

## Introducción

Para el siguiente trabajo se trabajó con Three.js para el desarrollo de una demo técnica. La misma consiste en una fábrica de piezas 3D. En ella, se encuentra una impresora que permitirá imprimir distintos modelos de piezas, generadas por rotación y por extrusión. Para trabajar con las piezas existirá un montacargas que podrá tomar las piezas, desplazarlas en la escena, y dejarlas en una estantería.

## Piezas

Dentro de las piezas a generar hay 2 tipos principales, las de barrido y las de rotación. Para la generación de ambos tipos de piezas se definieron curvas a partir de la cual generar todo el volumen. Para esto se utilizó la funcionalidad de Curve que permite obtener puntos en cualquier lugar de la curva especificándolos con un valor entre  $[0, 1)$ .

En el caso de las piezas de rotación, la idea es definir la curva en el plano XZ y después generar el volumen rotando en el eje Z. Para la generación del volumen se utiliza BufferGeometry. Esto consiste en definir todos los puntos que representan la figura de forma tal que representen el conjunto de triángulos que deseamos dibujar en pantalla. Para esto se generan múltiples curvas rotándolas sobre el eje Z. Luego, cada par de curvas adyacentes es unido de forma que generen el triángulo en cuestión. Además de definir los puntos es necesario definir las normales de esos puntos. Esto permite que la pieza responda a la iluminación. Para lograr que la pieza se genere parcialmente no se dibuja toda la curva sino un intervalo entre  $[0, d)$  con  $0 < d < 1$ .

En el caso de las piezas de barrido el procedimiento es muy similar. Se generan múltiples curvas y se unen de forma tal que definan los triángulos necesarios para visualizarla. En este caso, las múltiples curvas se generan en el plano XY. Luego son copiadas y movidas a lo largo del eje Z. En caso de tener un ángulo de barrido, además de desplazarlas a lo largo del eje Z, es necesario rotarlas sobre el mismo eje. Para lograr la generación parcial simplemente se genera una menor cantidad de curvas.

## Impresora 3D

La impresora 3D fue construida, como el resto de los componentes de la escena, generando un grupo de objetos. Esto facilita considerablemente el trabajo con múltiples geometrías, las cuales se agregan a este grupo y pueden trabajarse y desplazarse en conjunto. De esta forma, se construyó primero la base de la impresora y luego el ascensor, que consta de una columna y el cabezal (otro grupo de geometrías). Así todas estas partes forman lo que finalmente se tratará como la impresora.

Este objeto será el encargado de imprimir las piezas, por lo que la misma deberá poder lograr un efecto de generación parcial. Esto se realizó generando la pieza en múltiples pasos, incrementando en cada uno la altura final de la geometría a generar. Al mismo tiempo, el cabezal de la impresora se levantará en una proporción similar dando así el efecto de que es este quien imprima el objeto.

A esta estructura deberá acercarse el montacargas para poder levantar las piezas. Al haber una pieza en la impresora, no se podrá imprimir una nueva hasta que no se tome la anterior.

## Montacargas

El montacargas será el principal actor dentro de nuestra escena. El mismo, como la impresora, fue construido a partir de distintas geometrías que serán tratadas como un grupo. En este caso lo fueron por un lado el carro y por el otro el ascensor.

A diferencia de la impresora, el montacargas deberá recibir input del usuario, tanto para moverse como para levantar o bajar su ascensor y tomar o dejar las piezas. Para eso se definieron en el modelo las distintas funcionalidades y luego en el la escena principal las mismas se asociaron a los eventos de las teclas indicadas por la consigna. Se tomó en cuenta tanto cuando se presiona la tecla como cuando se libera.

Para el movimiento es muy útil la funcionalidad de agrupar objetos. Esto permite tener un sistema de referencia para ubicar los distintos componentes del montacarga, luego, cuando haya que desplazar, solo es necesario cambiar donde esta ubicado este sistema de referencia respecto a su padre, simplificando el movimiento de toda las piezas.

Para el movimiento tanto del carro como subir y bajar el ascensor se decidió implementar dos funciones, una al presionar la tecla, que aumentará la velocidad o la rotación correspondiente del componente, y otra al soltarla que la frena en la posición que haya quedado al momento de ejecutarse.

La funcionalidad más compleja será la de tomar la pieza de la impresora. Para realizar esto se definió la condición de que debe encontrarse a una distancia cercana a la impresora al presionar el botón de tomar la pieza o no tendrá efecto. Para poder tomar la pieza, lo que se hace es desvincular la misma del grupo correspondiente a la impresora y vincularla al del ascensor del montacargas. De esta manera, la misma se verá siendo “transportada” por este. Solo se podrá tomar una pieza en caso de que esté vacío. Una vez tomada la pieza solo podrá soltarla dejándola en un lugar libre de la estantería.

## Estanterías

Para las estanterías se utilizó también la funcionalidad de grupo. En este caso no fue solo de utilidad para definir la estructura en sí, sino que también fue útil para el manejo de las piezas a ser almacenadas. En este caso se generó un grupo para cada uno de los posibles lugares de la estantería a ser llenados. Luego estos grupos fueron almacenados en una lista. A la hora de querer almacenar un objeto, se recorre la lista y se puede buscar el grupo más cercano. Una vez encontrado el mismo se puede depositar la pieza en ese grupo, el cual ya está ubicado en la posición de la estantería que le corresponde, si este está vacío. Para esto bastará con sólo agregar la pieza en el centro de este sistema de referencias.

## Galpón

El escenario donde vivirán los componentes mencionados previamente no es más que algunas geometrías representando paredes y un suelo, las cuales una vez más, son

tratadas como un grupo. Tanto al suelo como a las paredes, se les aplicó una textura de grilla utilizando un Cargador de texturas provisto por Three.js. En este escenario se generan los componentes anteriormente mencionados, las luces y las cámaras pedidas por la consigna.

## Interfaz

La interfaz del juego se realizó utilizando el módulo *dat.gui* provisto por Three.js. Sobre esta se definieron dos menús. En el primero se encuentran los parámetros a modificar de la geometría, como cuál geometría imprimir, con qué rotación (solo valido para piezas de extrusión), la resolución, la altura y su material. Sobre este último puede utilizarse el wireframe para ver la malla de triángulos utilizada, pudiendo contemplarse mejor la resolución. En el segundo menú se encuentra la acción de imprimir la pieza, la cual será ejecutada por la impresora 3D.

## Cámaras

La aplicación cuenta con 6 cámaras distintas. Las primeras 3 son cámaras con un control orbital, estas están ubicadas para apuntar al centro, la impresora y la estantería. Para el manejo del control orbital se utilizó el OrbitController incluido en los ejemplos de Three.js. Se le realizó una mínima modificación para poder controlar el zoom de forma externa. Para implementar estas cámaras se utilizó un único objeto cámara y se modificó el target del orbit control. Este representa el punto del espacio sobre el cual se orbita.

Por otro lado, se encuentran las 3 cámaras relacionadas al montacarga. Para estas se utilizó un objeto cámara para cada una. Luego, cada una de estas cámaras se agregó al grupo del montacarga y se las ubicó de forma que produzcan el ángulo buscado.