

Librerías IFC.js

para una industria de la construcción y arquitectura colaborativa.

Antonio González Viegas¹

Conferencia realizada el 12 de enero del 2022, en el marco del ciclo de charlas "Paradigmas Colaborativos Abierto Herramientas Libres para Arquitectura y BIM" organizado por la Facultad de Arquitectura, Urbanismo y Geografía de la Universidad de Concepción.

114

Lo primero que hay que tener en mente para dar a conocer el proyecto colaborativo de la librería para IFC en Javascript acerca del cual expondré, es que IFC es el formato de archivos que utilizamos en BIM¹ cuando queremos trabajar con aplicaciones de diferentes marcas. Por ejemplo, si nos interesa ocupar una aplicación de **Autodesk** y queremos llevar esos datos a una aplicación de otra marca, de la marca que sea, IFC es el formato indicado para hacerlo. Por su parte, el **IFC.js** funciona como una librería de Javascript, que es un lenguaje de programación. Concretamente, entonces, estamos hablando de un lenguaje de programación que se utiliza para los navegadores Web, Internet y demás.

De este modo, en el mundo del software, el BIM es la forma de trabajar que tenemos los arquitectos y los ingenieros de la construcción mediante la cual, básicamente, en lugar de trabajar en 2D, ahora lo hacemos en 3D y con datos. Es parecido a pensar que antes dibujábamos planos y ahora creamos casas de los *Sims*².

¹Building Information Modeling.

²Videojuego "Los Sims".

El problema es que, para trabajar con datos en 3D, hace falta un software como Revit, Archicad, Allplan y la idea es crear un modelo tridimensional del edificio sobre el cual se puede hacer una serie de análisis que mejoren mucho el original o que permitan ahorrar mucho dinero. Por ejemplo, generar análisis energéticos que permiten prever cómo va a comportarse el edificio; análisis económicos que permiten ver, en tiempo real, el efecto que tienen las decisiones de proyectos sobre el precio real del edificio, entre otras.

Entonces, para conseguir todos esos objetivos, hace falta software. Y el problema es que este software es muy difícil y caro de hacer, pues solamente unas pocas empresas tienen acceso a la tecnología necesaria. De hecho, a la hora de hacer BIM tenemos cinco o seis empresas entre las que podemos elegir. Si comparamos ese número, por ejemplo, con cuantas empresas se pueden elegir a la hora de comprar un videojuego, ¿cuántas empresas de videojuegos hay en el mundo? Es un orden de magnitud completamente distinto y el precio en ambos casos es diferente.

La complejidad de las aplicaciones BIM y el IFC.js como solución

Revisemos entonces, lo siguiente: ¿Por qué es difícil crear aplicaciones BIM en comparación a crear videojuegos? Los videojuegos tienen una complejidad tecnológica mayor que las aplicaciones BIM. En una aplicación BIM, a fin de cuentas, dibujamos paredes, asignamos propiedades y un poco más. Quizás, como mucho, haremos un control de colisiones, es decir, cálculo estructural, pero, aún así, en un videojuego ocurren muchísimas más cosas.

De este modo, la realidad es que es difícil crear aplicaciones BIM porque, primero, **no hay ningún tipo de documentación**. No existen registros. Se debe empezar de cero, sin guía, ni referencia, ni tutorial.

En segundo lugar, **el formato IFC (para archivos BIM) es complejo para importar/exportar datos**. Cuando se crean aplicaciones BIM, siempre se querrá importar o exportar datos, porque sino tendríamos que crear todo el edificio en la aplicación que creamos y eso nunca ocurre. Un ejemplo concreto es que, si vamos a hacer una aplicación de cálculo estructural, necesitamos hacer una aplicación de presupuesto, instalaciones o lo que sea. Es decir, siempre vamos a querer importar datos de otra aplicación, ya sea Revit, Archicad, lo que sea y para hacer eso hace falta IFC, lo cual es complicado de implementar. Mi experiencia en empresas de software BIM me ha enseñado que normalmente existe un equipo de gente donde el único trabajo es resolver el tema de IFC. Trabajan 40 horas semanales, o sea tiempo completo, y su único trabajo es implementar el formato IFC. Entonces, esto deja fuera a todos los pequeños desarrolladores que podrían interesarse en trabajar en este campo.

En tercer lugar, realizar modelos 3D posee alta complejidad. El IFC.js, es un software BIM que busca tener la misma calidad que tienen las grandes empresas, pero, totalmente abierta y gratuita para que cualquiera la pueda utilizar. Por tanto, es una aplicación que permite crear software BIM a cualquier persona, incluso quienes no tienen tanta experiencia en programación. Incluso está pensado para usuarios que no saben programar, que no tienen experiencia en el tema BIM, porque no son arquitectos o ingenieros, pero aún así puede entrar aquí y crear aplicaciones similares a las que se venden en las grandes casas comerciales. Todo esto de manera gratuita y abierta para siempre.

En otras palabras, estamos intentando que quien lo desee pueda ingresar y, en un par de tardes, desarrolle su propia herramienta BIM para calcular estructuras, presupuestar un edificio, etc. Esta no es una aplicación final, esto es una pieza y la más complicada, para que luego llegue quien sea y cree su software sin partir del punto cero.

Este proyecto lo hemos iniciado tres desarrolladores: yo, que soy un arquitecto español, con conocimientos de programación y experimentado en BIM; Tom, ingeniero en software jefe, en BIMcollab, la empresa BIM de software más importante de Holanda y Harry, quien también es programador en una empresa BIM de Reino Unido. Y todo nuestro trabajo es ciento por ciento abierto, gratuito para siempre y documentado. Nuestra idea es que, aunque nosotros seamos los creadores, por así decirlo, todo esté guiado por una comunidad. Es decir, no somos nosotros los que tomamos las grandes decisiones.

La idea surge al comprender que es absurdo que una empresa tenga que crear su software BIM desde cero. Entonces, como hace falta una serie de recursos de los que una empresa media o pequeña no dispone, en lugar de hacer cada uno "la guerra" por su lado e implementar todo desde cero el producto que queremos para nuestra empresa, unimos fuerzas y creamos una base tecnológica común. Esto, con el fin de que si alguien mañana quiere hacer una herramienta BIM, no parta de la casilla cero, sino parta de la casilla 700, porque los primeros setecientos pasos los he dado yo y todos quienes están colaborando en el proyecto.

Muchas veces, cuando hablamos de Open Source, hablamos de cosas que no funcionan muy bien. Sin embargo, nosotros estamos formalizando nuestro proyecto mediante el involucramiento de empresas para que entren, participen y colaboren creando software BIM con el fin de que luego cualquiera pueda realizar el suyo. En este sentido, ya contamos con una lista de empresas contribuyendo.

Veamos un ejemplo concreto: a mí una empresa me pide un generador de planos a partir de modelos IFC, es decir, cargar un modelo IFC a un modelo BIM en el navegador o en una aplicación de escritorio y, asimismo, crear los planos CAD. Este trabajo yo podría hacerlo en el código cerrado de mi empresa, en la aplicación de mi empresa y sólo nosotros lo tendríamos. Pero, en lugar de hacer eso, yo cojo ese código y lo guardo en la librería, que entenderemos como una aplicación

abierta que cualquiera pudiera utilizar gratuitamente. De esta forma, todos los usuarios de la librería tienen un generador de planos que hacemos todos.

Es así como parte de la funcionalidad que implementaremos en nuestro día a día, lo hacemos directamente en la librería, de forma que, en lugar de avanzar paso a paso, estamos avanzando todos a la vez, colaborando para ayudarnos entre todos y así conseguir mayor cantidad de aplicaciones y de más nivel.

Hasta ahora, lo que ocurría era que, muchas veces, cuando alguien intentaba hacer una aplicación BIM no llegaba a lograrlo porque con un equipo de 10 personas, que trabajan 40 horas a la semana, no vas a conseguir un segundo Revit o Archicad. Es totalmente imposible, porque son aplicaciones que tienen mucha inversión encima; horas de ingeniería informática y al final, el unir fuerzas es lo único capaz de coger esta tecnología y visibilizarlas.

Una cosa que pareciera utópica, es el estándar en el mundo del software. Si alguno ha programado alguna vez, sobre todo en temas de programación web, sabe que todo el mundo comparte su trabajo. Las herramientas que se utilizan hoy en día a nivel web para crear aplicaciones web son herramientas *Open Source* de Google, Facebook. O sea, todas estas empresas se han dado cuenta que al final el *Open Source* es el camino a seguir.

El problema es que en la industria de la construcción nos hemos quedado atrás. Yo tengo amigos que son ingenieros de software y cuando les cuento lo que tenemos, me dicen: “eso es lo que teníamos nosotros hace 15 años”, y digo sí, que ese desfase existe. Al final, lamentablemente tenemos que lidiar con ello durante nuestro día a día. Como usuarios, esta situación se traduce en precios muy altos, las sugerencias para las empresas nunca se van a implementar, avance muy lento de desarrollo como programador. Asimismo, el problema es que hay un mercado muy lucrativo, similar al software de medicina, un mercado donde la gente está dispuesta a pagar mucho dinero, pero es difícil comenzar de cero y solo.

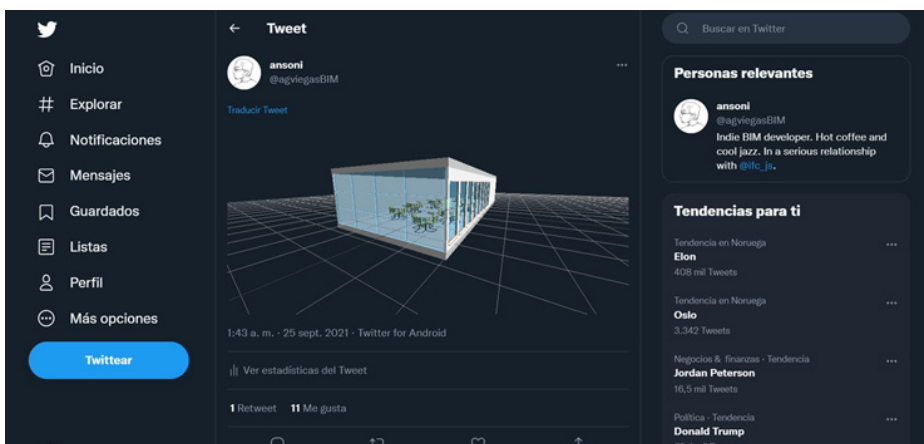
IFC.js ni es una empresa, ni es un producto. Es una iniciativa que hemos puesto en marcha y cada vez está sumando más empresas. Tenemos un canal de *discord* que es abierto, <https://ifcjs.github.io/info/>³, donde verás que hay más de mil personas ya; diferentes empresas que ya están creando sus productos y sus negocios encima de lo que nosotros hemos hecho y contribuyendo de vuelta. Al final, se está creando esta retroalimentación donde estamos intentando que todos podamos salir ganando y que todos puedan usar el software BIM.

³<https://ifcjs.github.io/info/>

Los usos de las librerías IFC.js: infinitas posibilidades

Este proyecto no es solo algo que nos permite, con Javascript, crear aplicaciones BIM 3D donde podéis seleccionar elementos, visualizar sus propiedades o editarlos. Esto nos permite también hacer aplicaciones web, aplicaciones multiplataformas, de escritorio, aplicaciones de servidor, etc.

Figura 1.
Visualizador BIM integrado a la plataforma Twitter, usando IFC.js



Nota: Demostración de IFC.js integrado en Twitter. **Fuente:** Imágen gentileza de Antonio Viegas, Enero 2022.

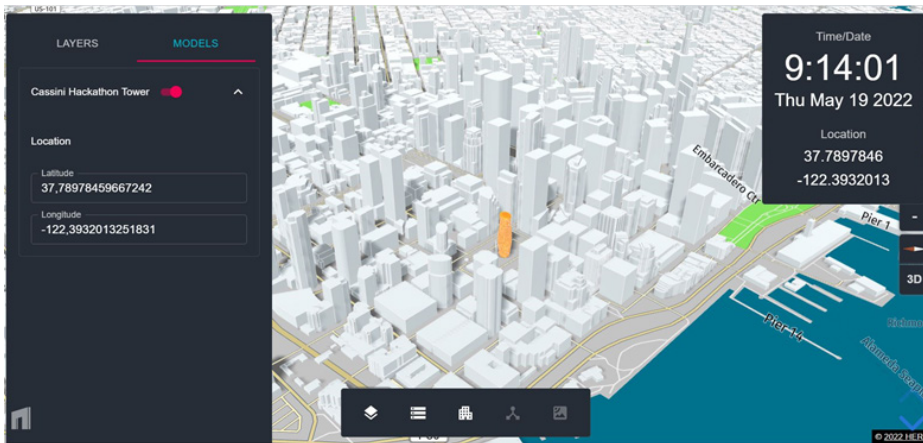
Por ejemplo, *Twitter*. Utilizando IFC.js, se puede programar una extensión de Google Chrome y la idea es convertir *Twitter* en un visor BIM (Figura 1). Se carga un modelo y puedo navegar por él. Entonces, se puede seleccionar elementos, crear planos de inserción, visualizar las propiedades directas e indirectas, hacer filtros, entre otras cosas. Todas las herramientas que está mostrando, por ejemplo, el generador de planos y demás, también se pueden utilizar directamente desde *Twitter*, porque es *Google Chrome* el que está siendo utilizado como herramienta BIM. Internet está escrito en Javascript y, como nuestra librería es puramente Javascript, se puede ejecutar de manera nativa en cualquier navegador y página web. La noción de que el BIM es aquello que ocurre dentro de una aplicación de escritorio, es anticuada: podemos hacer aplicaciones web que son tan potentes como las aplicaciones de escritorio, con posibilidades infinitas.

El proyecto, entonces, va de que, en poco tiempo, podamos crear nuestras propias aplicaciones BIM; trabajar con datos BIM y, si queremos, utilizarlas nosotros internamente para lo que sea. Incluso venderlas como soluciones, sin tener que pagarle nada a nadie, ni

siquiera a nosotros. Se puede mostrar un modelo BIM literalmente en cualquier parte.

Otro ejemplo: BIM + GIS (Sistema de Información Geográfica). Aquí se utilizó IFC.js y se unió con harp.gl, que es otra librería que existe para Javascript la cual básicamente permite crear aplicaciones GIS tridimensionales. Y solo en 48 horas, se implementó la aplicación. Un nuevo ejemplo de un modelo BIM-IFC cargado en el navegador. Aquí nuevamente el navegador está haciendo de aplicación BIM y está localizando este modelo en 3D de la ciudad de San Francisco (Figura 2). Esto es posible gracias al uso de librerías y códigos de otras personas, que en este caso el código de IFC.js es nuestro y harp.gl, de otras personas. Así, en lugar de inventar la rueda cada vez, ¿por qué no unimos fuerzas y resolvemos los problemas una sola vez? Para mí, la automatización real es la de la revolución industrial, es decir, una automatización donde se crea un sistema que de pronto le hace la vida más fácil a todo el mundo. No a una sola persona. De lo contrario, sería pretender que todos nos vamos a convertir en una fábrica, de las cuales somos el único cliente. A mí eso no me parece realista. Por el contrario, esto es como intentar coger problemas que tenemos todos, que en este caso es crear aplicaciones BIM con una serie de funcionalidades, y resolverlas para todos: resolver los problemas una sola vez.

Figura 2.
Sitio web como visualizador 3D de ciudades, usando IFC.js.



Nota: Demostración de IFC.js en un sitio web. **Fuente:** Imágen gentileza de Antonio Viegas, Enero 2022.

Como ideas finales, me gustaría reiterar que **el BIM es posible fuera de las aplicaciones BIM tradicionales**. No debe tener licencia privada, porque sólo son datos. Además, **el BIM no tiene porqué pertenecer a grandes empresas**. Cada persona puede tener su propuesta de valor y puede competir en igualdad de condiciones. No como ocurre ahora, que si tú entras a competir en ese mercado, siempre estás en

desventaja porque todas las grandes empresas tienen gran cantidad de funcionalidades que tú no te puedes permitir.

Ideas finales

En ese sentido, se puede crear diversas opciones, por ejemplo, una plataforma BIM con estándar –como Google Cloud o Amazon Web Service– utilizando Google Drive e incluso Dropbox. Esto se podría hacer sin problemas, añadiendo un visor web totalmente integrado.

De este modo, lo que estamos haciendo es replicar lo que tienen otras industrias de software y traerlo al mundo BIM. ¿Por qué no podemos tener esto en abierto y gratis para todos? Y así, que cualquiera puede crear sus aplicaciones BIM. Nosotros creemos que ese es el camino a seguir, en lugar de seguir dejando que solo los grandes actores de la industria lo puedan hacer.