

CES471877

Como implementar la automatización en sus proyectos de diseño

David LICONA
Autodesk

Objetivos de aprendizaje

- Identificar las oportunidades de automatización en los procesos actuales
- Pensar lógicamente para desarrollar soluciones de automatización
- Crear soluciones escalables que puedan aplicarse en todos los proyectos
- Implementar la automatización dentro de una organización

Descripción

La industria de la infraestructura está rezagada de otras industrias en cuanto al uso de la automatización. Dynamo para Civil 3D ayuda a abordar este problema, pero muchas empresas tienen dificultades para implementar la automatización en sus equipos. ¿Qué procesos deben automatizarse? ¿Cómo llevar a cabo tal cambio en un equipo? La presentación se centra en cómo implementar la automatización en procesos existentes: identificar las oportunidades de automatización, desarrollar la solución y cómo ponerla en práctica en los diferentes equipos dentro de una organización. Utilizaremos un ejemplo de un proceso común en ingeniería civil como el movimiento de tierras e ilustraremos los pasos para desarrollar e implementar soluciones escalables que permitan automatizar el proceso de diseño. Finalmente, hablaremos de las mejores prácticas para implementar la automatización exitosamente: estandarización, organización de documentos, requisitos tecnológicos, recursos necesarios, etc.

Conferencista

Guatemalteco de nacimiento y francés de adopción desde hace poco. Soy un ingeniero civil con varios años de experiencia en el diseño, construcción y operación de infraestructuras lineales. Tuve la suerte de poder participar en diferentes fases de proyectos (factibilidad, diseño, construcción) y en roles diversos (propietario, diseño y supervisión, constructor). Sin embargo, durante todas estas experiencias, por muy diferentes que fueran, siempre busqué hacerme la vida más fácil automatizando mis tareas cotidianas, con los medios que tenía en ese momento. Esto es porque no puedo soportar perder mi tiempo con tareas inútiles o sin un valor añadido y es lo que me motivó para realizar esta presentación. Hoy, tengo la suerte de que mi trabajo consiste a precisamente a ayudar a los clientes de Autodesk a aumentar el valor que obtienen de nuestro software y hacer sus procesos de diseño más fáciles, para concentrarse en las funciones que proveen valor realmente.

Introducción

El interés de usar Dynamo para automatizar es que es accesible a todos los usuarios de Civil3D, sean programadores o no. Dynamo permite al usuario programar visualmente los procesos que se utilizan para el procesamiento de datos o el diseño de geometría, desarrollar modelos paramétricos, así como ampliar las capacidades de Civil 3D.

Dynamo puede ayudar a reducir el esfuerzo de realizar tareas repetitivas, aumentar la coherencia en los procesos de diseño, reducir errores, aumentar la productividad y ampliar las capacidades de los equipos de proyectos.

El propósito de esta presentación es describir las mejores prácticas para identificar casos para automatizar, desarrollar soluciones escalables e implementar Dynamo dentro de una empresa, más allá de un proyecto único o un solo equipo. Los diferentes temas serán ilustrados gracias a la utilización de un caso práctico.

Limitaciones de este documento

Este documento es una guía de consejos prácticos para implementar la automatización en una organización. No es una guía de formación de Dynamo para Civil 3D.

La intención es proporcionar información sobre puntos relevantes de los aspectos de Dynamo y su utilización con Civil 3D y que han sido cubiertos por los consultores de Autodesk en varios talleres y presentaciones para diversos clientes.

Dynamo Foundation

La información proporcionada aquí es parte de una oferta creada por Autodesk Consulting: Dynamo Foundation.

Esta se basa en la experiencia de varios consultores en todo el mundo y en todo tipo de organización. Está compuesta de las mejores prácticas recopiladas a través de estas experiencias. El propósito de esta oferta es ayudar a los clientes con los casos de automatización más complejos y garantizar la implementación y adopción de la automatización mediante Dynamo.

Identificar las oportunidades de automatización

Antes de lanzarse a la implementación de Dynamo, es necesario definir el marco general del que hablamos cuando hablamos de automatización.

En pocas palabras, podemos definir la automatización como "La reducción de la intervención humana para realizar tareas".

En la práctica, podemos ver la automatización en muchos aspectos de nuestro alrededor, obviamente con diversos grados de automatización. Creo que todos hemos estado en una situación en la que nos enfrentamos a un archivo de Excel con una macro excelente, que nos permite formatear el archivo, extraer la información que necesitamos e incluso imprimir el archivo. Todo al utilizar un solo botón.

Muchos de los programas desarrollados en Dynamo funcionan como una macro, solo que esta vez aplicada a nuestros diseños de infraestructura. Un botón para crear puntos a lo largo de una carretera, un botón para exportar los property sets de nuestro modelo a Excel, un botón para diseñar nuestras plataformas de movimiento de tierras, etc.

Todos estos son ejemplos de lo que podemos lograr con Dynamo para Civil 3D.

Programación Visual

La programación visual es un tipo de programación en la cual los programas (código) se "escriben" de manera gráfica. No es necesario conocer un idioma de programación en particular o una sintaxis en particular. Solo se necesita conocer el software o la interfaz que estamos utilizando para realizar la programación visual.



Figura 1 : Programación Visual (fuente: Fotos por [Daniel Tuttle](#) y [Vitae London](#))

Si observamos la imagen anterior, podemos hacer una analogía con un piano. La programación visual es como las teclas del piano. En la caja del piano, invisible a nuestros ojos, encontramos todo lo que nos permite realizar la programación visual (el código actual, API, etc.). Aunque no conocemos esto, sabemos que, si tocamos una tecla en particular, vamos a escuchar su sonido.

Dynamo para Civil 3D

En Dynamo, tenemos nodos que representan operaciones (como las teclas de nuestro ejemplo) y que vamos a unir los unos a los otros para realizar flujos de trabajo que nos permitan crear objetos automatizados (entre otras cosas).

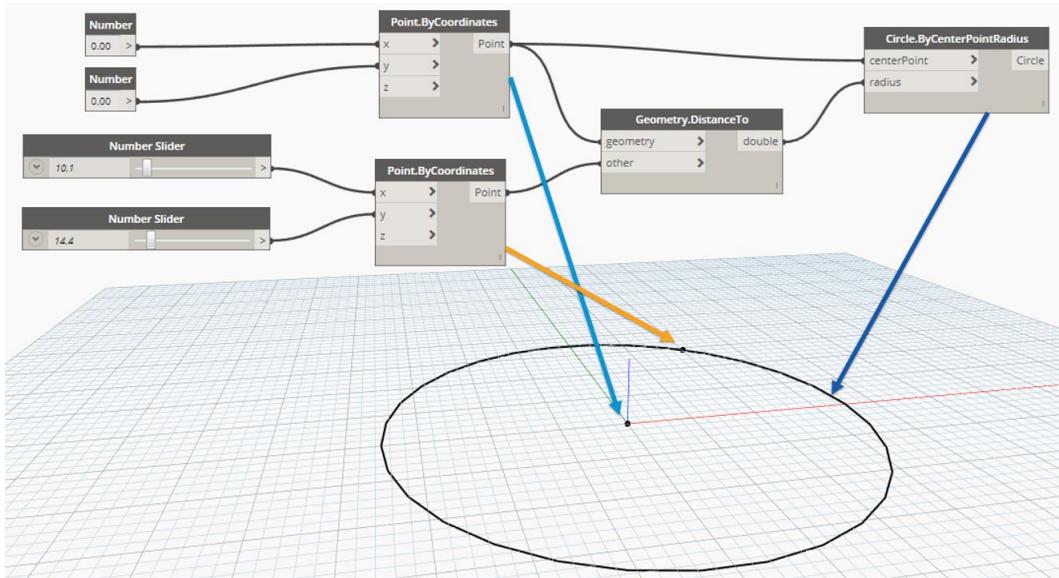


Figura 2 : Programación visual en Dynamo

Este documento no es un curso de Dynamo y para comenzar a aprender Dynamo hay varias fuentes de información como el Dynamo Primer. Al final de este documento, pueden encontrar fuentes recomendadas para obtener más información sobre como aprender a utilizar Dynamo.

Por lo tanto, voy a asumir que el lector de las siguientes páginas conoce un mínimo de Civil 3D y Dynamo para Civil 3D y está aquí para saber cómo ser más activo en el uso de Dynamo, cómo aprovechar esta herramienta para automatizar sus procesos de diseño y sobre todo cómo implementar la automatización en su organización, más allá del proyecto o el equipo único.

Objetivo de la automatización

La automatización permite:

- Llevar a cabo **más** estudios o estudios **más** complejos;
- Trabajar **mejor**, evitando errores humanos y mejorando la calidad de los estudios;
- Con **menos** recursos y **menos** tiempo.

Siendo más concretos en cuanto a los casos prácticos y para empezar a imaginar cómo utilizar la automatización para mejorar los procesos, esta ayuda a:

- Automatizar tareas repetitivas;
- Crear flujos de trabajo complejos, conectando varias fuentes de datos;
- Explorar alternativas de diseño rápidamente;
- Acceder a los datos ocultos del modelo;
- Crear y analizar formas geométricas complejas.

Dynamo es particularmente eficaz para la creación de formas geométricas complejas, gracias al gran contenido de nodos que posee en su biblioteca de geometría.

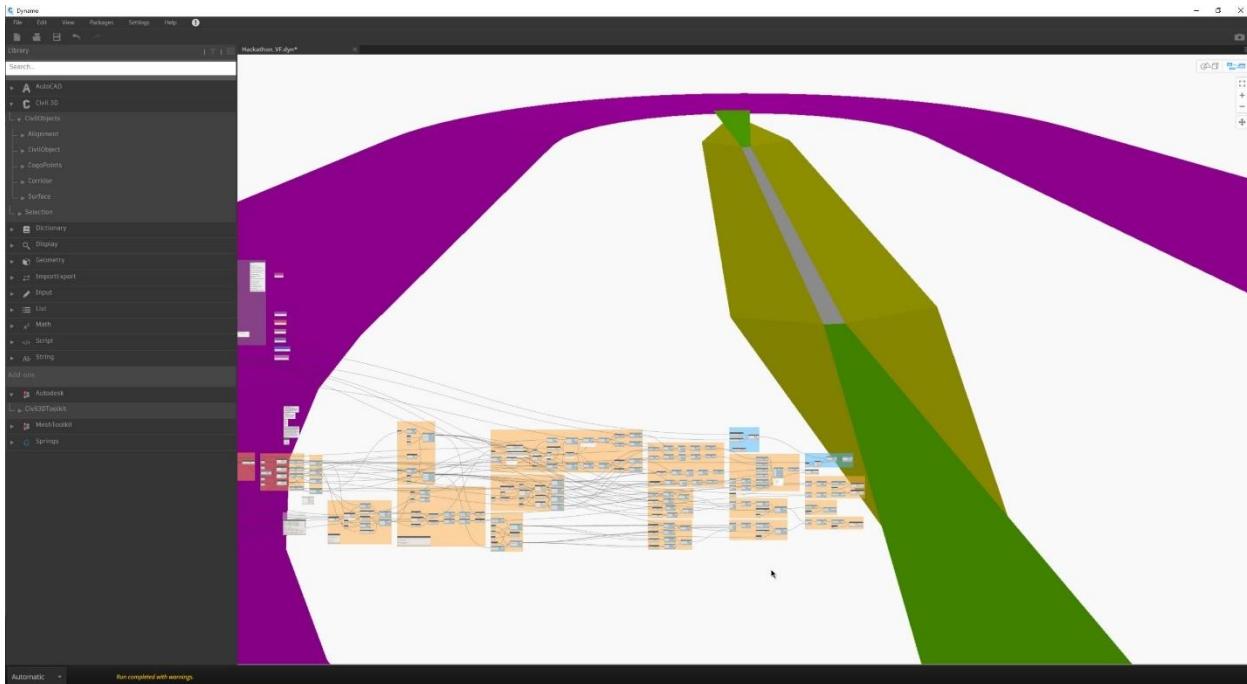


Figura 3 : Ejemplo de la utilización de Dynamo para crear soluciones geométricas complejas

Finalmente, quiero señalar que la automatización puede beneficiar a todo tipo de organización, sin importar su tamaño:

- Empresas pequeñas: Pueden competir con las más grandes gracias a la automatización;
- Empresas medianas: Pueden reducir los costos internamente y reinvertir en la innovación;
- Empresas grandes: Pueden implementar las mejores prácticas de diseño de una manera más eficaz.

Identificación de oportunidades de automatización

¿Cómo identificar las tareas o procesos que se pueden ser susceptibles a ser automatizados?

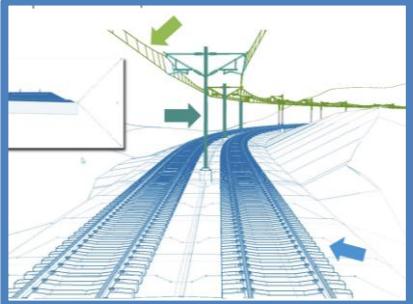
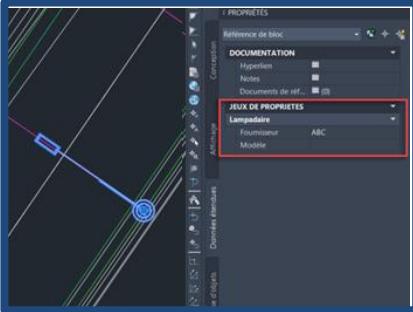
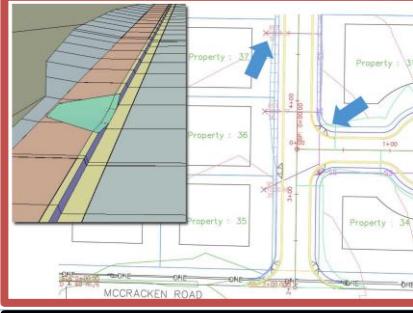
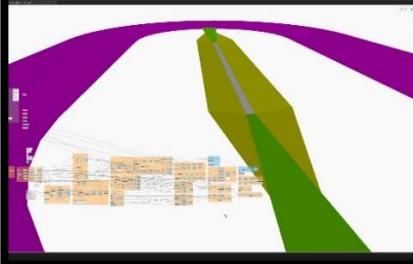
Lo mejor, es comenzar por hacerse ciertas preguntas:

- ¿Esta tarea/proceso es repetitiva? "
- ¿No necesito pensar mucho para poderlo realizar?
- ¿Es esta una tarea en la gran mayoría de mis proyectos?
- ¿Paso demasiado tiempo para hacerla manualmente?
- ¿Es esta una tarea particularmente susceptible a errores humanos?

Si la respuesta es sí a alguna de estas preguntas, entonces es un caso que vale la pena de explorar. A menudo, la respuesta es sí a más de una de estas preguntas: Una tarea/proceso demasiado repetitivo es muy probable que genere errores humanos.

Otra forma sencilla de identificar casos para automatizar es simplemente pensar en aquellas tareas que nos frustran a diario. A menudo, estas tareas frustrantes son repetitivas con poco valor añadido y, por tanto, ideales para automatizar.

Tenemos acceso a casos prácticos, muy bien documentados en los [ejemplos de Dynamo](#) proporcionados con la instalación de Civil 3D o en los webinars disponibles en [youtube](#) y en los cuales podemos encontrar algunas de las características descritas anteriormente:

Tarea / Proceso repetitivo	Insertar bloques a lo largo de un proyecto lineal (durmientes, postes de luz, catenarias)	
Tarea / proceso a realizar de manera sistemática	Aplicación de los property sets	
Tarea / proceso que toma mucho tiempo	Modelización de rampas para aceras	
Tarea muy compleja	Superficies imaginarias para el diseño aeroportuario	

Priorizar las ideas

Una vez definidos los casos susceptibles a automatizar, hay que priorizarlos. Para esto, propongo utilizar una matriz que utilizaba a menudo cuando era gerente de proyectos de infraestructura ferroviaria. Esta consiste a dar una nota a cada uno de los casos identificados, basada en dos criterios:

- La facilidad para automatizar el proceso;
- El impacto potencial de la automatización.

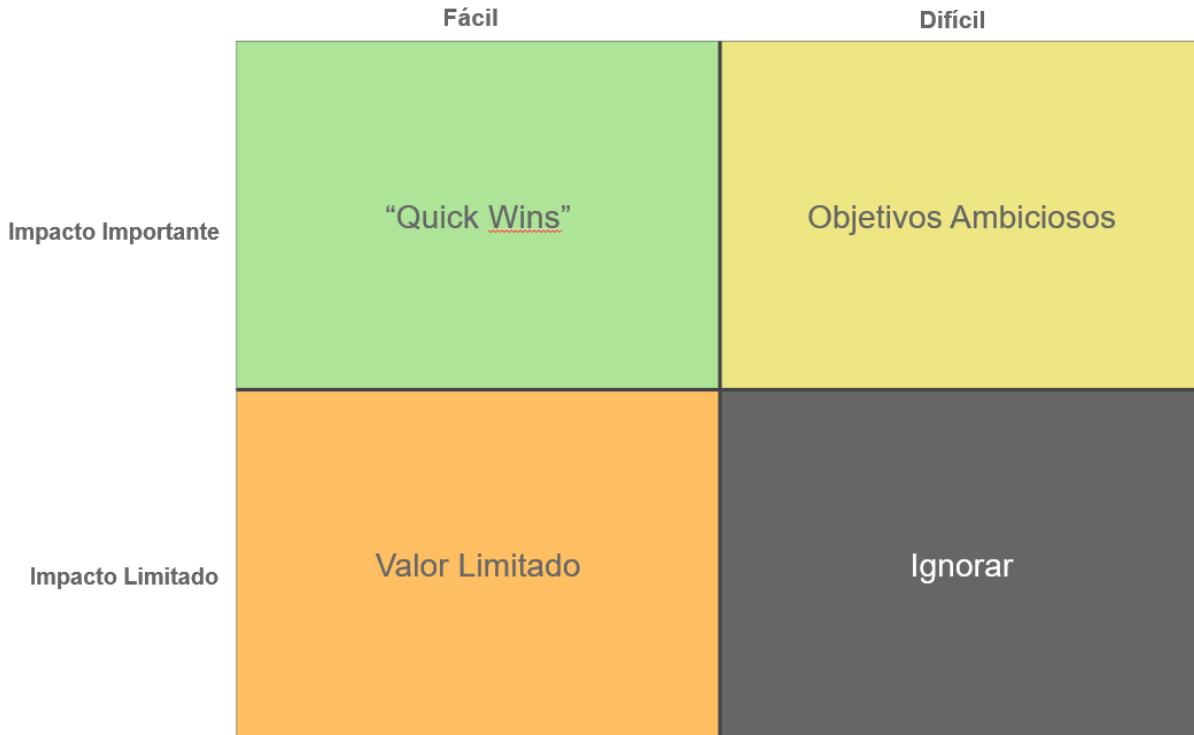


Figura 4 : Matriz propuesta para priorizar los posibles casos de automatización

Por ejemplo, si una idea es muy fácil de lograr, pero impacta un solo proyecto, se trata de un caso nicho y entonces es una idea de valor limitado.

Al principio, debemos concentrarnos en los "Quick Wins": casos de impacto importante en la organización y que se pueden solucionar con un esfuerzo limitado. Es muy importante tener resultados visibles rápidamente para facilitar la adopción de la automatización.

Por supuesto, para juzgar si un caso es fácil o difícil de realizar, depende de la experiencia de cada uno en el manejo de Dynamo, Civil 3D y los diferentes lenguajes de programación que podemos utilizar dentro de Dynamo.

La base para abrir otras posibilidades de automatización

La programación visual para automatizar procesos es una etapa en un camino que lleva a casos de automatización aún más poderosos. Podemos referirnos al "Diseño Generativo" como el siguiente paso, donde podemos automatizar la generación de una multitud de variantes para evaluar una solución y obtener así una solución optimizada.



Figura 5 : Del Diseño Paramétrico a Machine Learning (fuente : foto ML [AltumCode](#) - [Unsplash](#))

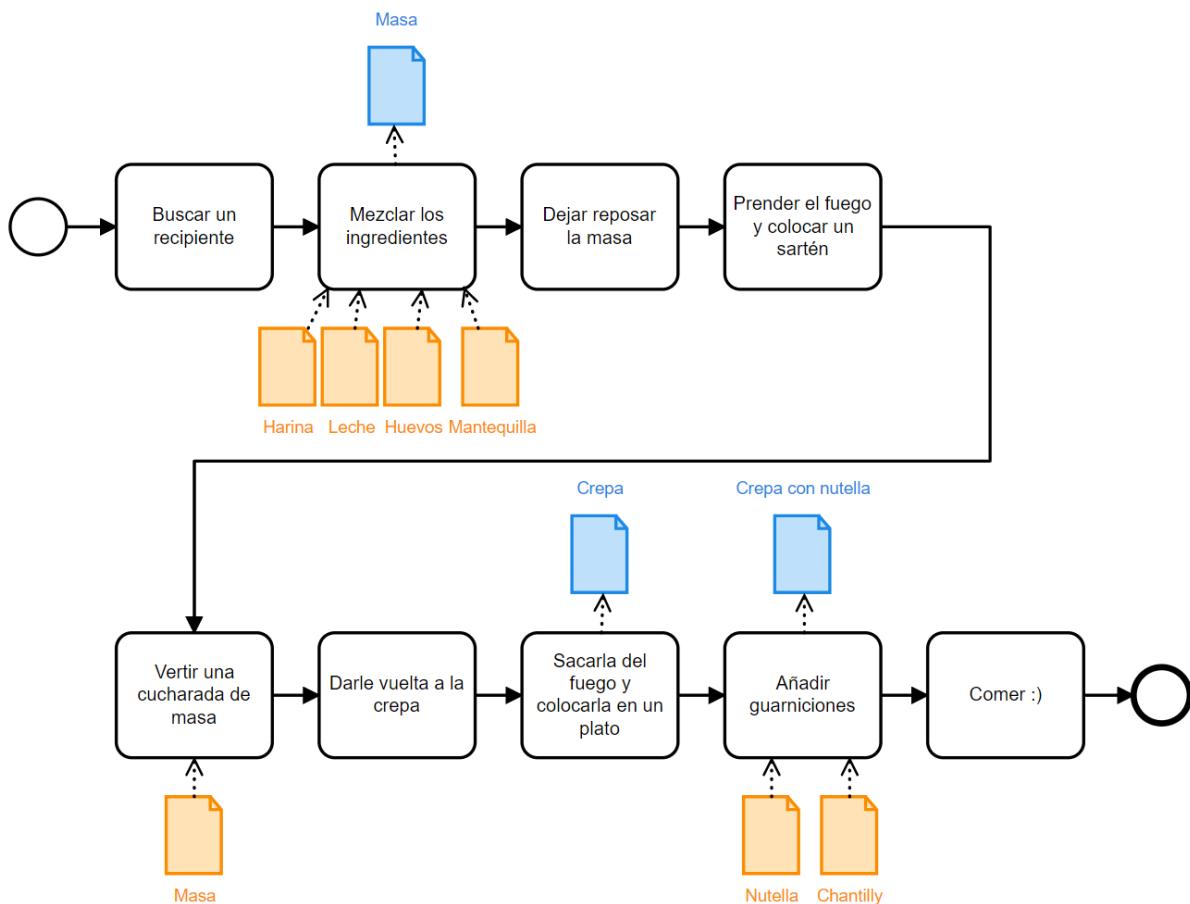
En conclusión

- La automatización nos ayuda a hacer más, mejor y con menos recursos;
- Existen varios casos prácticos de uso, para identificarlos es necesario observar si hay procesos actuales que sean muy repetitivos, sistémicos, iterativos o de bajo valor añadido;
- Una vez que se han identificado los casos a automatizar, hay que priorizarlos. Es necesario concentrarse en los "Quick Wins", los casos con alto impacto y mínimo esfuerzo de realización;
- No hay que tener miedo de lanzarse en un proceso de automatización.

Pensar lógicamente para crear soluciones de automatización

Cuando llegué a Francia, una de las primeras cosas que aprendí a cocinar fueron las crepas.

Podría escribir la receta como una secuencia lógica de tareas que describen los pasos para llegar a la meta (¡las crepas!). En la imagen acá abajo, los elementos en amarillo representan las entradas requeridas para completar una tarea (ingredientes) y los elementos en azul representan las salidas que trae cada tarea (resultado intermedio).



Escribir las etapas a seguir para poder resolver un problema, antes de tratar de resolverlo, es fundamental. Esto permite:

- visualizar el problema como una secuencia de tareas lógicas a seguir;
- Darse cuenta fácilmente si nos hemos saltado un paso;
- Verificar si los resultados son los que esperábamos;
- Si es posible de obtener todas las entradas que necesitamos.

Podemos darnos cuenta muy rápidamente si estamos saltando etapas, o si las salidas esperadas no se corresponden con las entradas necesarias.

Herramientas

Para construir este tipo de diagrama, recomiendo usar [Cawemo](#). Una herramienta en línea, gratuita, muy fácil de usar, un poco limitada dada su sencillez pero que aporta una gran ventaja: permite construir diagramas de forma colaborativa, en los que varias personas pueden realizar modificaciones al mismo tiempo.

Desarrollo de las soluciones

Al construir gráficos Dynamo, debemos tener en cuenta que necesitamos gráficas que sean:

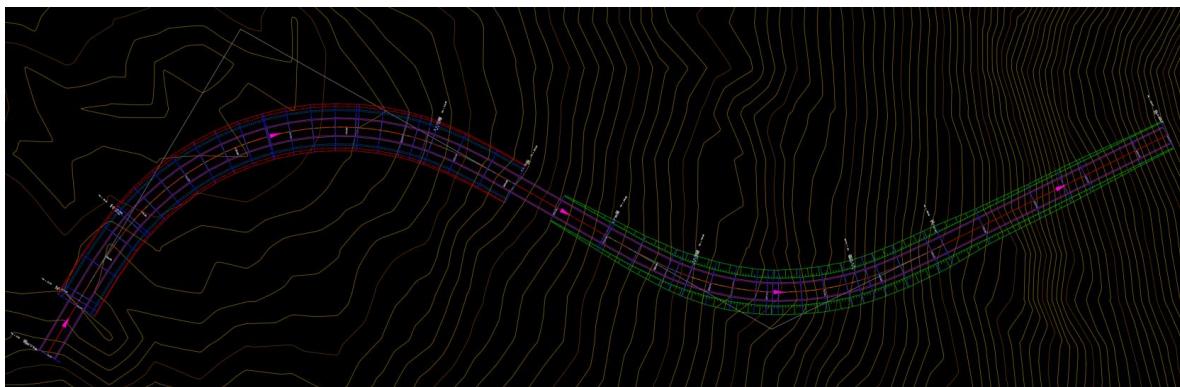
- Eficientes;
- Genéricos;
- Escalables

Sin embargo, lo más importante es que queremos gráficas que funcionen. Por tanto, recomiendo realizar una "Prueba de concepto (POC)" antes de lanzarse a crear gráficas ambiciosas. Cabe decir que a los programas en Dynamo los llamamos gráficas o scripts.

Empezar por un POC nos permite saber rápidamente si lo que queremos hacer es realizable. Luego podemos modificar la gráfica para que tenga las características mencionadas anteriormente.

Dataset Genéricos

Para realizar pruebas y crear POCs, utilizo un modelo genérico y simple. La imagen de abajo muestra justo un modelo de una carretera. No es representativo de la realidad ni detallado, pero permite que la gráfica Dynamo se ejecute mucho más rápido.



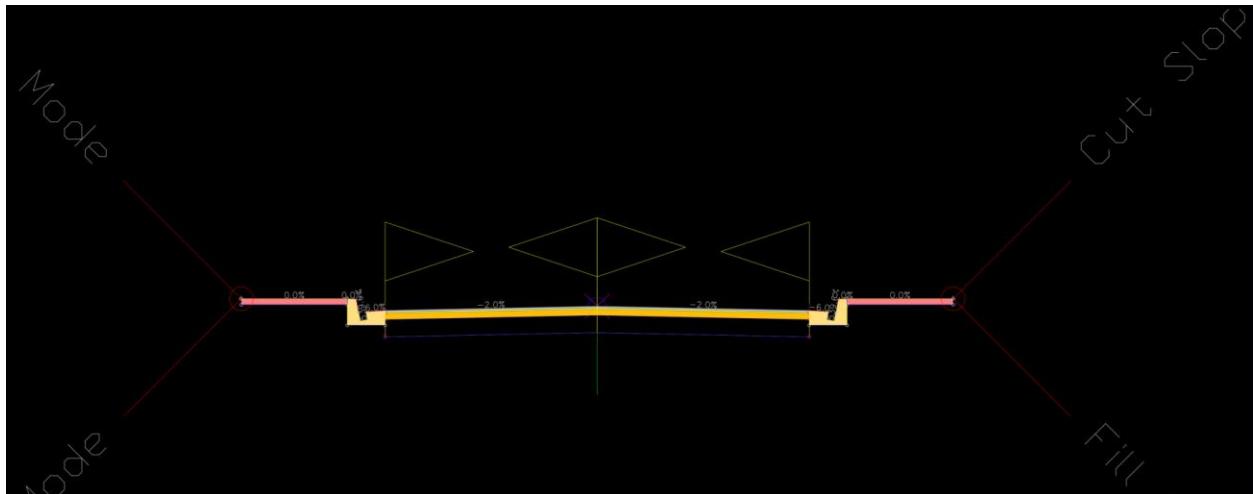


Figura 6 : Dataset genérico

También he encontrado que los modelos de ejemplo que vienen con la instalación de Civil 3D son un buen punto de partida para realizar pruebas. Puede encontrar estos archivos [aquí](#).

Algunos claves de Dynamo

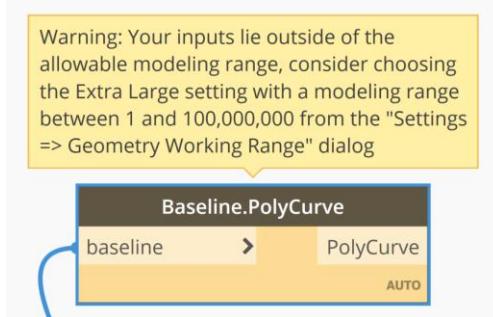
Aunque este curso es para personas que ya conocen Dynamo, es importante tener en cuenta algunos conceptos al crear las gráficas de Dynamo.

Intervalo de Geometría

Una de las advertencias que encontramos con más frecuencia en Dynamo se refiere al intervalo de geometría. Esta advertencia aparece cuando estamos trabajando lejos del 0,0,0 en Dynamo y por lo tanto siempre aparece cuando estamos trabajando con modelos geolocalizados.

Normalmente, esta advertencia no altera el resultado y el gráfico se ejecuta sin problemas. Generalmente, se recomienda que NUNCA se cambie el intervalo de geometría, ya que la precisión será reducida y corremos el riesgo de perder datos.

Sin embargo, cuando usamos el Dynamo Player (hablaremos de esto más adelante) para ejecutar una gráfica, el Dynamo Player va a decir que la gráfica se ejecutó con errores y esto puede ser problemático para los usuarios que son nuevos en Dynamo. Pueden perder la confianza en el resultado. Para eliminar estas advertencias, se puede utilizar el nodo "Function Apply":



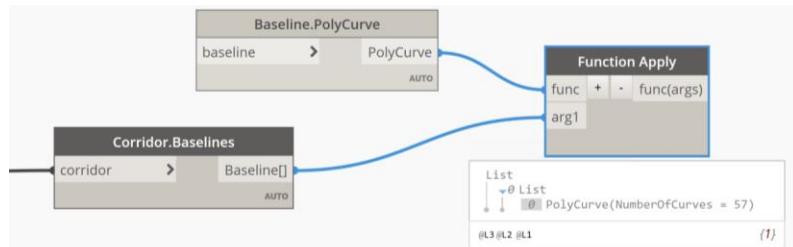


Figura 7 : Utilización del nodo “Function Apply” para eliminar las advertencias del intervalo de geometría

Sin embargo, en ciertos casos, puede ser que la modelización no funcione con el intervalo de geometría en “medium” y estando tan lejos del 0,0,0 de Dynamo. En estos casos, será necesario enviar la representación del modelo en Dynamo cerca del 0,0,0 para realizar las operaciones necesarias. Aquí hay un [hilo del foro sobre este tema](#).

Binding

Dynamo creará un vínculo entre la gráfica y los elementos creados por ella en el modelo (Civil 3D). Esto significa que Dynamo recordará cuáles fueron los elementos creados por él. Si ejecutamos una gráfica más de una vez en el mismo modelo, borrará los elementos creados en ejecuciones anteriores de esta gráfica. Esto es muy importante a tener en cuenta para la colocación de bloques (postes de luz, por ejemplo). Por lo tanto, deben de colocarse todos los bloques a la vez o tener gráficas diferentes para hacerlo. Para obtener más información, puede visitar esta [discusión](#) en el foro de Dynamo.

Objetos provisionales

Algunos nodos que interactúan directamente con objetos de Civil 3D y objetos de AutoCAD podrán crear los objetos directamente en el modelo. A veces estos nodos son parte solamente de una operación intermedia en el proceso por lo cual no queremos conservar los objetos provisionales creados por ellos, ya que podemos contaminar el modelo original. Podemos utilizar los “Code Blocks” para recopilar todos los objetos temporales creados y eliminarlos utilizando el nodo “Object Delete”.

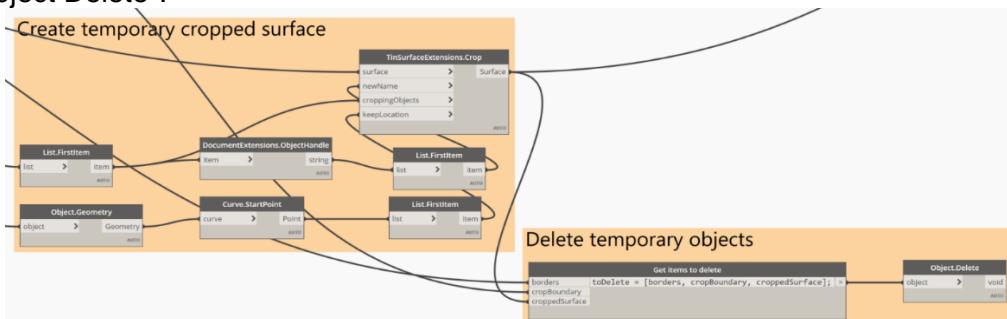


Figura 8 : Suprimir los objetos provisionales

En conclusión

- Hay que describir la secuencia lógica para resolver el problema para poder visualizar mejor el proceso e indicar las entradas y salidas esperadas;
- Debemos de crear "POC" rápidamente para demostrar si nuestra lógica funciona. Podemos usar modelos o datasets genéricos para ir más rápido;
- Al momento de construir gráficas en Dynamo, no hay que cambiar el intervalo de geometría y debemos tener en cuenta el “binding” y el uso de objetos provisionales.

Desarrollar soluciones escalables

En la medida de lo posible, queremos tener gráficas que funcionen con el Dynamo Player. Es decir, que se puedan ejecutar directamente desde Civil 3D, sin necesidad de entrar a Dynamo.

El Dynamo Player permite cambiar los datos de entrada, así como la configuración deseada. Gracias a esto, podemos por ejemplo probar varias soluciones rápidamente (colocando un bloque cada 25, 50 o 100 m). También permite declarar salidas o “outputs” de las gráficas para mostrar información al usuario si lo deseamos.

El Dynamo Player permite entonces explotar la lógica (gráfica) creada por otra persona. No es necesario conocer Dynamo sino solo estar consciente del propósito de la gráfica. Esto nos da una gran flexibilidad y es muy poderoso porque no se necesita capacitar a todas las personas en una empresa, sino solo a aquellas que construirán las gráficas. ¡Sin conocer el programa, todas las personas pueden beneficiarse potencialmente de Dynamo!

Instrucciones

Las gráficas construidas con Dynamo deben contar con cierta información para garantizar su uso correcto y especialmente en el caso de que sucedan problemas al momento de ejecución. Es mejor tener la información directamente en Dynamo que en un documento externo que se pueda perder fácilmente. La siguiente información debe incluirse en una gráfica:

- Propósito de la gráfica (su función);
- La versión;
- Las suposiciones hechas;
- Los datos de entrada esperados;
- Las optimizaciones posibles;
- Los problemas identificados;
- Los resultados que podemos esperar;
- Las pruebas realizadas (en cuales casos fue probada);
- El autor (a quien contactar en caso de problemas);

Modularizar

Finalmente, para poder contar con gráficas eficientes, genéricas y escalables, es recomendable tener gráficas que realicen una sola función. Es mejor tener 4 gráficas diferentes que tengan solo 1 función cada una, que 1 gráfica que realice 4 funciones.

Esto facilita la construcción de la solución, la realización de pruebas, y la comprensión de las gráficas entenderlas para un usuario nuevo y encontrar errores cuando surgen problemas.

En conclusión

- Crear soluciones que puedan ser ejecutadas directamente desde el Dynamo Player;
- Incluir instrucciones dentro de las gráficas;
- Modularizar las gráficas para que sea más fácil realizar pruebas y resolver los problemas en casos de errores.

Caso práctico

Vamos a usar como caso práctico un proyecto en el cual trabajé en este verano para poder visualizar mejor los conceptos presentados.

El objetivo del proyecto era automatizar la modelización de los movimientos de tierra de una carretera como objetos (sólidos 3D), para poder realizar BIM 4D y 5D.

Esto debido a que cuando realizamos el diseño de carreteras en Civil 3D, los movimientos de tierra no son modelizados como objetos.

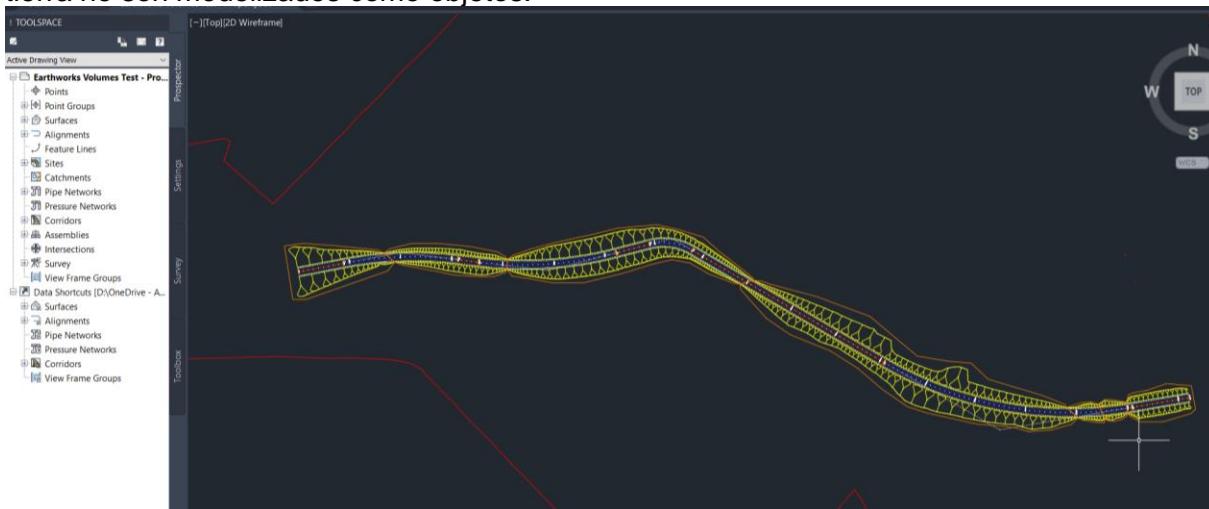


Figure 9 : Modelo Civil 3D provisto por el cliente

Si conocen Civil 3D, saben que existe una función para simplemente crear sólidos entre dos superficies. Sin embargo, hacer el proceso manualmente puede llevar mucho tiempo y, dependiendo de las condiciones del modelo, pueden surgir problemas. (por ejemplo, crear sólidos entre superficies no funciona con superficies formadas por más de 100K triángulos).

- De ahí la idea de crear un flujo de trabajo automatizado que hiciera todo esto con unos pocos clics. Por lo tanto, buscamos crear un script de Dynamo para: Modelizar los sólidos 3D representando los movimientos de tierra para cualquier proyecto de carretera;
- Dividir los sólidos:
 - Corte/Relleno;
 - A la mitad (en la línea de base);
 - Por cuerpo de tierra.
- Añadir los property sets y actualizar las propiedades:
 - Altura máxima del sólido;
 - Altura promedio del sólido.

Todo esto en solo 4 días para completar todas las tareas: reuniones con el cliente, desarrollo de la solución y documentación final.

Desglose del problema

Podemos traducir el problema en un flujo de trabajo de la siguiente manera:

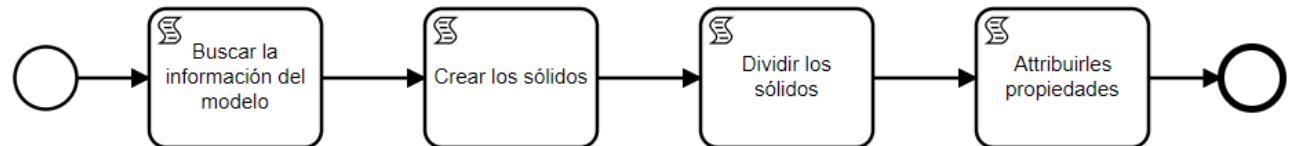


Figura 10 : Flujo de trabajo representando el script Dynamo a crear

Una vez que se tenga una idea del proceso, se puede continuar con:

- Identificación de las entradas;
- Identificación de las salidas;
- Y, sobre todo, **identificación de las etapas que no sabemos hacer.**

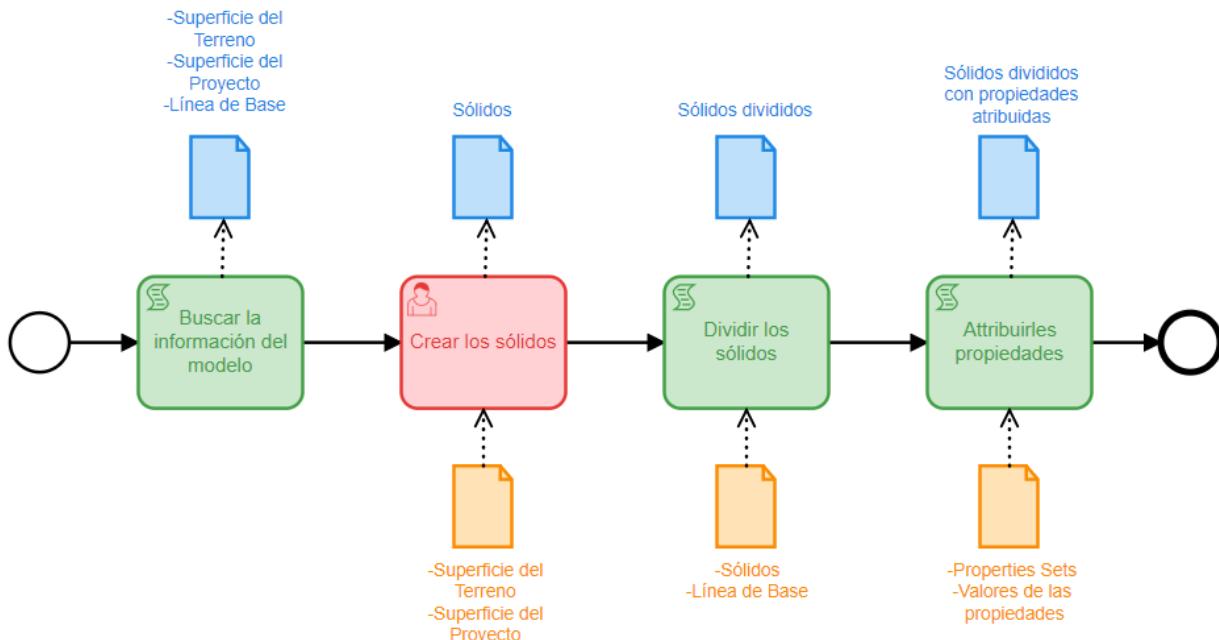


Figura 11 : Identificación de entradas/salidas y etapas complicadas

En ese momento, en Dynamo, no existían nodos para crear sólidos entre dos superficies. Así que hablé con mi colega Paolo Serra e identificamos tres posibilidades:

- Verificar si la funcionalidad para crear los sólidos entre dos superficies estaba disponible en la API de Civil 3D, para poder crear un nodo Dynamo que realizará esta operación,
 - Utilizar las líneas características del proyecto 3D para crear superficies y luego crear los sólidos;
 - Utilizar las secciones transversales para dibujar secciones y crear sólidos entre las secciones.
- La funcionalidad estaba disponible en la API de Civil 3D y Paolo pudo entonces crear un nodo para utilizar estas funcionalidades del programa. Hablaremos más adelante, pero esto muestra la importancia de contar con recursos especializados dentro de una organización.

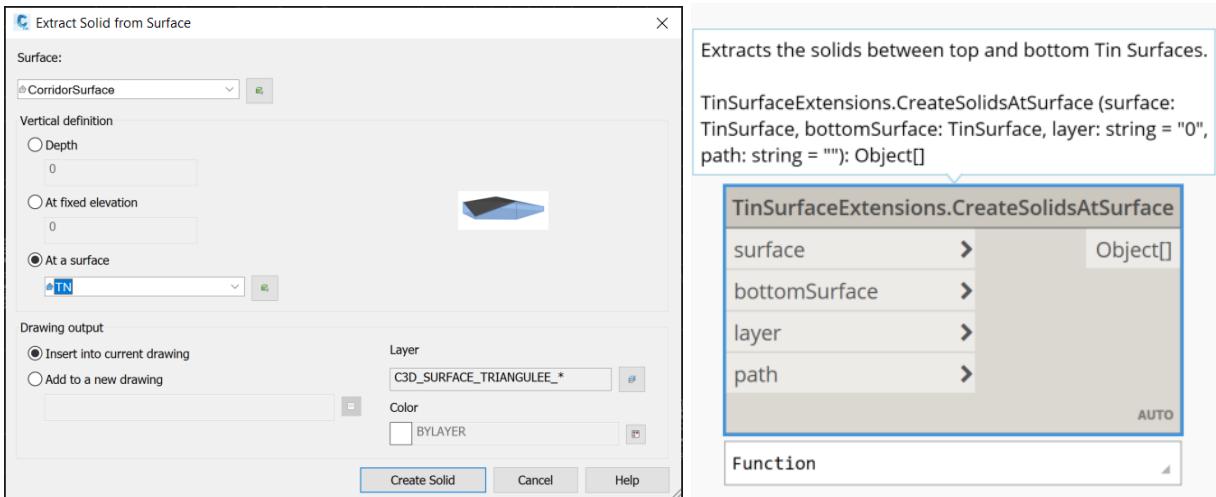
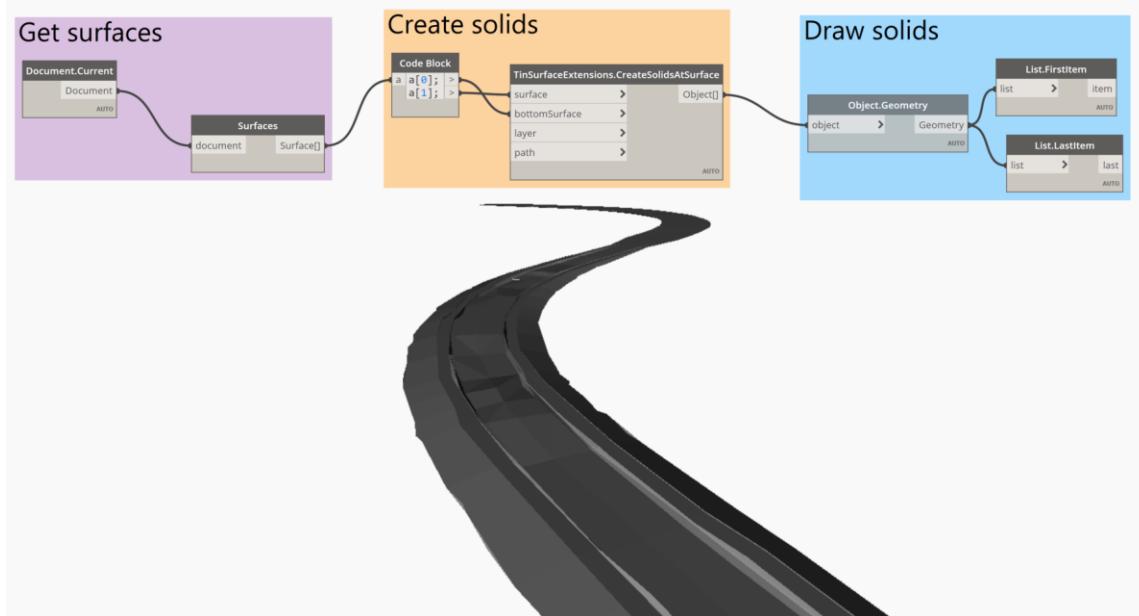


Figura 12 : Nodo desarrollado para crear sólidos entre dos superficies

Creación de un POC

El primer paso consiste en probar si las diferentes funcionalidades de la gráfica pueden ser creadas como previsto. La siguiente imagen muestra una prueba, muy rápida de realizar, que demuestra que la creación de sólidos entre dos superficies funciona (que es lo que necesitamos). Podemos también observar que la forma en que se recupera la información de Civil 3D no es ideal y debe mejorarse más adelante.



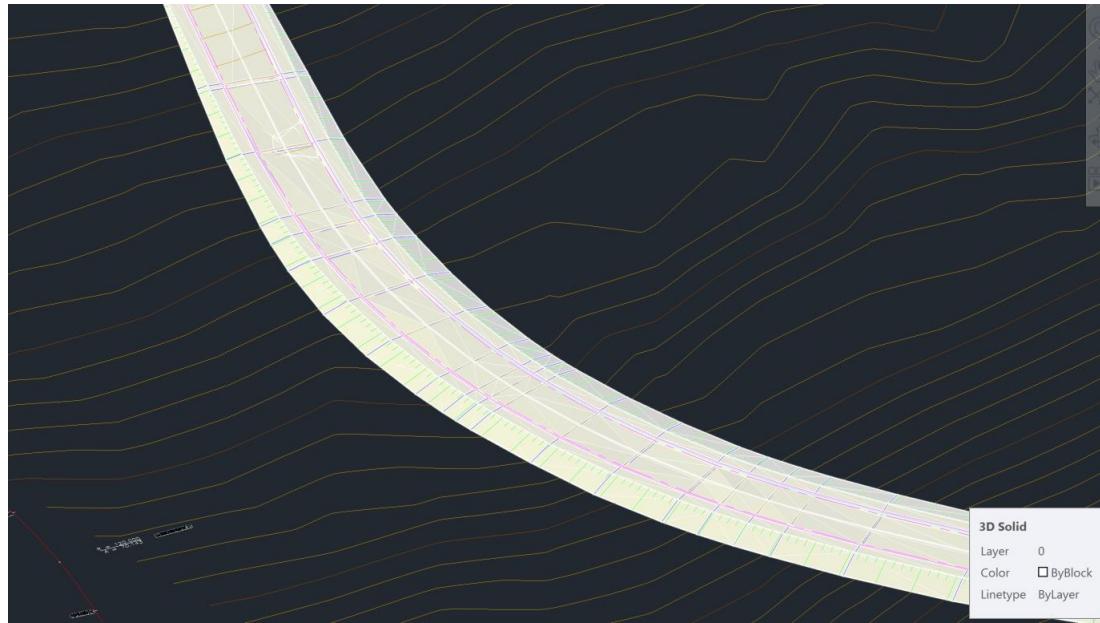
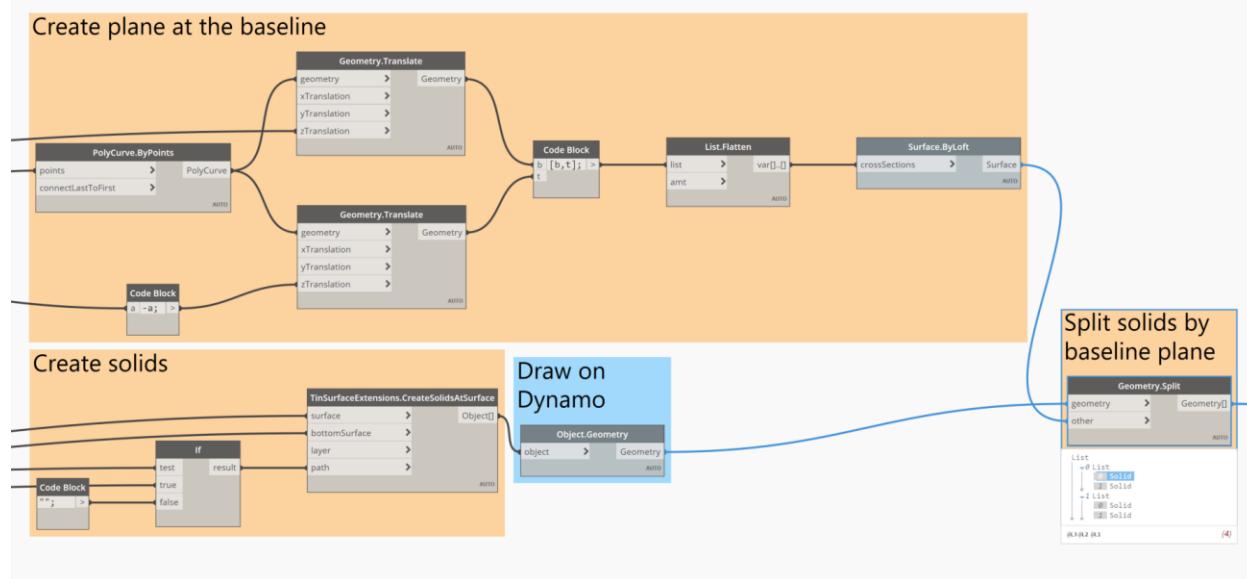


Figura 13 : Resultado de un POC para probar si los sólidos podían ser creados entre dos superficies

Para demostrar que los sólidos podrían ser divididos, se realizó la siguiente prueba, cortándolos a la mitad utilizando la línea de base:



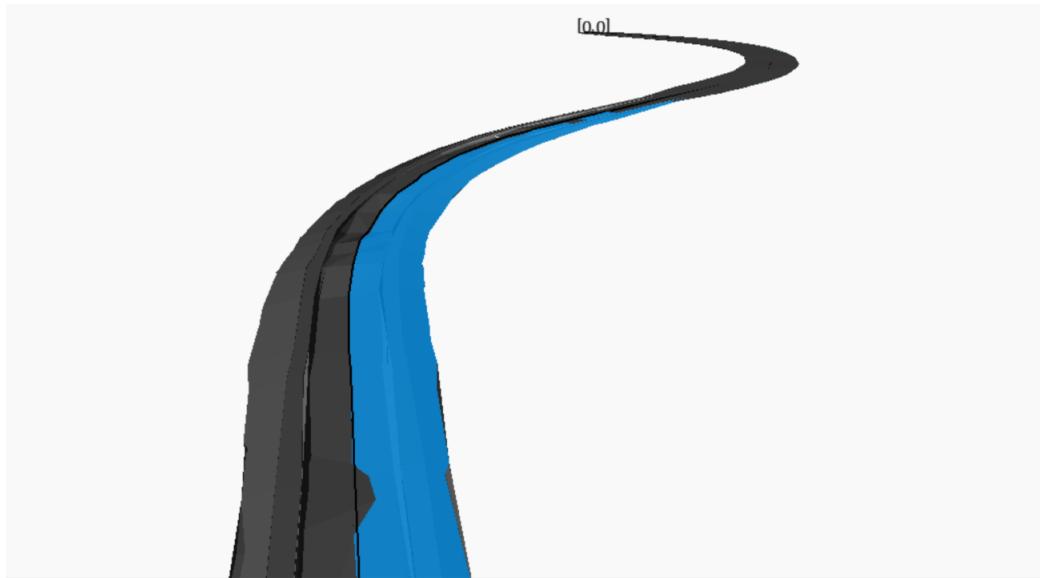


Figura 14 : Test para cortar los sólidos

De la misma manera, la aplicación y atribución de los property sets fue demostrada:

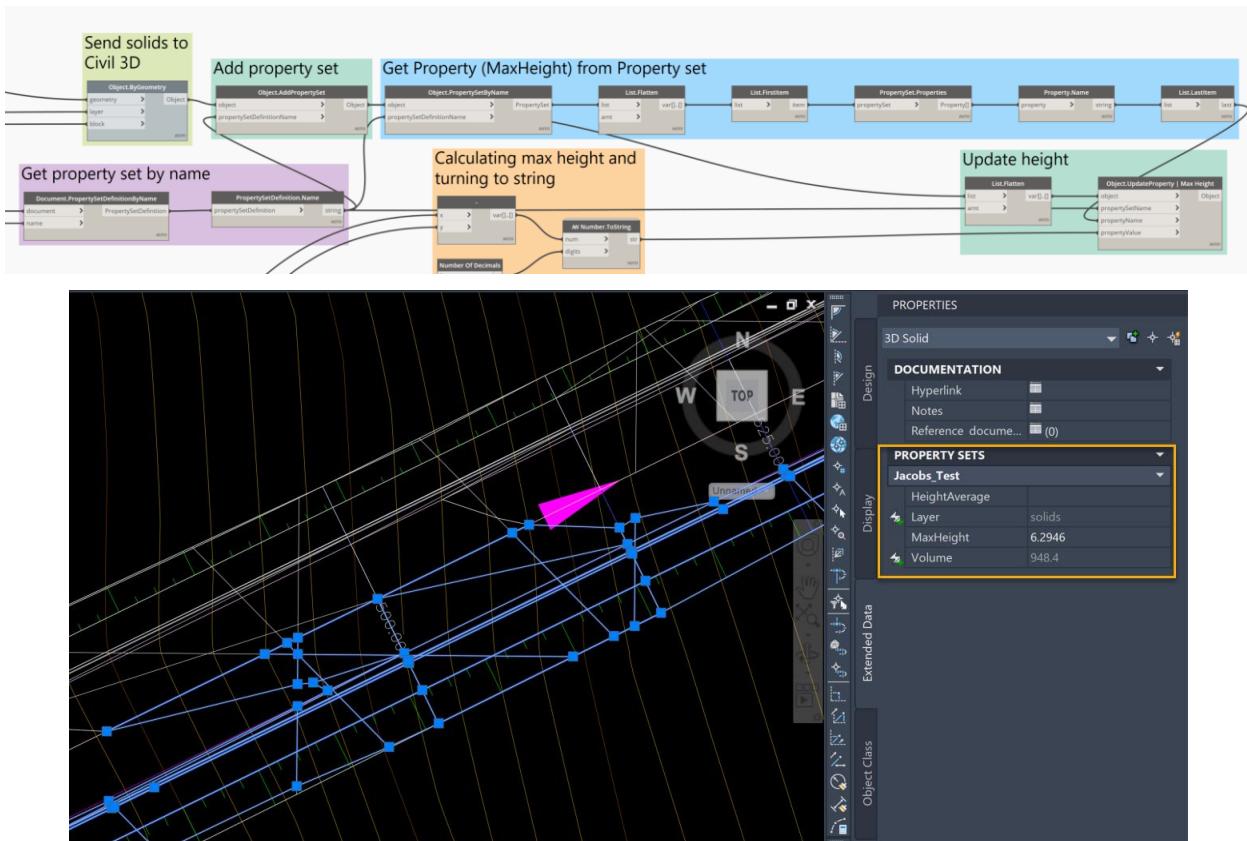


Figura 15 : Test para la aplicación de los property sets

Podemos ver la gráfica completa en esta imagen:

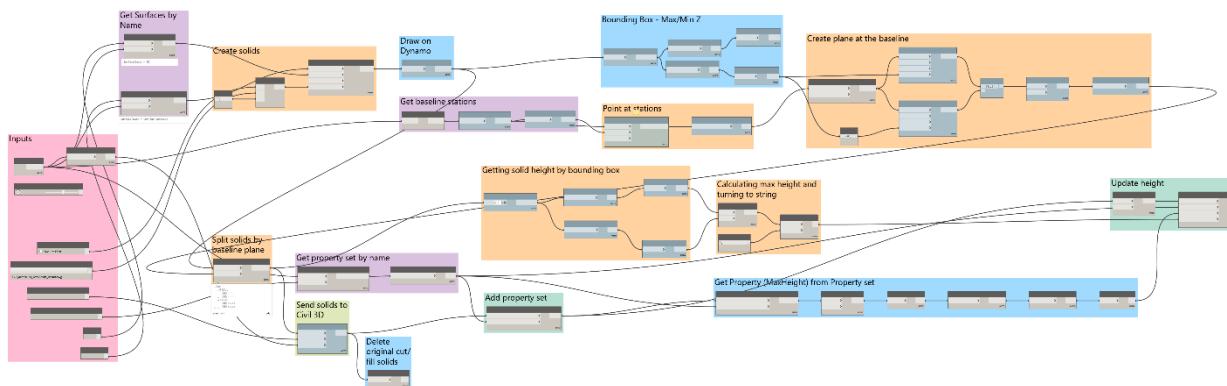


Figura 16 : POC para realizar sólidos que representan los movimientos de tierra de una carretera

La realización de este POC permitió demostrar que la lógica utilizada para la construcción de la gráfica funciona. Ahora, es necesario cambiar la gráfica para poder industrializarla y conseguir cumplir con todos los objetivos.

Industrialización

Para poder industrializar la gráfica, las modificaciones siguientes deberían de ser realizadas a la gráfica POC:

- Elegir cuáles serán las entradas paramétricas a incluir para poder ser manipuladas desde el Dynamo Player;
- Realizar la división de los sólidos como solicitado por el cliente, es decir por cuerpo de terracería;
- Calcular todas las propiedades requeridas, es decir, agregar el cálculo de la altura promedio;
- Probar en los modelos reales suministrados por el cliente.

Se tomó la decisión de modularizar la gráfica en tres gráficas diferentes para facilitar estas operaciones:

1. Creación de sólidos;
2. Manipulación de sólidos (por movimiento de tierra);
3. Aplicación, cálculo y actualización de property sets y sus propiedades.

En las primeras pruebas realizadas en el modelo proporcionado por el cliente, los sólidos no fueron creados. Gracias a las pruebas realizadas anteriormente, sabíamos que el nodo Dynamo estaba funcionando bien y por tanto que el problema venía de la API de Civil 3D. Al intentar replicar la operación manualmente, se detectó el problema:

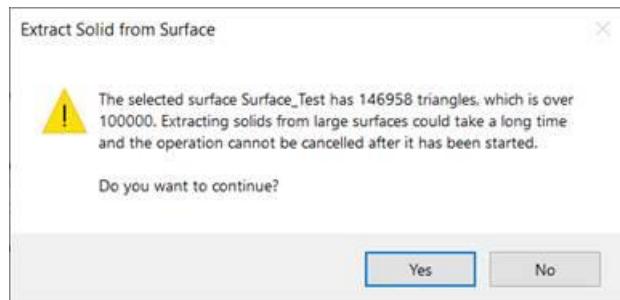


Figura 17 : Problema al crear sólidos entre superficies cuando existen superficies formadas por más de 100K triángulos

Por lo tanto, podemos inferir que es necesario recortar la superficie del terreno antes de poder lanzar la creación de sólidos:

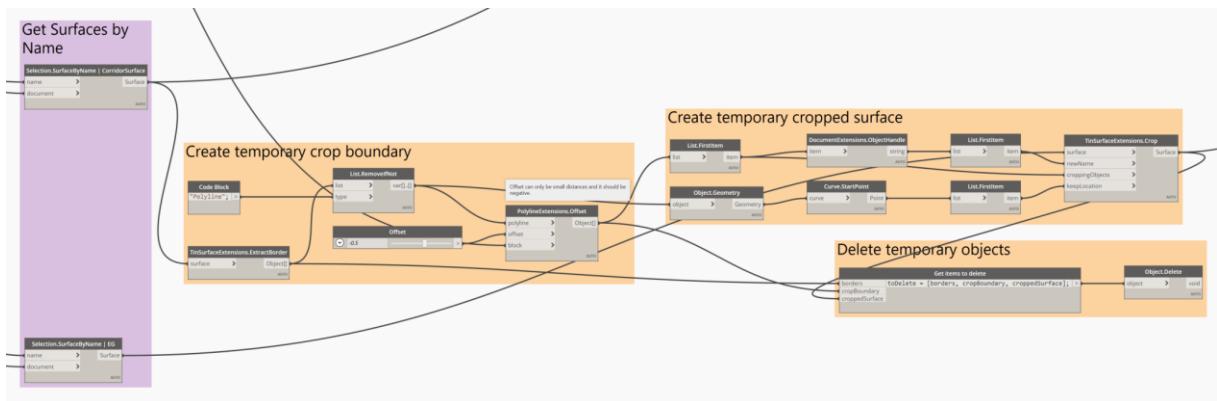


Figura 18 : Recortar la superficie del terreno

Con el uso de la superficie recortada, los sólidos fueron creados correctamente :

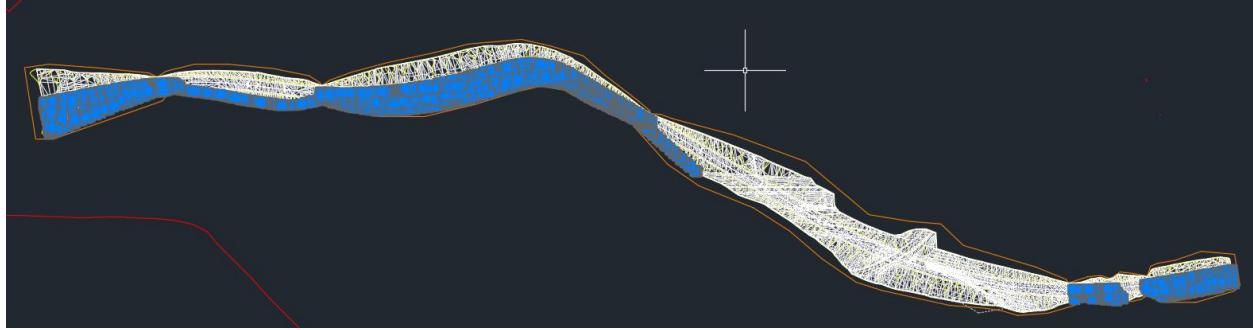


Figura 19 : Sólidos creados en el modelo proporcionado por el cliente

En cuanto a la división de los sólidos por cada movimiento de tierra, la realización de un proceso automatizado hubiera requerido demasiado tiempo (más allá de los 4 días previsto). Por esta razón, se decidió aceptar, por el momento, el realizar algunas etapas manualmente: la creación de polilíneas para mostrar el límite de cada cuerpo de tierra y por tanto el lugar en donde se realizará el corte de los sólidos. El proceso de corte sí que fue automatizado.

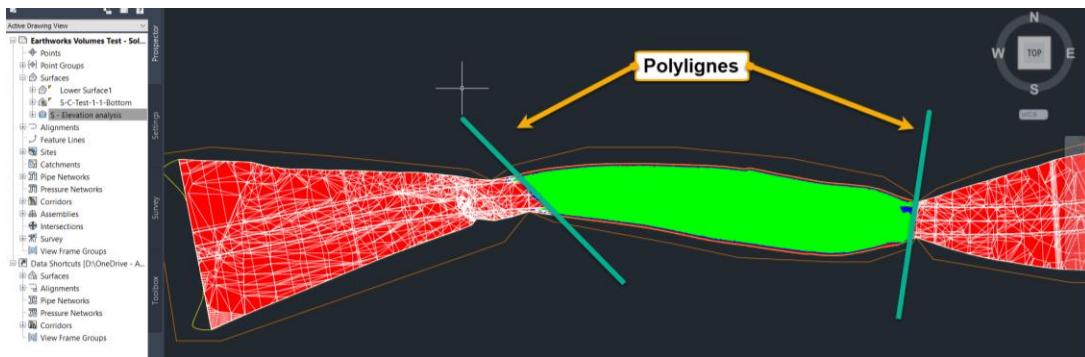


Figura 20 : Utilización de polilíneas creadas manualmente para determinar el eje de corte para dividir los sólidos

Finalmente, se utilizó la biblioteca de diccionarios en Dynamo, para permitir que el usuario final pudiera utilizar el Dynamo Player de manera interactiva cuando escogiera las superficies, proyectos y líneas de base necesarias para crear los sólidos:

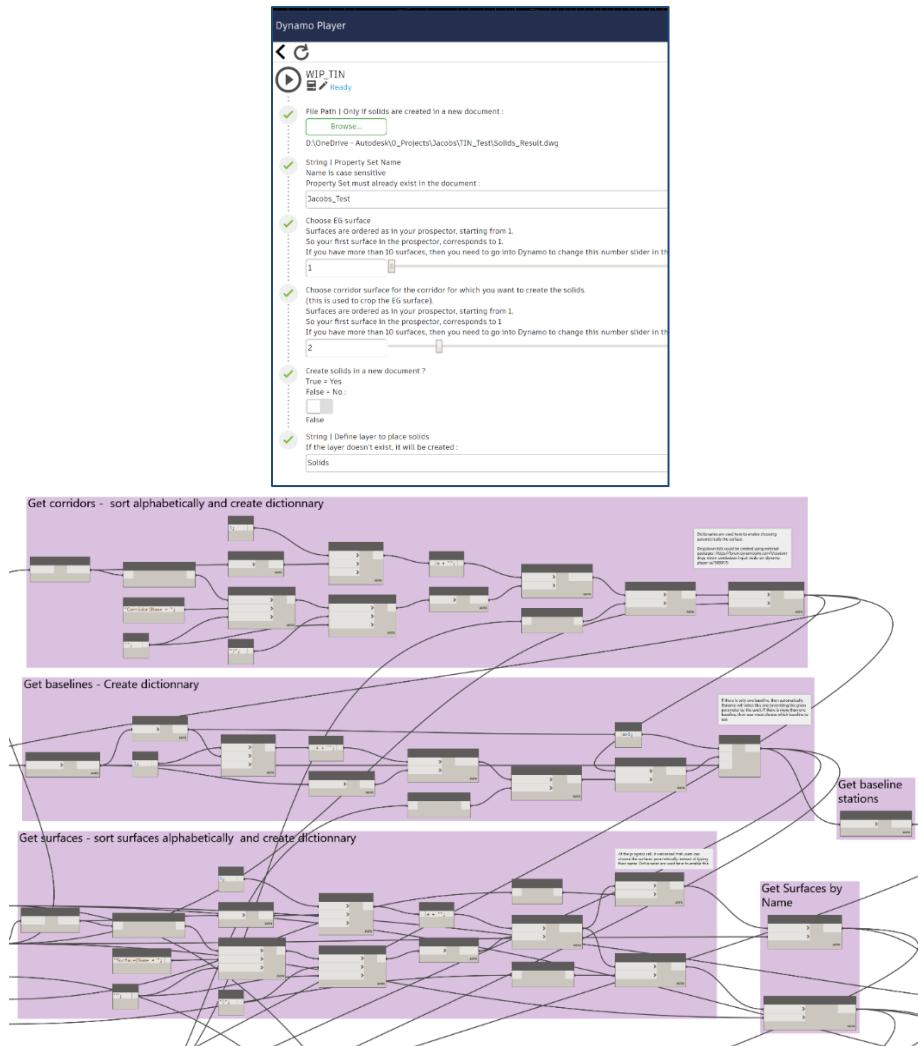


Figura 21 :Utilización de diccionarios para que el usuario pueda elegir de manera paramétrica en el Dynamo Player

Al final, estas fueron las gráficas resultantes:

- Creación de sólidos

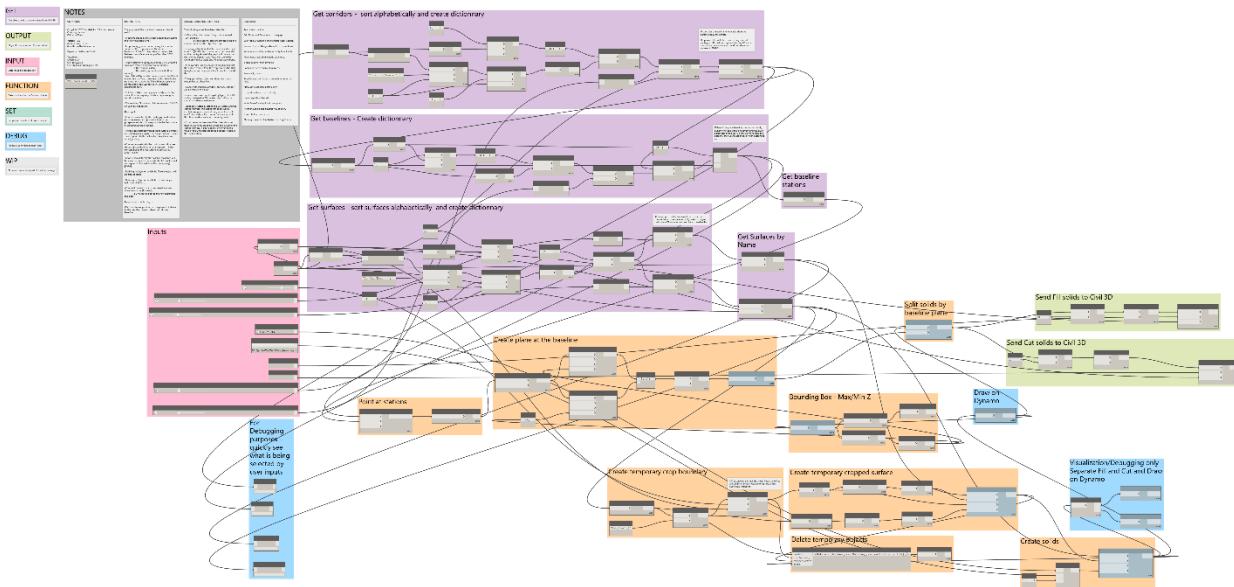


Figura 22 :Gráfica para crear sólidos representando el corte y relleno de una carretera

- Manipulación de sólidos (por movimiento de tierra);

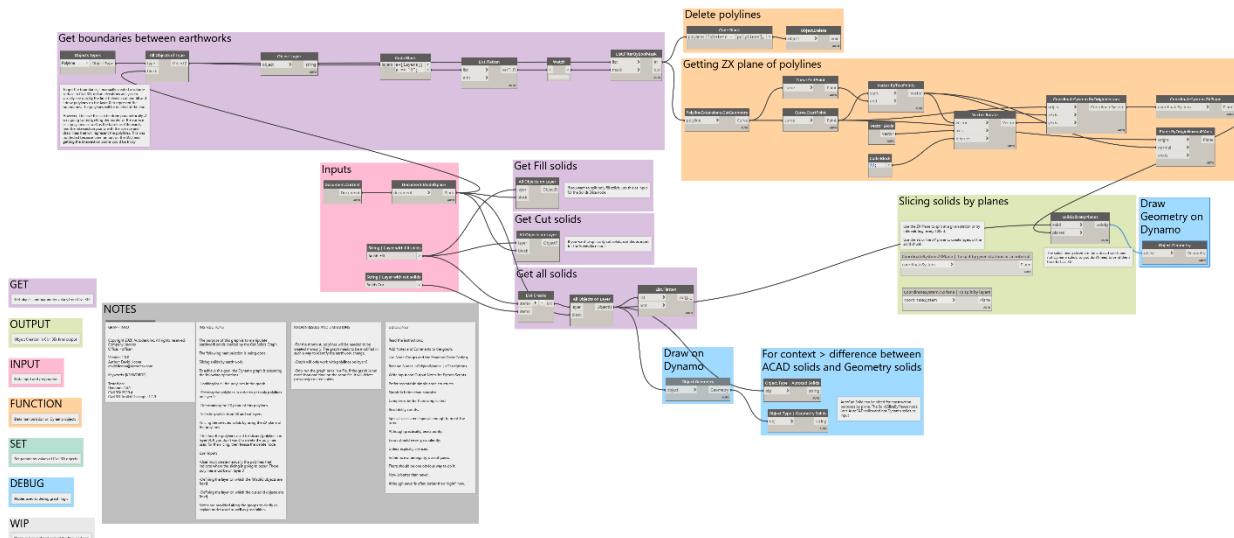


Figura 23 : Gráfica para dividir los sólidos por movimiento de tierra

- Aplicación, cálculo y actualización de property sets.

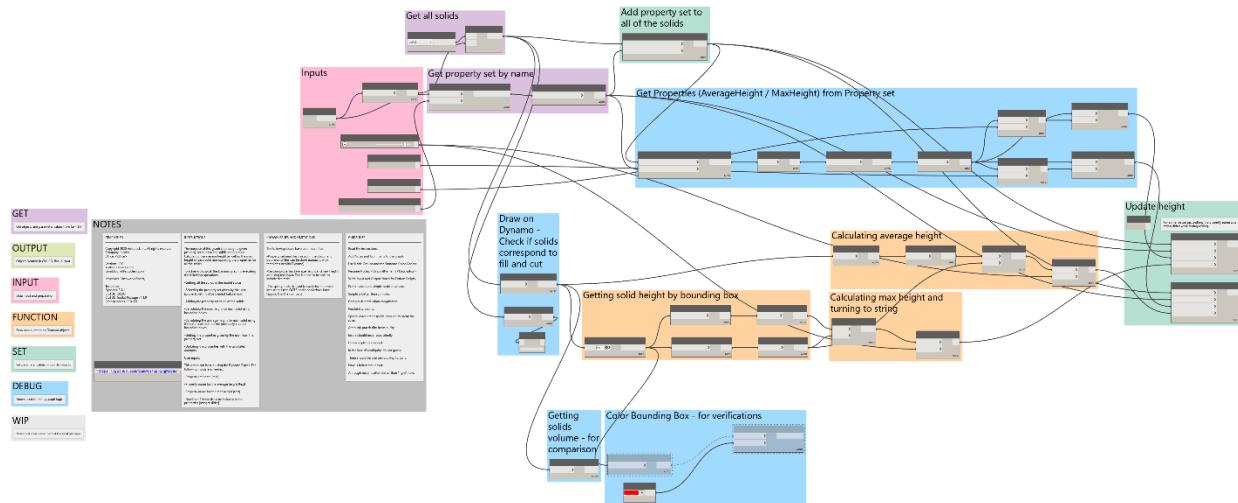


Figura 24 : Gráfica para atribuir property sets y calcular las propiedades de los sólidos

Implementación / Adopción

Para llevar a cabo un cambio consecuente (como la automatización) dentro de una organización, se debe tener en cuenta los siguientes aspectos:

- **Personal:** podría ser necesario contratar nuevas personas con las habilidades adecuadas o capacitar personal interno;
- **Organización:** la introducción de la automatización puede requerir la creación de nuevos roles y responsabilidades, la definición de un plan de implementación para la nueva estrategia y la gestión de la comunicación con las partes interesadas;
- **Procesos:** es necesaria la coordinación de las personas implicadas y el cambio de tareas y procesos que constituyen el modelo actual del negocio;
- **Tecnologías:** puede ser necesario de cambiar o actualizar los componentes tecnológicos para permitir utilizar la solución.

Nomenclatura de archivos

Las convenciones de nomenclatura para guardar archivos se aplican a la mayoría de los archivos dentro de una empresa (o al menos deberían). Los archivos de Dynamo no deben escapar a esta regla. Las convenciones de nomenclatura aclaran el contenido, ayudan a estructurar la gestión de contenido y permiten encontrar fácilmente los archivos.

Ejemplo de una nomenclatura de archivo:

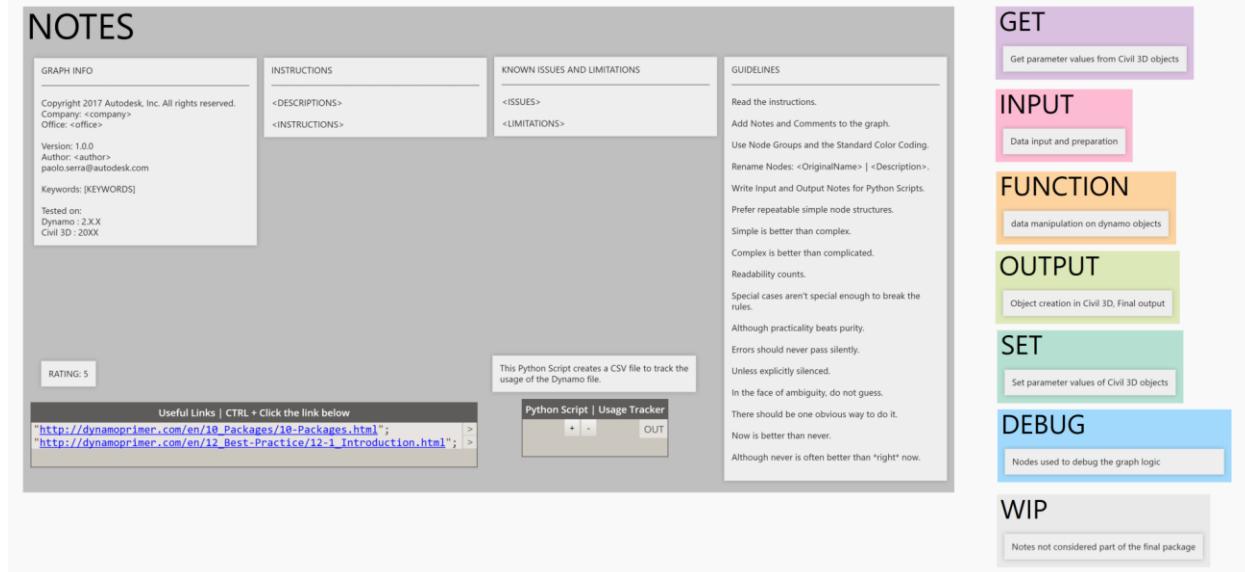
- Programa_Disciplina_UtilizaciónBIM_Propósito_Versión
- « **C3D_Carreteras_Documentación_PerfilLongitudinal_V1** »

Plantilla

Debe existir una plantilla estándar para las gráficas Dynamo creadas dentro de una empresa.

Esta debe aplicarse en cada gráfica. Pueden aprovecharse los colores en Dynamo para introducir una codificación de colores. La plantilla también debe incluir las instrucciones con la información de la gráfica.

Aquí está la plantilla utilizada en Autodesk y que puede servir como ejemplo:



The screenshot shows the Autodesk Dynamo Standard Template. It includes sections for GRAPH INFO, INSTRUCTIONS, KNOWN ISSUES AND LIMITATIONS, and GUIDELINES. A sidebar on the right lists categories: GET, INPUT, FUNCTION, OUTPUT, SET, DEBUG, and WIP.

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

Figura 25 : Ejemplo de la plantilla estándar para los archivos Dynamo

Almacenamiento de archivos

Los archivos de Dynamo deben centralizarse en una estructura de carpetas por disciplina y / o funcionalidad. Los archivos de origen deben estar separados de los archivos que se utilizarán en los proyectos. Cuando se necesita utilizar una gráfica Dynamo en un proyecto, la gráfica debe de copiarse desde la biblioteca centralizadas a la carpeta del proyecto, para poder crear una relación entre el modelo Civil 3D y la gráfica Dynamo. Esto también permite administrar las actualizaciones del modelo y adaptarse a la configuración de un proyecto específico.

Las nuevas soluciones de Dynamo deben desarrollarse en un entorno separado de las soluciones aprobadas (esto puede ser tan simple como una carpeta accesible solo para las personas que desarrollan las soluciones).

Pruebas y validación

Al desarrollar una gráfica de Dynamo, es importante validar la solución con el usuario final e integrar cualquier comentario relevante en la solución. Una solución debe de ser validada no solo si ejecuta la función que se espera sino también leyendo las instrucciones y la documentación incluidas para asegurarse de que sean legibles y fáciles de entender.

Se debe alentar a los usuarios finales a documentar qué tan bien se desempeñó la solución Dynamo dentro del contexto del proyecto. Esto es crucial para comunicar al resto de la organización sobre el éxito de la automatización, de modo que otros coordinadores de equipo y usuarios finales puedan beneficiarse de la experiencia previa.

Seguimiento

La recopilación de información sobre el uso de la solución Dynamo ayuda a determinar cuáles son las soluciones que los equipos encuentran más beneficiosas, así como el valor asociado con la solución. Se deben de tener indicadores de desempeño para poder calcular el valor.

Las gráficas de Dynamo deben contener un script de seguimiento de uso que recopile información sobre el uso del gráfico (tiempo, usuarios, nombre de la computadora, nombre del gráfico, ubicación del gráfico y nombre del modelo).

Comunicación

Una campaña de comunicación eficaz es la clave para garantizar la adopción de la automatización dentro de una organización.

Se deben de comunicar los éxitos obtenidos a los otros equipos y departamentos dentro de una empresa. Por lo tanto y para cada solución, hay que solicitar comentarios / retroalimentación de los usuarios finales para cuantificar los beneficios asociados con la automatización. Por ejemplo, ¿cuánto tiempo ahorró el usuario final con la solución? Estos elementos se pueden utilizar durante las campañas de comunicación.

También es muy importante basarse en los testimonios de los usuarios de Dynamo. La comunicación se tendrá más en cuenta si es transversal y no vertical (es decir, no viene solo por un canal de jerarquía).

En cuanto las herramientas o metodologías, el video es un excelente canal de comunicación. Es dinámico y visual y, por lo tanto, más atractivo que un correo electrónico, por ejemplo. Hay que realizar la narración de la historia teniendo en cuenta a la audiencia y por lo tanto adaptar la narrativa sin olvidar de explicar bien el problema, analizar la solución y destacar el valor obtenido.



Figura 5 : Concentrarse en la narración para crear una campaña de comunicación eficaz (fuente : [Rawpixel](#))

Grupo de utilizadores

Una de las mejores prácticas para garantizar la adopción de la automatización es formar un colectivo de usuarios de Dynamo que puedan apoyarse entre sí y, lo que es más importante, impulsar el cambio necesario para implementar la automatización dentro de los equipos.

Recursos humanos

Es importante tener en cuenta que el desarrollo de gráficas Dynamo requiere ciertos conocimientos y capacidades que pueden no existir actualmente dentro de una organización. Por lo tanto, se debe de:

- Capacitar al personal internamente o;
- Invertir en la adquisición de talento.

Para la creación de las primeras soluciones en Dynamo, una buena idea es ser acompañado por un experto que pueda guiar el desarrollo y la implementación de las soluciones.

En conclusión

- Algunas de las mejores prácticas son: la utilización de plantillas estándar con un código de colores e instrucciones. Se debe medir la utilización de las gráficas e instaurar indicadores de desempeño. Finalmente, el nombre de los archivos Dynamo debe de seguir una nomenclatura específica y los archivos deben de ser centralizados en un repertorio;
- Realizar campañas de comunicación mostrando el valor añadido de las gráficas. Apoyarse en un grupo de usuarios para extender el alcance de la campaña de comunicación;
- Tener un plan de capacitación o de adquisición de talento.

Otras fuentes

Para aprender a utilizar Dynamo:

- [Primer](#) ;
- [Accesorize your Design – Dynamo for Civil 3D](#) | **Stacey Morkin y Dylan Kahle**

Para adquirir más experiencia en Dynamo:

- [Dynamo Forum – Civil 3D](#) ;
- [Computational Design for Civil Engineers](#) | **Paolo Serra y Safi Hage**
- [Dynamo in Civil 3D Introduction—Unlocking the Mystery of Scripting](#) | **Jowenn Lua y Andrew Milford** ;
- [Generating, Transforming, and Analyzing Railway Design Data in Civil 3D and Dynamo](#) | **Wouter Bulens y Steve Crokaert**.

Clases AU 2020 (no hay link disponible) :

- Diseño Generativo | **Raquel Bascones y Paolo Serra** ;
- Civil 3D Toolkit | **Jowenn Lua**.

Agradecimientos

Me gustaría agradecer a varias personas que me apoyaron en la realización de este curso: **Philippe Bonneau** por su valioso consejo y guianza, **Emmanuel Lagardette** por alentarme a participar este año en AU, **Jowenn Lua** y **Raquel Bascones** por compartir sus experiencias, **Paolo Serra** a quien llamo a menudo cuando tengo preguntas sobre Dynamo para Civil 3D, **Daniel Fernandez** por ayudarme en la sesión de preguntas y **Lucie MAGAUD** y **Margaux GRANDON** por escuchar la presentación y darme retroalimentación.