de paramètres sur des objets Civil 3D

## DEBUG

Noeuds utilisés pour débbugger la macro

## WIP

Notes non considérées comme faisant partie de la version finale



# Stockage des fichiers

- Centralisé
- Différentiation entre la source et le fichier Dynamo utilisé pour le projet (**BINDING**)





# Testing et validation

- Validation par l'utilisateur final
- Process interactif
- Intégrer du feedback pertinent





# Suivi de l'utilisation

- **Mesurer** l'utilisation :
  - Savoir quels sont les graphes les plus pertinents
  - Calculer le ROI sur la **VALEUR**
- Au consulting, nous avons développé un outil pour le faire





# Communication

- Communiquer les réussites
- Privilégier la vidéo
- « Story telling »
  - Description de la problématique
  - Comment elle a été résolue
  - Donner des preuves sur la valeur apportée:





# Groupe d'utilisateurs

- Super utilisateurs
- « Parlent métier »
- Facilitent la communication et adoption





# Ressources

- Besoin des ressources spécialisées :
  - Formation de ressources
  - Acquisition du talent
  - Accompagnement pour les premiers cas





# Références

- Primer
- Dynamo Forum – Civil 3D
- Cours AU précédents Dynamo - Civil 3D :
  - [Computational Design for Civil Engineers](#) | Paolo Serra et Safi Hage
  - [Dynamo in Civil 3D Introduction–Unlocking the Mystery of Scripting](#) | Jowenn Lua et Andrew Milford
  - [Accesorize your Design – Dynamo for Civil 3D](#) | Stacey Morkin et Dylan Kahle
  - [Generating, Transforming, and Analyzing Railway Design Data in Civil 3D and Dynamo](#) | Wouter Bulens et Steve Crokaert
- Cours AU 2020 (pas de lien disponible) :
  - Generative Design | Raquel Bascones et Paolo Serra
  - Civil 3D Toolkit | Jowenn Lua





# Objectifs d'apprentissage



## IDENTIFIER LES OPPORTUNITÉS D'AUTOMATISATION

- L'automatisation pour faire plus, mieux et avec moins de ressources
- Cas d'usage : tâches répétitives, tâches avec peu de valeur ajoutée, processus itératifs, etc.
- Se focaliser sur les « quick wins »



## DÉVELOPPER DES SOLUTIONS INDUSTRIALISABLES

- Faire de solutions que puissent être utilisées avec le Dynamo Player
- Imbriquer les graphes avec des notes
- Modulariser les solutions pour tester et débbuger plus facilement



## PENSER ANALYTIQUEMENT

- Décrire la séquence logique pour résoudre la problématique
- Créer un « POC » rapidement, utiliser de datasets génériques
- Ne pas changer l'intervalle de géométrie et avoir en tête le binding et l'utilisation des objets temporaires



## IMPLÉMENTER L'AUTOMATISATION

- Gabarits avec des codes couleurs, mesurer l'utilisation des graphes, suivre une convention de nommage
- Communiquer en montrant la valeur gagnée, s'appuyer sur le terrain pour communiquer
- Mettre en place une politique de formation ou d'acquisition du talent



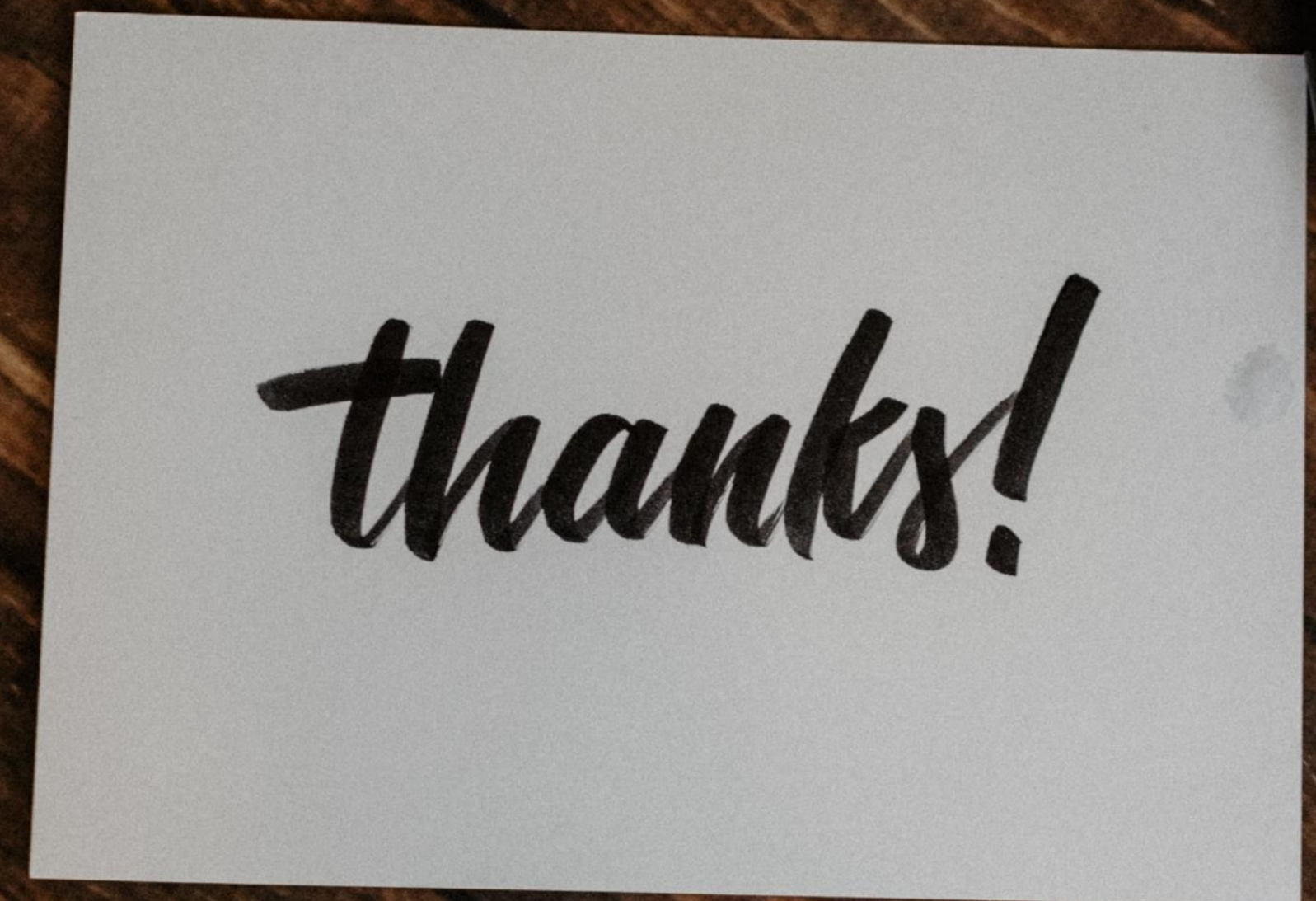
# Remerciements

POUR L'AUTORISATION DE PARTAGER AVEC VOUS CE  
CAS PRATIQUE :

L'équipe de Jacobs et spécialement à Duncan TAYLOR

POUR LEUR AIDE ET CONSEIL TECHNIQUE :

Paolo SERRA, Jowenn LUA et Philippe BONNEAU





**L'automatisation n'est pas aussi effrayante  
que cela puisse paraître, elle  
peut être vraiment simple.**

**Pensez aux choses qui vous frustrant ou qui  
ne nécessitent pas de réfléchir, cela pourrait  
vous faire gagner beaucoup de temps.**

Sol AMOUR, Product Manager Dynamo





Autodesk and the Autodesk logo are registered trademarks or trademarks of Autodesk, Inc., and/or its subsidiaries and/or affiliates in the USA and/or other countries. All other brand names, product names, or trademarks belong to their respective holders. Autodesk reserves the right to alter product and services offerings, and specifications and pricing at any time without notice, and is not responsible for typographical or graphical errors that may appear in this document.

© 2020 Autodesk. All rights reserved.