

DISEÑO DE UN PROTOCOLO DE GENERACIÓN  
DE ELEMENTOS ESTRUCTURALES  
UTILIZANDO OBJETOS  
BIM



autora: Desiré Vignozzi  
tutor: Agustín Pérez García  
grado en Fundamentos de la Arquitectura  
curso 2016-2017



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA

# ***Diseño de un protocolo de generación de elementos estructurales utilizando objetos BIM***

*Trabajo Final de Grado:*

*Grado en Fundamentos de la Arquitectura*

*Desiré Vignozzi*

*Septiembre 2016*



*Cada día, las palabras Building Information Modeling (BIM) aparecen con más frecuencia en nuestro sector de la Arquitectura. Esto junto a la curiosidad que siempre tuve de saber más de ella, fue la razón de la elección del tema de mi TFG.*

*Esta herramienta, que es ampliamente aceptada y se encuentra en rápida expansión a nivel internacional, nos permite obtener con un formato abierto, llamado Industry Foundation Class (IFC), un proyecto neutral que consiste en un modelo con datos agregados para toda la industria AEC, el cual se utiliza durante el ciclo de vida completo del edificio. Todo ello genera numerosas características ventajosas para cualquier empresa o trabajador de nuestro sector, analizadas y explicadas en el presente trabajo.*

*El objetivo es desarrollar las bases de una aplicación informática capaz de exportar desde Architrave, aplicación no nativa BIM y por lo tanto no conectable a este tipo de archivos, la definición geométrica de dos modelos estructurales utilizando un formato mucho más abierto, multidisciplinar, tridimensional y ventajoso como es el IFC. Esto permitirá intercambiar información sobre esa geometría con otras aplicaciones, orientadas o no al cálculo de estructuras, y clasificar a estos archivos en conectables BIM.*

*Para concluir, los IFC son archivos estándar capaces de poder compartir información con aplicaciones de esta misma categoría, que ocupan menos espacio que cualquier otro formato, que nos evitan la pérdida de información al transferir archivos entre diferentes aplicaciones y que en definitiva, aumentan nuestra productividad.*

Palabras clave:

*Building Information Modeling, Industry Foundation Class, Architrave, AutoLISP, conectable, y estructura.*



*The words Building Information Modeling comes out very frequently in the architectural sector, therefore I was inclined and very interested to deepen my knowledge in this field and select it to draft my TGF.*

*This powerful tool, which is analyzed and explained and currently in a rapid expansion worldwide, allows us to get an open format, the so called Industry Foundation Class (IFC), a neutral file consisting of an aggregated data model for all the AEC industry, which could be used during the whole building's lifecycle. This leads to several benefits for the companies and workers in the sector, analyzed and explained in this work.*

*The purpose is to develop the basis for software applications able to export from Architrave, no native BIM application and therefore no connectable to this kind of files, the geometrical definition of two structural models by using a much more versatile, multidisciplinary, tridimensional and advantageous like the IFC. It would allow to exchange information on this geometry with other applications, aimed or not at the structural calculation and classify these files as BIM connectable.*

*To sum up, these standard files can share information with applications of the same category, taking less space than any other format and preventing the loss of information during files transferences between different software and ensuring, ultimately, an increase in productivity.*

Key Words:

*Building Information Modeling, Industry Foundation Class, Architrave, AutoLISP, conectable, and structure.*



A mi tutor y profesor, *Agustín Pérez García*, por su disponibilidad, interés y ayuda mostrada en la realización del presente trabajo.

A mis padres, *Diana y Fabio*, por el apoyo incondicional, en éste y cualquier proyecto de mi vida.

Gracias!





- **AEC:** Architecture, Engineering & Construction
- **AIA:** American Institute of Architects
- **BIM:** Building Information Modeling
- **BIS:** Building Information Systems
- **BEP:** BIM Execution Plan
- **FM:** Facilities Management
- **IAI:** International Alliance for Interoperability
- **IFC:** Industry Foundation Class
- **LEED:** Leadership in Energy and Environmental Design
- **LOD:** Level of Detail / Development
- **TFG:** Trabajo Final de Grado
- **UPV:** Universidad Politécnica de Valencia



<b>RESUMEN.....</b>	<b>03</b>
<b>ABSTRACT.....</b>	<b>05</b>
<b>AGRADECIMIENTOS.....</b>	<b>07</b>
<b>ACRÓNIMOS UTILIZADOS.....</b>	<b>09</b>
<b>ÍNDICE.....</b>	<b>11</b>
<b>1. INTRODUCCIÓN.....</b>	<b>15</b>
1.1. OBJETIVOS.....	17
1.2. METODOLOGÍA UTILIZADA.....	18
1.2.2. METODOLOGÍA OPERATIVA.....	19
<b>2. CONTEXTO .....</b>	<b>20</b>
2.1. BIM EN ESPAÑA. OBJETIVOS.....	21
2.2. BIM EN UE Y EEUU.....	23
<b>3. MARCO TEÓRICO .....</b>	<b>25</b>
<b>3.1. ORIGEN DEL BIM.....</b>	<b>26</b>
3.1.1. BUILDING SMART.....	27
3.1.2. OBJETIVOS GENERALES DEL BIM.....	29
<b>3.2. BUILDING INFORMATION MODELING.....</b>	<b>30</b>
3.2.1. CARACTERÍSTICAS DEL ENTORNO BIM.....	32
3.2.1.1. APLICACIONES BIM.....	32
3.2.2. NIVELES DE INFORMACIÓN DEL MODELO BIM.....	33
3.2.3. DIMENSIONES BIM.....	38
3.2.4. BENEFICIOS DE LA METODOLOGÍA .....	39

3.2.4.1. MULTIVISTA.....	40
3.2.4.2. INTEROPERABLE Y COMPATIBLE .....	40
3.2.4.3. INTERDISCIPLINAR .....	41
3.2.4.4. PRODUCTIVO.....	41
3.2.4.5. REDUCE TIEMPO Y COSTES.....	41
3.2.4.6. PERMITE FLUJOS DE INFORMACIÓN .....	44
3.2.4.7. PROMUEVE LA EFICIENCIA ENERGÉTICA.....	45
3.2.5. DIFICULTADES DE LA METODOLOGÍA .....	45
<b>3.3. INDUSTRY FOUNDATION CLASS.....</b>	<b>46</b>
3.3.1. DEFINICIÓN Y ORIGEN.....	46
3.3.2. PRINCIPALES CARACTERÍSTICAS.....	48
3.3.3. ORGANIZACIÓN DE LA INFORMACIÓN.....	49
3.3.4. CRONOLOGÍA DE VERSIONES.....	50
3.3.5. SOFTWARE COMPATIBLES.....	51
<b>4. MARCO PRÁCTICO.....</b>	<b>54</b>
<b>4.1. ARCHITRAVE Y SU RELACIÓN CON EL BIM.....</b>	<b>55</b>
4.1.1. ARCHITRAVE Y SUS CARACTERÍSTICAS.....	55
4.1.2. AUTOCAD, ARCHITRAVE Y BIM.....	57
4.1.3. PROGRAMACIÓN EN VISUAL LISP.....	58
4.1.3.1. LA CONSOLA DE VISUAL LISP.....	59
4.1.4. PROGRAMACIÓN PARA ARCHIVOS IFC.....	62

<b>4.2. DISEÑO DEL PROTOCOLO DE GENERACIÓN DE ELEMENTOS ESTRUCTURALES UTILIZANDO OBJETOS BIM.....</b>	<b>64</b>
4.2.1. FUNCIONES LISP UTILIZADAS .....	65
4.2.2. FUNCIONES IFC UTILIZADAS .....	71
4.2.3. EXPLICACIÓN POR FASES DEL PROTOCOLO .....	78
<b>4.3. APLICACIÓN DEL PROTOCOLO CREADO .....</b>	<b>94</b>
4.3.1. CREACIÓN DE UN PÓRTICO EN ARCHITRAVE .....	94
4.3.1.1. ANÁLISIS DOCUMENTO IFC OBTENIDO .....	96
4.3.1.2. VISUALIZACIONES PROGRAMAS BIM .....	104
4.3.2. CREACIÓN DEL CROWN HALL EN ARCHITRAVE .....	110
4.3.2.1. ANÁLISIS DOCUMENTO IFC OBTENIDO .....	113
4.3.2.2. VISUALIZACIONES PROGRAMAS BIM .....	115
<b>5. CONCLUSIONES.....</b>	<b>121</b>
<b>TABLA DE ILUSTRACIONES.....</b>	<b>126</b>
<b>FUENTES .....</b>	<b>131</b>



Fue en el 2010 cuando empecé la carrera de Arquitectura y con ello las asignaturas de dibujo a mano alzada que nos ayudarían a entender y analizar los espacios proyectados. Seguidamente aprendimos a utilizar AutoCAD, que después de haber dibujado a mano alzada, fue para nosotros una ventaja debido al ahorro de tiempo y a los resultados obtenidos. Esto justo fue lo que sucedió en el ámbito de arquitectos en los años 1980 cuando se produjo la difusión global del CAD fundada por Jhon Walker. Este proceso permitió a los profesionales del sector aumentar su productividad y llegar a lograr diseños que sin estas herramientas hubieran sido imposibles. Esta migración del papel al ordenador actualmente ya se ha consolidado, pero el nivel tecnológico de su implementación ha sido muy bajo en general. Esta tecnología basada en la representación, aparte de consumir enormes cantidades de tiempo, es muy dada a la propagación de errores en el diseño, los cuales acaban apareciendo en la fase de producción de obra a expensas del promotor, el contratista o el arquitecto. De ahí que se ha ido investigando en nuevas metodologías que desarrollan el proceso constructivo intentando resolver dichos inconvenientes y dificultades.

*Building Information Modeling (BIM)*, es una metodología de trabajo que está generando un cambio importante en la forma que teníamos, y aún se tiene, de gestionar los proyectos constructivos y la toma de decisiones que éstos conllevan. Ampliamente aceptada ya en el mundo anglosajón y en rápida expansión a nivel internacional, su grado de conocimiento e implantación en España es menor en comparación. Esto no sucede porque se dude de las mejoras que parece aportar dicho método sino también debido, tal vez, a que nuestro sector, el de la construcción, ha estado parado por un tiempo debido a la “burbuja inmobiliaria” vivida durante los años 1997-2007.

Después de la crisis y la escasa actividad del sector, se buscan nuevas ideas, nuevas formas de hacer las cosas que reduzcan el grado de incertidumbre que todo proyecto constructivo lleva inherente, que nos haga más competitivos, que nos permita ofrecer mejores servicios, con mayor garantía en el resultado final y valor añadido.

Por estas y otras tantas razones como el hecho de tener curiosidad en aprender y descubrir una herramienta útil, que aumenta mi productividad, competitividad y que está en pleno auge en España y en todo el mundo, me interesé por el tema del BIM y opté por esta opción para mi TFG.



## 1.1. OBJETIVOS

Los objetivos centrales de este trabajo son, en primer lugar, establecer un **marco teórico** de éste en el que **explicar y analizar la implantación de la tecnología BIM** y cómo ello puede mejorar nuestra competitividad en el sector de la construcción, arquitectura y estructura. Esto supone un estudio de la tecnología detallado para poder conocer todas sus **utilidades, características y cada uno de los agentes intervinientes en el proceso y la viabilidad y vigencia de esta metodología** a lo largo de todo el ciclo de vida del proyecto constructivo. A su vez, explicar y justificar el motivo por el cual sería muy ventajoso en nuestro campo de la arquitectura y la construcción el hecho de compartir información por una única tipología de archivo, este es el IFC, evitando así errores de comunicación y reinterpretación.

En segundo lugar, se encuentra el **marco práctico**, en el cual se pretende principalmente **desarrollar una aplicación informática capaz de exportar desde Architrave** (aplicación No Nativa BIM) **la definición geométrica de dos modelos estructural** (los proyectos de un pórtico sencillo y el del "Crown Hall") **al formato Industry Foundation Class**, perteneciente a la metodología **BIM**. Al exportarlo a IFC se consigue un **formato mucho más abierto** que los utilizados actualmente, multidisciplinar, tridimensional, que ocupa menos memoria, que no se desfasa con el tiempo debido a que es un formato de texto y el cual **permite la interoperabilidad** entre todos los programas BIM y los conectables BIM.

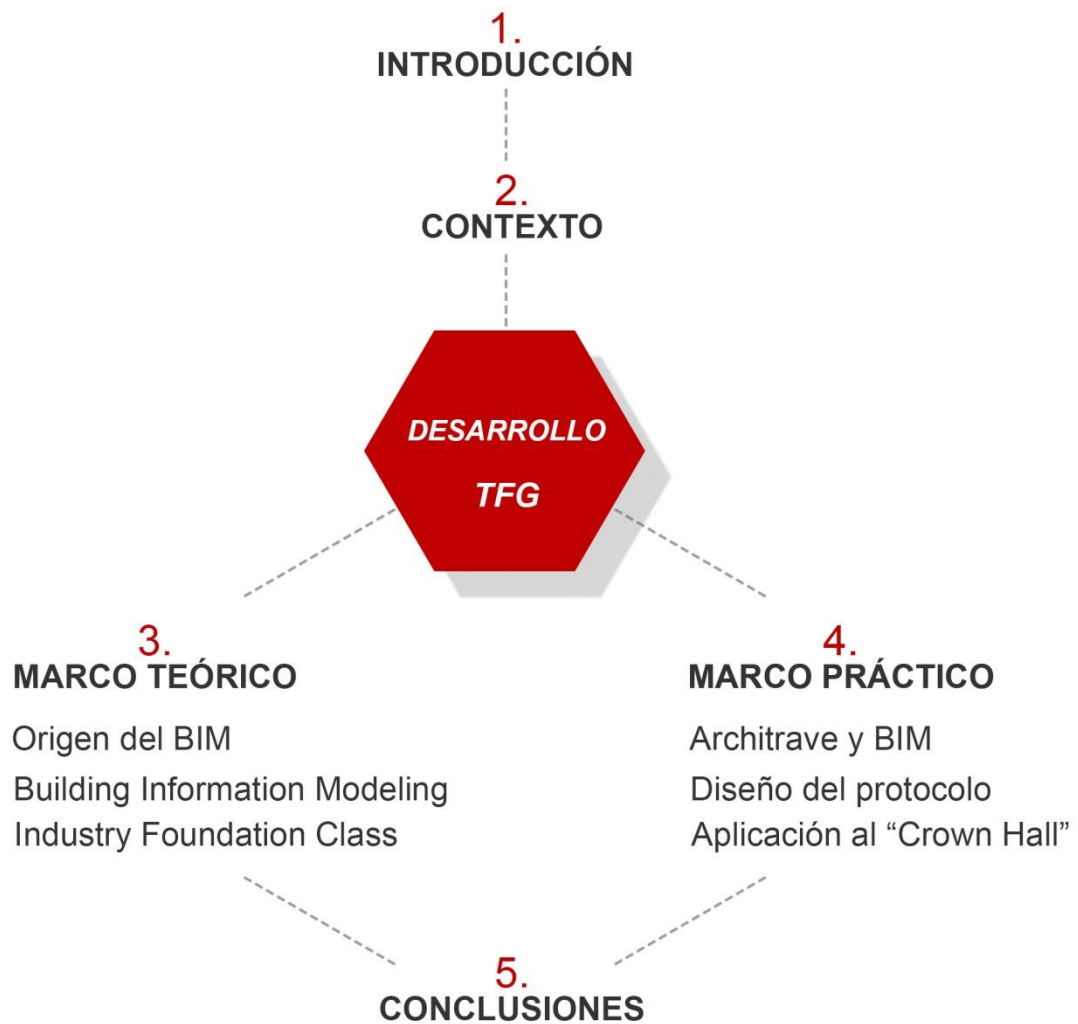
Para llegar a realizar esta labor se precisará un **estudio y análisis de programación** mediante el programa Visual LISP (formato origen) y se precisará de un estudio a través del cual conocer más profundamente cómo funciona y las bases del formato IFC (formato final).

## 1.2. METODOLOGÍA UTILIZADA

Para la realización del actual TFG se ha seguido la metodología que se cita a continuación:

- **Investigación de la situación** de conocimiento y uso de la **metodología BIM** y su repercusión en la sociedad académica así como en el sector de la construcción en Europa, España y el resto del mundo.
- **Recopilación de información del origen del BIM** y sus características generales, beneficios, inconvenientes, aplicaciones, niveles de información, etc.
- **Aprendizaje autónomo** sobre el uso de Autodesk **Revit** para poder experimentar los beneficios y características que nos brindan las aplicaciones BIM y sobre las cuales va el TFG, así como sus dificultades.
- **Búsqueda de información del formato IFC** (en el cual se basa la metodología BIM); su origen, principales **características**, la organización de la información, sus **versiones** y software o aplicaciones compatibles.
- Iniciación en el **aprendizaje** en el mundo de la **programación**, con Visual Lisp con la ayuda de los libros de programación y la asistencia constante a clases y tutorías con profesores especializados.
- **Modelado de algunos proyectos** con **software BIM** (Revit) para realizar las ejemplificaciones de algunos conceptos explicados en el trabajo.
- **Uso de Architrave** para la realización de dos modelos y posteriormente ser exportados a formato IFC y mostrar la compatibilidad con los demás programas BIM.
- **Estudio del formato IFC** para poder conseguir la exportación a esta tipología de archivo y poder experimentar la versatilidad y beneficios que aporta al proceso de realización de un proyecto.
- Uso y aprendizaje del formato IFC **mediante la importación y exportación de archivos en aplicaciones BIM** y conocimiento de las múltiples ventajas que aporta el hecho de la realización de todo un proyecto en un único archivo en lugar de ir creándolo en múltiples.

## 1.2.2. Metodología operativa



**Figura 1.1.**  
*Metodología operativa. (elaboración propia)*



## 2.1. BIM EN ESPAÑA. OBJETIVOS.

El BIM es una metodología de trabajo para la gestión de proyectos que, a través de un modelo digital 3D, reduce costes, acorta tiempos de diseño y producción y mejora la calidad de los proyectos de ingeniería, arquitectura y construcción. Cada día más se es más consciente de los múltiples beneficios que aporta en nuestro sector de la Arquitectura. (

Actualmente, el nivel de implantación de la metodología BIM en España es en general muy bajo con respecto a los demás países. La “burbuja inmobiliaria” y la crisis en nuestro sector han tenido como consecuencia el hecho de que esta nueva era del BIM llegue más tarde a nuestro país. Sin embargo esta situación está progresando.

El día 14 de Julio de 2015, se dio un importantísimo paso en España cuando la ministra de Fomento, Ana Pastor, presidió el **Acto de Constitución de la Comisión para la implantación de la metodología BIM**, donde se anunció que esta metodología se iría introduciendo progresivamente en España.

Nuestro país se encontraba a la cola con respecto a los demás países europeos en cuanto al nivel de penetración de la metodología BIM debido a la no existencia de obligatoriedad del uso de BIM. Pero durante los últimos dos años las cosas en España han empezado a cambiar y tanto la administración pública como las empresas privadas han reaccionado: **el mecanismo de la implantación ya está en marcha.**

Todo ello servirá también para apoyar un mayor y mejor posicionamiento de la industria española en el mundo. (MINISTERIO DE FOMENTO, julio 2015)

### Hoja de ruta



**Figura 2.1.**  
*Hoja de Ruta en España. (www.esbim.es)*

El plan de acción detallado es el siguiente (ESBIM,2016):

1. **Impulsar la implantación de BIM** en la industria de la construcción española a partir de la creación de un Grupo de Trabajo con una participación abierta a todo el sector, tanto público como privado.
2. **Fomentar el uso de BIM** en todo el ciclo de vida de las Infraestructuras.
3. **Sensibilizar a las Administraciones Públicas en el establecimiento de requisitos BIM** en las licitaciones de infraestructuras con el objetivo de reducir sus costes.
4. **Establecer un calendario para la adaptación de normativa para el empleo generalizado de BIM.**
5. **Desarrollar los estándares nacionales** que posibiliten el uso homogéneo de BIM.
6. **Realizar el mapa académico de la formación BIM** en España y promover su inclusión en planes de estudio.
7. **Promover la digitalización de los trabajos** derivados del desarrollo de las infraestructuras, desterrando el formato físico, con el consiguiente ahorro económico y medioambiental.
8. **Fomentar la aplicación de “Open BIM”**, es decir que todas las operaciones relacionadas con BIM se basen en estándares abiertos y universales, interoperables entre sí.
9. **Apoyar un mayor y mejor posicionamiento de la industria española en el mundo a través del empleo de la metodología BIM.**
10. **Afianzar la participación de España en los foros de decisión internacionales.**



**Figura 2.2.**  
Logo ESBIM. ([www.esbim.es](http://www.esbim.es))

Por otro lado, el sector privado, para poder participar en proyectos en el extranjero, se ha visto obligado a ir adoptando el BIM en su flujo de trabajo ya que esto mejora la competitividad en cuanto al acceso a la adjudicación de contratos en otros países donde la crisis inmobiliaria no existe o no es tan aguda.

Desde la administración pública se ha fijado el siguiente **calendario de objetivos**:

- **Objetivo 2017:** *Adopción de estándares IFC, guías, clasificaciones y procesos de entrega del modelo digital pensando en cada fase del proyecto constructivo, de su ejecución, del mantenimiento posterior y de su integración en la ciudad. Definición de unos protocolos comunes para la creación y definición de la información compartida entre los agentes orientados a la plena interoperabilidad entre las partes.*
- **Objetivo 2018:** *Los equipamientos y las infraestructuras públicas de presupuesto superior a 2 M€ deberán producirse en BIM en las fases de Diseño y Construcción. Circunscribir este objetivo en proyectos de obra nueva.*
- **Objetivo 2020:** *Todos los equipamientos y las infraestructuras públicas deberán producirse en BIM en todas las fases: diseño – construcción – mantenimiento. Circunscribir este objetivo a todos los proyectos de obra nueva y rehabilitación.*

## 2.2. BIM EN UE Y EEUU

Todo esto se ha puesto en marcha debido a que somos conscientes de que de manera imprescindible, las empresas y profesionales españoles que queramos participar en proyectos de construcción, reforma, instalación y explotación en otros países como EE.UU., Reino Unido, Centro y Norte de Europa, Emiratos Árabes, Sudéste Asiático, China o Australia, tenemos que implantar el modelo BIM y sus herramientas correspondientes de manera obligatoria para acceder a licitaciones, contratos y colaboraciones.

Los países escandinavos y anglosajones, Alemania, Holanda, Colombia, Chile o China, entre otros, son algunos de los ejemplos de países que ya han implementado **estándares BIM a nivel gubernamental**. (Blog AULATEMÁTICA, abril 2016)



Implantación de BIM a nivel Mundial. España se unirá en 2018

**Figura 2.3.**

*Implantación del BIM a nivel Mundial. (<http://blog.aulatematica.com/>)*

En el **caso británico**, que empezó a adoptar la estrategia BIM en 2006, la implantación de esta metodología ha supuesto un **ahorro en costes de alrededor de 15-20%**, y gracias al impulso de la administración pública, ha crecido tan rápidamente, que en 2016 el uso obligatorio de BIM ya es una realidad.

En el 2014, **la Unión Europea** ya solicitó a los países miembros que consideraran la incorporación de la metodología BIM para afrontar la actualización y modernización de las normativas de contratación y licitaciones públicas.

El BIM permite mejorar también, entre otras cosas, los procesos de rehabilitación de edificios, el control de la viabilidad económica de las inversiones y la limitación del impacto ambiental. El hecho de que esta metodología genere beneficios ya es una realidad. (Blog AULATEMÁTICA, abril 2016)





### 3.1. ORIGEN DEL BIM

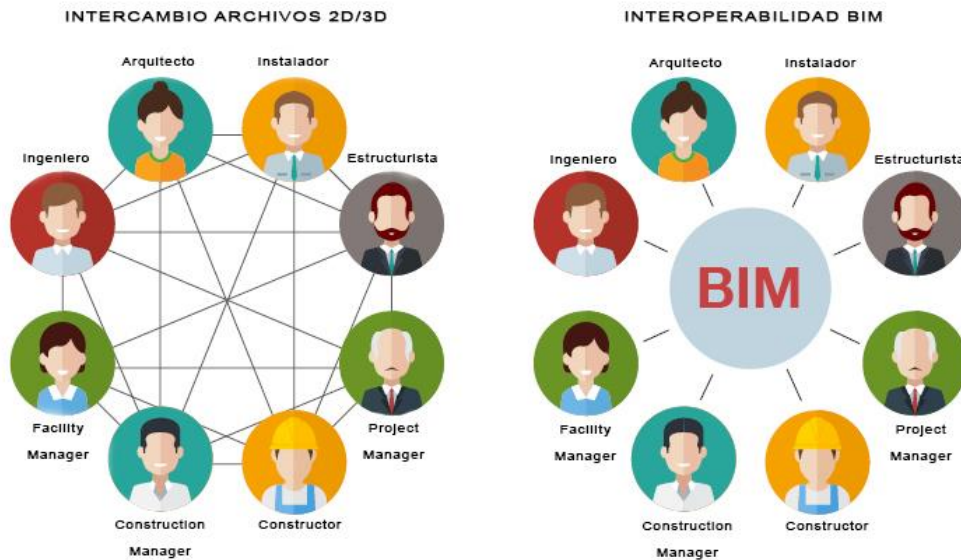
Hoy en día, en cualquier proyecto arquitectónico hay una gran cantidad de información que hay que compartir con los diferentes arquitectos, ingenieros y diseñadores. Esta información al no ser en el mismo tipo de archivo, sino cada uno diferente dependiendo del programa con el que se realice, crea inevitablemente una **distorsión de información**. Es decir, este método obliga a reinterpretar la información que se nos facilita, a pasarla por **el filtro de nuestro entendimiento**. Y no siempre el mensaje que pretenden que nos llegue y el que nosotros interpretamos coinciden en su significado.

Un proyecto inmobiliario constituye un sistema de información y comunicación en sí mismo, por lo tanto es inviable el hecho de que los distintos participantes compartan su información de forma electrónica en ficheros de distintos formatos y extensiones, generados por diferentes aplicaciones informáticas que son incompatibles entre sí. De este modo, la información que nosotros aportemos a estos documentos no quedará insertada en los mismos, sino que se convertirá en anotaciones manuscritas o comentarios, que no formarán parte del formato original.

De ahí que lo que se intentaba encontrar era un método que en lugar de describirte el proyecto a través de modelos no conectados presentados en 2d como con CAD, ideara una **nueva generación de aplicaciones que trabajasen con bases de datos**, que en vez de con miles de representaciones literales (2D Y 3D), contuvieran objetos paramétricos con información multidisciplinar. Estas bases de datos que supusieron una revolución conceptual y tecnológica se conocen como Modelos de Información y en el caso del modelado de edificios BIM (Building Information Modeling).

**Compartir en lugar de intercambiar.** Recoger la información aportada por distintos profesionales al mismo tiempo en un **único contenedor** de la información del edificio proyectado al que todos los agentes participantes participan de diversas formas. Se conectan a la base de datos desde la cual se accede a ese contenedor único en el cual toda la información está relacionada y a través de la cual se permite ir creando el edificio iterativamente, realizar cambios paramétricos (cualquier cambio realizado, lo recalcula y modifica en el interior del proyecto), evaluaciones globales, detección de interferencias, etc. en tiempo real. A su vez, con la ventaja añadida de trabajar siempre en formato visual 3D, sobre un único repositorio de planos 2D/3D. (FUENTES GINER, B., 2015)

Y con esto se formalizó la interoperabilidad en la gestión de la información y del conocimiento del proceso constructivo.



**Figura 3.1.**  
 Interoperabilidad BIM. (<http://biblus.acca.it/ifc-cose-e-a-che-serve/>).

### 3.1.1. BuildingSMART

En el 1994 Autodesk junto a doce empresas estadounidenses que adoptaron el nombre de **Industry Alliance** iniciaron un consorcio de industrias que les asesorasen en el desarrollo de una parte del lenguaje de programación que permitiese soportar el **desarrollo integrado de distintas aplicaciones**. La primera de las ideas consistía en crear un **lenguaje común** que utilizarasen todas las aplicaciones y que fuese útil para resolver los distintos cálculos necesarios para definir el proyecto. Además, dicho lenguaje debía ser **capaz de contener toda la información significativa del edificio** y de cada uno de los elementos que lo componen y **permitir la agregación de más datos** e información a medida que era necesario. (FUENTES GINER, B., 2015)

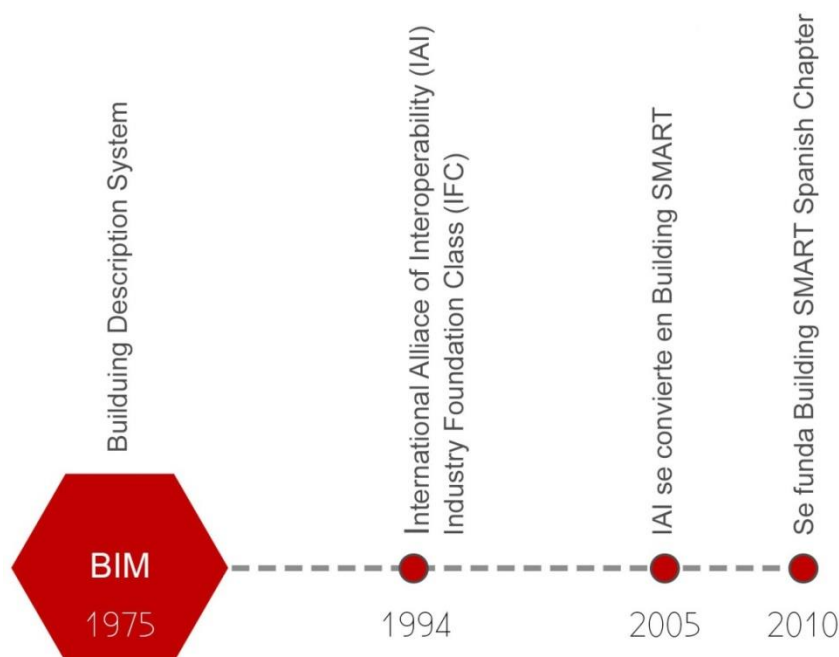
Este conjunto de empresas, en el 1997 cambió su nombre por el de **International Alliance for Interoperability (IAI)** y se constituyó en una organización sin ánimo de lucro con el objetivo fundamental de publicar en formato abierto, para todo aquel interesado, las **Industry Foundation Class (IFC)**, como un producto internacional, neutral consistente en un modelo con datos agregados para toda la industria AEC que pudiera ser utilizado durante el ciclo de vida completo del edificio. BuildingSMART fomenta la interoperabilidad y la cooperación entre agentes del sector de la construcción a través de la metodología BIM basada en estándares abiertos:

En el 2005, en su reunión anual que tuvo lugar en Noruega, el IAI decidió cambiar su nombre por el que hoy en día es conocido: **buildingSMART**. Actualmente tiene capítulos organizados por regiones que comparten el mismo idioma: Australasia, Benelux, Canadá, China, el grupo de países de habla francesa, el grupo de países de habla alemana (Alemania, Austria y Suiza), Hong Kong, Italia, Japón, Corea, Oriente Medio, EEUU, los países nórdicos (Dinamarca, Finlandia y Suecia), Noruega, Singapur, España, Gran Bretaña e Irlanda.

**En España su capítulo fue fundado en 2010.** Cualquier profesional o empresa del sector de la construcción puede ser miembro de buildingSMART. La última versión actualizada de este documento es datado en el 2014 (**BUILDING SMART Spanish Chapter**).

Los beneficios que se alcanzan mediante el BIM son los siguientes (buildingSMART 2016):

- **Comunicación:** Favorece la comunicación entre los diferentes agentes y sistemas
- **Internacionalización:** Facilita el acceso a otros mercados.
- **Competitividad:** Mejora continua a partir de la repetitividad de los procesos
- **Valoración del cliente:** aval frente a muchos clientes, como la Administración Pública.



**Figura 3.2.**  
Cronología BIM. (elaboración propia)

## 2.1.2. OBJETIVOS GENERALES DEL BIM

La propiedad y el modelado de la construcción apuntan a soportar un ciclo completo del diseño y la construcción que sea de alta calidad, eficiente, seguro y conforme con un desarrollo sostenible. Los modelos del edificio (BIM) se utilizan a lo largo de todo el ciclo de la vida del edificio, empezando en el diseño inicial, continuando durante la construcción e incluso más allá, hasta el uso del edificio y la gestión de equipamiento (FM Facilities Management) una vez que el proyecto de construcción ha finalizado.

Los **modelos del edificio con información** (BIM) permiten obtener una serie de **RESULTADOS** como son (buildingSMART España 2014):

- Dar soporte a las decisiones de inversión, comparando la funcionalidad, el alcance y los costes de las diferentes soluciones.
- Análisis comparativo de los requisitos energéticos y medioambientales, para elegir las opciones de diseño y objetivos para el seguimiento posterior de la explotación del edificio y sus servicios.
- Visualización del diseño y estudios de viabilidad de la construcción.
- Mejora del aseguramiento de la calidad y del intercambio de datos para hacer el proceso de diseño más efectivo y eficiente.
- Uso de los datos del proyecto del edificio durante las operaciones de construcción y explotación y mantenimiento.

Como bien se indica en la **Guía de Usuarios BIM** antes comentada; para obtener un modelo satisfactorio para todos los agentes, deben establecerse prioridades y objetivos específicos en el proyecto para el uso del modelo. Estos requisitos específicos de proyectos deberían ser definidos y documentados de acuerdo a las bases generales establecidas en esta serie de publicaciones y que se suelen concretar en un documento específico del proyecto denominado "Plan de Ejecución BIM (BIM Execution Plan) BEP"

Los **OBJETIVOS generales de los BIM** incluyen:

- Dar soporte a la toma de decisiones del proyecto.
- Dar una base a los aspectos contractuales alineados con los objetivos del proyecto utilizando el modelo de información del edificio como referencia.
- Visualizar soluciones de diseño.
- Asistir durante la fase de diseño y coordinar entre distintos diseños.
- Incrementar y asegurar la calidad del proceso de construcción y el producto final.

- Analizar con más detalle los procesos durante la fase de construcción permitiendo optimizarlos y hacerlos más eficientes.
- Mejorar la seguridad durante la fase de construcción y explotación del edificio.
- Dar soporte a los análisis de costes del proyecto y del ciclo de vida del edificio.
- Permitir la gestión y la transferencia de datos del proyecto durante la operación.

### 3.2. BUILDING INFORMATION MODELING

BIM, es el acrónimo que significa **Modelado de la Información de la Construcción**, aunque detrás del nombre se esconden múltiples definiciones dependiendo de la fuente consultada, de ahí que vaya a hacer un recopilatorio de las más importantes y que mejor lo definen según mi punto de vista.

Definición según **buildingSMART Spanish Chapter**:

*“**Building Information Modeling (BIM)** es una metodología de trabajo colaborativa para la creación y gestión de un proyecto de construcción. Su objetivo es centralizar toda la información del proyecto en un modelo de información digital creado por todos sus agentes. BIM supone la evolución de los sistemas de diseño tradicionales basados en el plano, ya que incorpora información geométrica (3D), de tiempos (4D), de costes (5D), ambiental (6D) y de mantenimiento (7D). El uso de BIM va más allá de las fases de diseño, abarcando la ejecución del proyecto y extendiéndose a lo largo del ciclo de vida del edificio, permitiendo la gestión del mismo y reduciendo los costes de operación.”*

Para **Autodesk**, compañía del programa AUTOCAD y REVIT ARCHITECTURE entre otros, ambos software empleado para la realización de este trabajo final de grado, el BIM es:

*“El Modelado de Información para la Edificación (BIM – Building Information Modeling) es un método innovador para facilitar la comunicación entre los sectores de la arquitectura, la ingeniería y la construcción. Con BIM, arquitectos e ingenieros generan e intercambian información de manera eficiente, crean representaciones digitales de todas las fases del proceso de construcción y simulan el rendimiento en la vida real, lo que perfecciona el flujo de trabajo, aumenta la productividad y mejora la calidad...”*



### 3.2.1. CARACTERÍSTICAS DEL ENTORNO BIM.

El BIM tiene como ventaja la simulación del proceso proyecto-construcción en un entorno visual 3D. Esta simulación tiene como motor un paquete de software capaz de interpretar la base de datos que contiene la información del modelo virtual del edificio. El software recomendado debería tener una certificación mínima de IFC 2X3, que significa que el software puede exportar el modelo en el formato indicado (Requisito técnico que especifica la Guía uBIM).

Tener, previamente al edificio construido, un modelo virtual del edificio implica que es posible ensayar el proceso constructivo, experimentar el comportamiento del edificio frente a cualquier tipo de acciones y realizar los ajustes a la documentación del proyecto antes de iniciar su construcción real.

Las características de BIM que hacen posible la **adición, gestión y compartición de la información** del edificio y la **simulación del proceso constructivo** son: (Coloma 2008)

- **Coordinada:** Modelo tridimensional creado como **contenedor único** de toda la información, la cual se comparte entre todos los agentes del proyecto, de ahí que sea **multidisciplinar** y **multiusuario**.
- **Coherente:** Un único modelo y múltiples vistas evitando así informaciones erróneas.
- **Computable:** Al tener el modelo en 3d se puede realizar un análisis de **cuantificación** y **calificación** de materiales para hacer un previo cómputo de cantidades necesarias para el proyecto.
- **Continuada:** La información del edificio en el BIM no cesa debido a que cubre todo su ciclo de vida: proyecto, construcción, rehabilitación y demolición.
- **Paramétrica:** La información es paramétrica, esto es editable y agregable. Cada elemento constructivo del proyecto tiene entidad propia, relaciones con respecto a otros elementos constructivos del proyecto, características y situación concreta en el proyecto.

#### 3.2.1.1. APLICACIONES BIM.

En cuanto a las aplicaciones BIM hay **dos tipologías** generales: las **aplicaciones BIM (Nativas BIM)** propiamente dichas y otras que no se ajustan a la anterior definición pero que sí que pueden conectarse a ellas (*No nativas BIM*).



En cuanto a las **aplicaciones BIM**, son aquellas que emplean como entidades de trabajo principal objetos paramétricos de cualquier disciplina que son capaces de relacionarse entre ellos y de los que se puede extraer diversos tipos de información, entre los que se incluye representaciones gráficas pero también alfanuméricas.

Por otro lado se encuentran las **aplicaciones conectables**, que son aquellas que están preparadas para conectarse con aplicaciones BIM y extraer de sus modelos aquella información que les sea más útil para sus fines.

Este apartado se explicará más profundamente con posterioridad cuando se hable del IFC y sus posibilidades de intercambio, mediante los programas *Nativos BIM* y *No Nativos BIM*.

### **3.2.2. NIVELES DE INFORMACIÓN DEL MODELO BIM.**

En una *metodología BIM*, el **modelo** es el “contenedor” central donde todos los agentes de la construcción que intervienen en él (AEC Industry) pueden acceder, depositar y recoger información.

Unas de las características principales de la metodología BIM es que el **BIM Manager** o coordinador BIM sea quien se ocupe de la integridad del modelo BIM y facilite el intercambio entre los diversos agentes, todo esto a través del **Plan de ejecución BIM** (BEP: BIM Execution Plan)

Por tanto, uno de los principales aspectos del **Coordinador BIM** (*BIM Manager*) será, a través del **Plan de ejecución BIM** (*BEP: BIM Execution Plan*), velar por la integridad del modelo BIM y facilitar el intercambio de información entre diversos actores.

Dependiendo del fin para el que se comparta la información del modelo, se requerirá de mayor o menor contenido de información; de ahí que hiciera falta establecer un **nivel de detalle común** entre todos los agentes intervinientes en cada una de las fases del proyecto. Quien deberá establecer de antemano este nivel de detalle y requisitos en cada etapa del proyecto será el *Coordinador BIM*.

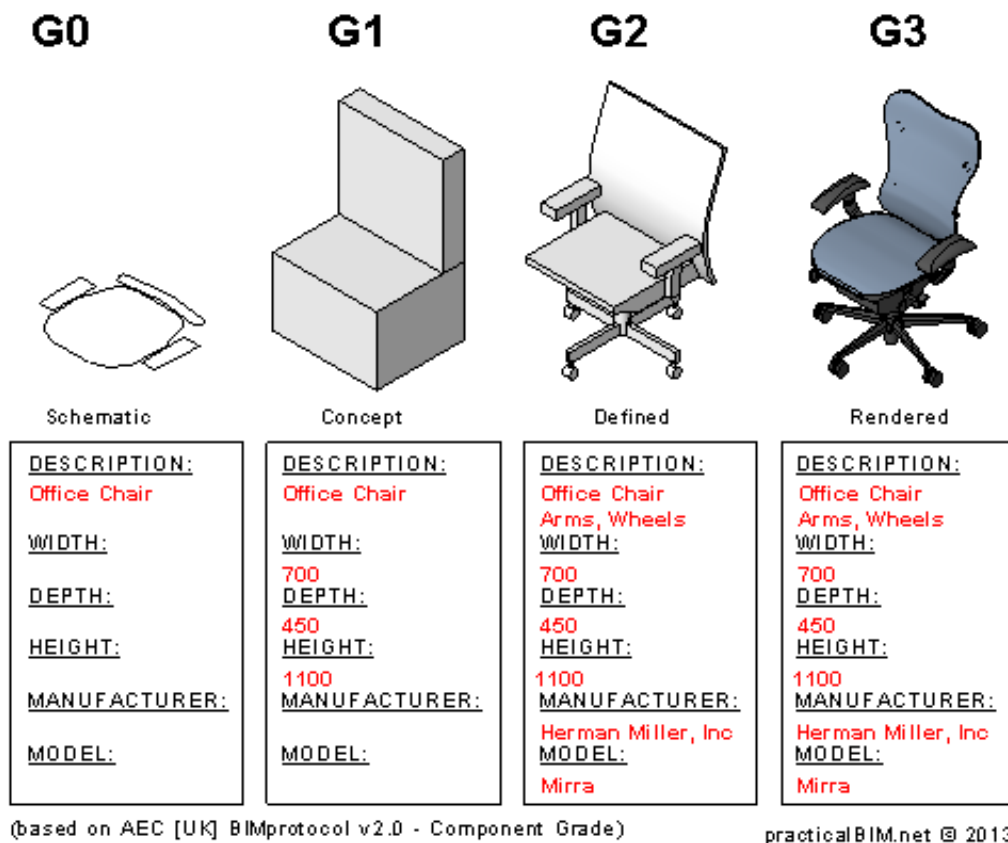
Una guía que ayuda a establecer estos diferentes niveles de definición son los **Niveles de Detalle** (LoD- Level of Detail). La primera empresa en introducir este concepto fue *Vico Software* (actualmente en el grupo Trimble y originalmente vinculada a Graphisoft) debido a que lo necesitaban para poder desarrollar su software de mediciones y presupuestos utilizando BIM.

Sin embargo, en el mundo del BIM el acrónimo **LoD** hace referencia a **dos definiciones distintas**:

• **NIVEL DE DETALLE (LoD)**

Nivel de Detalle o *Level of Detail* hace referencia a la **cantidad de información** que posee un modelo BIM, no por ello a la calidad de ésta ya que es siempre precisa; debido a que toda información que facilita BIM sea menor o mayor es acertada y certera. (ALONSO MADRID, J. 2013)

## LEVEL of DETAIL



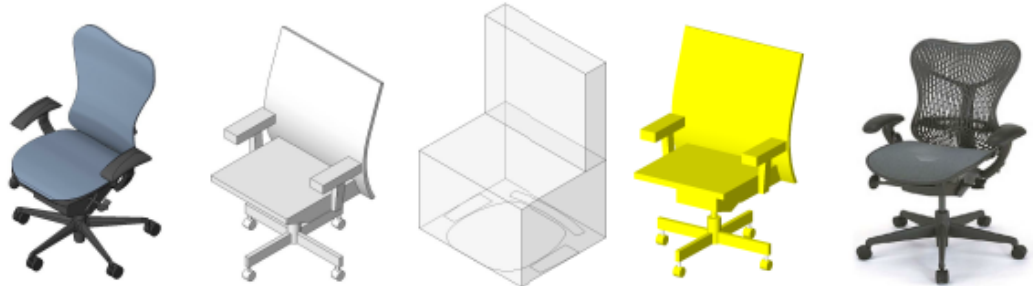
**Figura 3.4.**  
Representación Level Of Detail. (practicalBIM.net)

• **NIVEL DE DESARROLLO (LoD)**

Nivel de Desarrollo o *Level of Development Specification* se trata de un documento de **BIMForum** el cual se basa en documento *AIA G202-2013 Building Information Modeling Protocol Form*, que pretende valorar el **motivo por el cual sirve la información contenida** en lugar de la cantidad de la información. Es decir, la cantidad y calidad estarán predefinidas en cada fase teniendo en cuenta el trabajo que se va desarrollar con él. (ALONSO MADRID, J. 2013)

## LEVEL of DEVELOPMENT

**LOD 100      LOD 200      LOD 300      LOD 400      LOD 500**



Concept (Presentation)    Design Development    Documentation    Construction    Facilities Management

<u>DESCRIPTION:</u> <b>Office Chair</b> Arms, Wheels <u>WIDTH:</u>  <u>DEPTH:</u>  <u>HEIGHT:</u>  <u>MANUFACTURER:</u> Herman Miller, Inc. <u>MODEL:</u> Mirra <u>LOD:</u> <b>100</b>	<u>DESCRIPTION:</u> <b>Office Chair</b> Arms, Wheels <u>WIDTH:</u> <b>700</b> <u>DEPTH:</u> <b>450</b> <u>HEIGHT:</u> <b>1100</b> <u>MANUFACTURER:</u> Herman Miller, Inc. <u>MODEL:</u> Mirra <u>LOD:</u> <b>200</b>	<u>DESCRIPTION:</u> <b>Office Chair</b> <b>Arms, Wheels</b> <u>WIDTH:</u> <b>700</b> <u>DEPTH:</u> <b>450</b> <u>HEIGHT:</u> <b>1100</b> <u>MANUFACTURER:</u> Herman Miller, Inc. <u>MODEL:</u> Mirra <u>LOD:</u> <b>300</b>	<u>DESCRIPTION:</u> <b>Office Chair</b> <b>Arms, Wheels</b> <u>WIDTH:</u> <b>685</b> <u>DEPTH:</u> <b>430</b> <u>HEIGHT:</u> <b>1085</b> <u>MANUFACTURER:</u> <b>Herman Miller, Inc</b> <u>MODEL:</u> <b>Mirra</b> <u>LOD:</u> <b>400</b>	<u>DESCRIPTION:</u> <b>Office Chair</b> <b>Arms, Wheels</b> <u>WIDTH:</u> <b>685</b> <u>DEPTH:</u> <b>430</b> <u>HEIGHT:</u> <b>1085</b> <u>MANUFACTURER:</u> <b>Herman Miller, Inc</b> <u>MODEL:</u> <b>Mirra</b> <u>PURCHASE DATE:</u> <b>01/02/2013</b>
--	---	--	---	--

(Only data in red is useable)

practicalBIM.net © 2013

**Figura 3.5.**  
 Representación Level Of Development. (practicalBIM.net)

En el documento **LOD Specification 2015** se definen en el apartado *Fundamental LOD Definitions* de la siguiente manera los **niveles de información básica de un modelo BIM**:

**LOD 100** : Diseño representado con un **símbolo** o con cualquier otra **representación genérica**, es decir, conceptual y general. En este nivel no se tendrá en cuenta la precisión ya que no serán representaciones geométricas, solo meros símbolos mostrando algún componente del modelo pero no su forma, tamaño o ubicación precisa, de ahí que la **información es aproximada**.

**LOD 200** : Diseño que aporta una **información general con magnitudes**; cantidades aproximadas, tamaño, forma, localización y orientación. En este nivel también se puede añadir **información no-geométrica**.

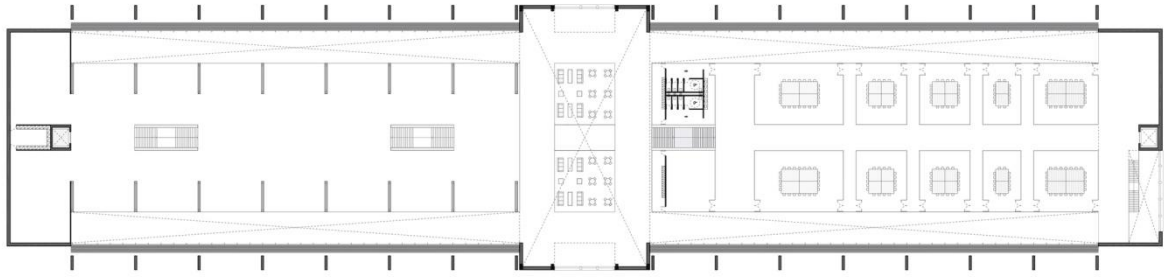
**LOD 300** : Diseño gráficamente representado con **información y geometría precisa**; cantidad, tamaño, forma, localización y orientación. La información no gráfica también se puede añadir al modelo. En este nivel las medidas y características se pueden medir directamente desde el modelo sin hacer falta mirar la información no gráfica adjunta.

**LOD 350** : En este nivel a diferencia del anterior también se definen las **piezas necesarias para la coordinación del elemento con otros elementos cercanos o adjuntos**. Diseño representado gráficamente como un sistema específico, definido en términos de cantidad, tamaño, forma, ubicación, orientación e interfaces con otros sistemas del edificio.

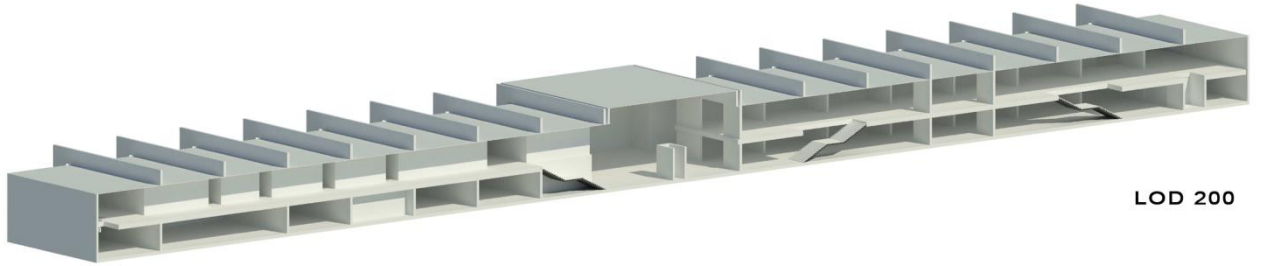
**LOD 400** : Diseño que contiene el detalle preciso para la fabricación o construcción debido a su alta exactitud. Además de la información del anterior nivel, este tiene la información necesaria para la fabricación, montaje, ensamblaje y detalles necesarios para la construcción del edificio.

**LOD 500** : Nivel que **representa el proyecto ya construido** ya que el modelo es adecuado para el mantenimiento y explotación del edificio.

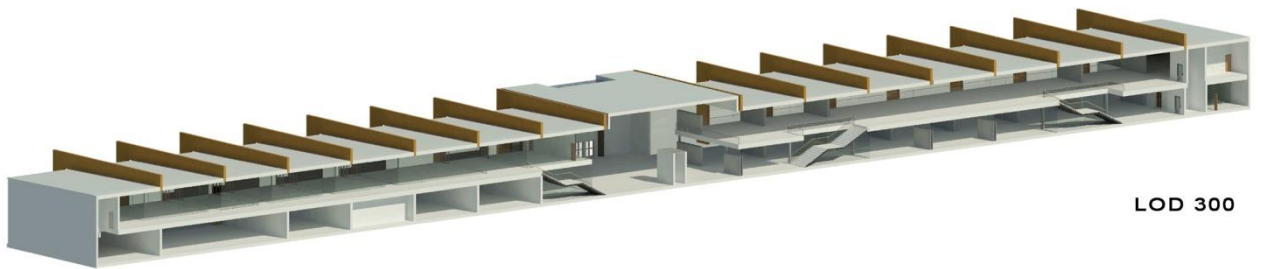
*Figura 3.6. (página siguiente)  
Proyecto propio que muestra los diversos niveles de detalle LOD  
(elaboración propia)*



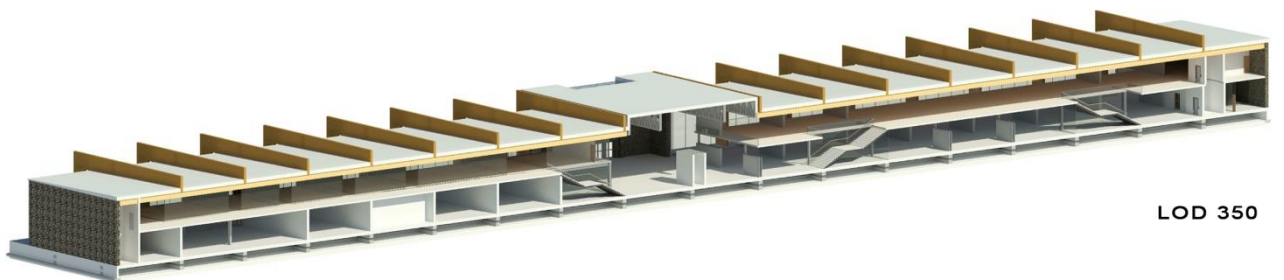
LOD 100



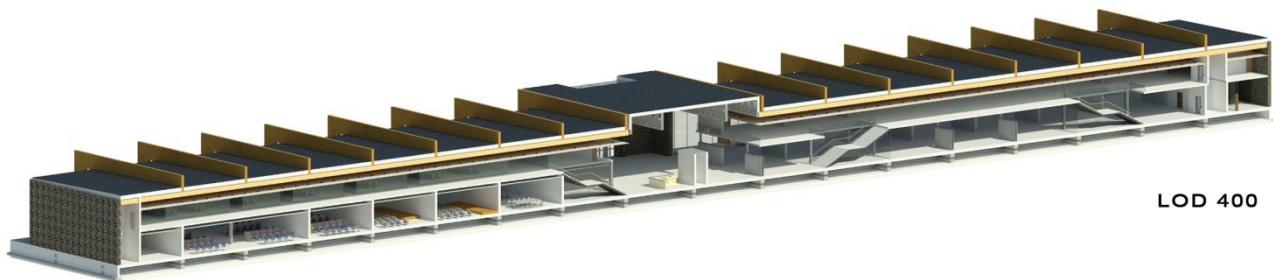
LOD 200



LOD 300



LOD 350



LOD 400

### 3.2.3. DIMENSIONES BIM.

Como bien he explicado anteriormente, un modelo BIM va mucho más allá que un simple modelo en 3D, ya que contiene mucha más información que aquella visible a primera vista. Seguidamente clasificaré todo aquello que abarca BIM hablando dimensiones de manera más detallada:

**2D** : Dibujos creados en **AutoCAD**.

**3D** : **Diseño espacial**. Con la llegada del BIM es cuando se puede considerar que las 3 dimensiones van mucho más allá que una simple representación en 3d, ya que éste nos garantiza una información paramétrica del modelo virtual que nos añade información mucho más amplia.

**4D** : **Tiempo**. Esta dimensión permite una planificación, control de ejecución y simulación de las fases del proyecto y de la obra. Se asigna una secuencia constructiva de aquello que se irá realizando para optimizar las operaciones y con ello aumentar la productividad.

**5D** : **Costes**. El BIM da la oportunidad de controlar los costes y planificarlos. Se tendrá conocimiento de costes unitarios, por partidas o por tablas planificadas por nosotros mismos.

**6D** : **Sostenibilidad – Green BIM**. Permite realizar un análisis virtual de energía y seguimiento de los elementos sostenibles. Seguimiento LEED (La certificación LEED (Leadership in Energy and Environmental Design), promovida por el Consejo Norteamericano de Edificios Verdes (USGBH), es un estándar voluntario y la referencia en edificación sostenible a nivel mundial, aceptada globalmente como norma en cuanto al diseño, edificación y gestión de edificios sostenibles y ecoeficientes.).

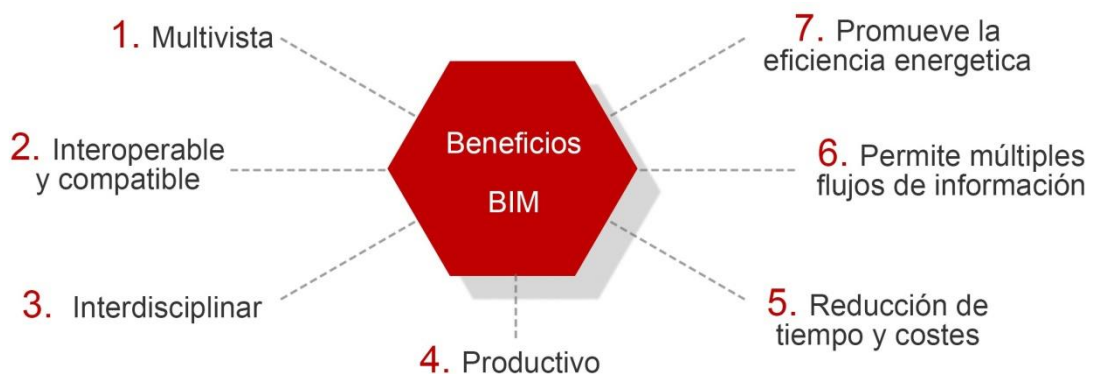
**7D** : **Mantenimiento**. Esta dimensión se utilizará una vez ya tengamos el edificio construido (también llamado “as built”). Permite conocer el estado actual de los elementos necesarios de mantenimiento, como pueden ser las instalaciones dotándolas de información necesaria para garantizar su buen comportamiento y mantenimiento.



**Figura 3.7.**  
Dimensiones BIM. (<http://www.bimbarcelona.com/bim-es-el-presente-no-el-futuro/>)

### 3.2.4. BENEFICIOS DE LA METODOLOGÍA.

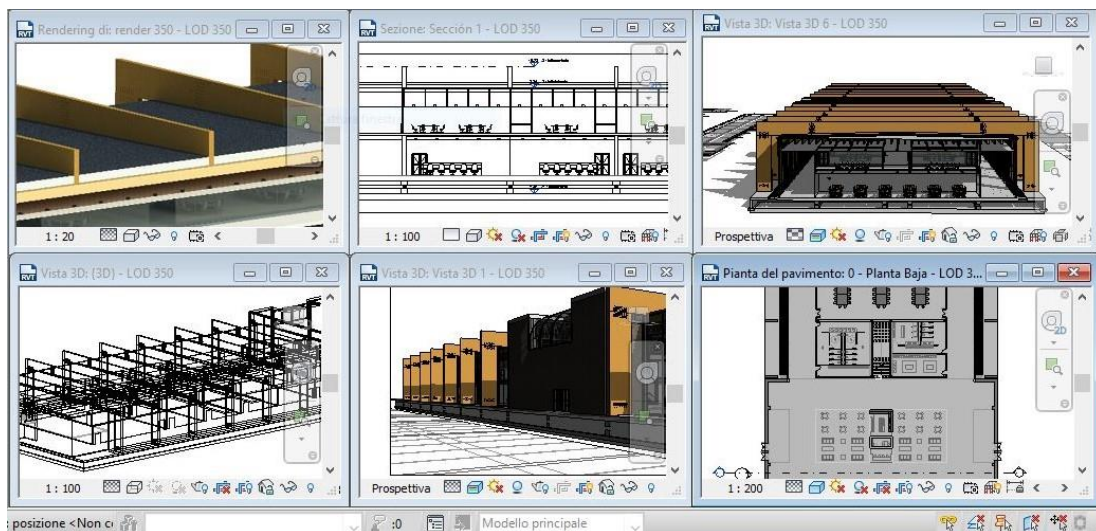
La metodología BIM aporta mayoritariamente beneficios a aquellos agentes que lo utilizan sobre todo en los campos de la arquitectura, construcción, estructura, ingeniería y diseño. Los siguientes beneficios, por lo tanto, recaen sobre los autores/arquitectos del modelo, promotores, constructores y empresas colaboradoras:



**Figura 3.8.**  
Beneficios del BIM (elaboración propia).

### 3.2.4.1. MULTIVISTA.

A diferencia del CAD, en el cual creábamos las diferentes vistas 3D del proyecto independientes entre ellas, en el caso del BIM realizamos un **único modelo/contenedor de información** desde el cual podemos obtener las **múltiples vistas** que precisemos a tiempo real. La visualización de las vistas 3D que se obtienen del mismo modelo hacen que éstas sean más exactas y fiables; evitando fallos e incoherencias que antes eran inevitables en alguna ocasión en el CAD debido a que eran dibujos independientes. Por lo tanto, esto significa que se experimenta una **mejora en cuanto a la comunicación y comprensión** del proyecto.



**Figura 3.9.**  
BIM Multivista. (elaboración propia de un proyecto en Revit)

### 3.2.4.2. INTEROPERABLE Y COMPATIBLE.

Esta característica es una de las principales del mundo BIM; el poder **globalizar e integrar toda la información de los diferentes agentes** del proyecto sobre la que pueden trabajar a tiempo real. Esta integración a su vez genera múltiples ventajas como por ejemplo que permite su correcta comunicación, detección de interferencias y la generación automática de toda la información del proyecto, de la cual se obtiene un mejor resultado.



La interacción entre los diferentes agentes se puede realizar desde **diferentes o iguales aplicaciones**. Suele ser lo más habitual el hecho de que se utilicen diferentes aplicaciones debido a que se juntan diferentes disciplinas en un mismo modelo, de ahí que se utilice el formato por excelencia del BIM; el **IFC** (en el cual profundizaré con posterioridad), éste supone el lenguaje de definición y agregación de datos que hace posible la interoperabilidad, compatibilidad y desarrollo de la documentación de proyecto procedente de distintos profesionales.

### **3.2.4.3. INTERDISCIPLINAR.**

El BIM es interdisciplinar debido al **conjunto de disciplinas que generan el modelo** y añaden la información a ese mismo contenedor/archivo el cual pueden compartir, esto hace que la información sea bidireccional y paramétrica y permite un mejor seguimiento del ciclo de vida del edificio. En el caso de la arquitectura, el conjunto de profesionales o disciplinas que colaboran son: el arquitecto, el promotor, el ingeniero, las empresas de instalaciones y todas aquellas que se precisen dependiendo del tipo de proyecto que se vaya a llevar a cabo.

### **3.2.4.4. PRODUCTIVO.**

El hecho de que el BIM permita la pre-visualización y pre-construcción del proyecto que se pretende realizar anticipa muchas cuestiones que permiten programar y planificar la obra virtualmente, **eliminando errores de descoordinación de la información** y por consiguiente, aumentando la productividad si se comparara con métodos anteriores.

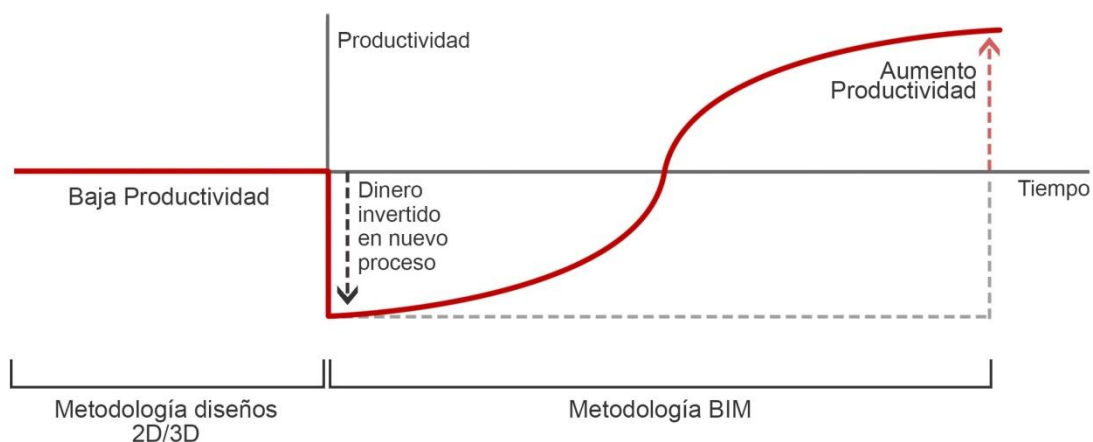
La mecanización de tareas repetitivas, la integración de información y la interoperabilidad entre aplicaciones BIM permiten elaborar una **mejor documentación de proyecto** y, por tanto, las posibilidades de obtener un **mejor producto al final del proceso y por consiguiente mayor competitividad**. (FUENTES GINER, B.2014).

### **3.2.4.5. REDUCE TIEMPO Y COSTES.**

El hecho de que todos los agentes del proyecto añadan su propia información a un mismo archivo o contenedor común, ya es un ahorro de tiempo importante debido a

que ya no hará falta unir dicha información, interpretarla y adaptarla a un mismo archivo.

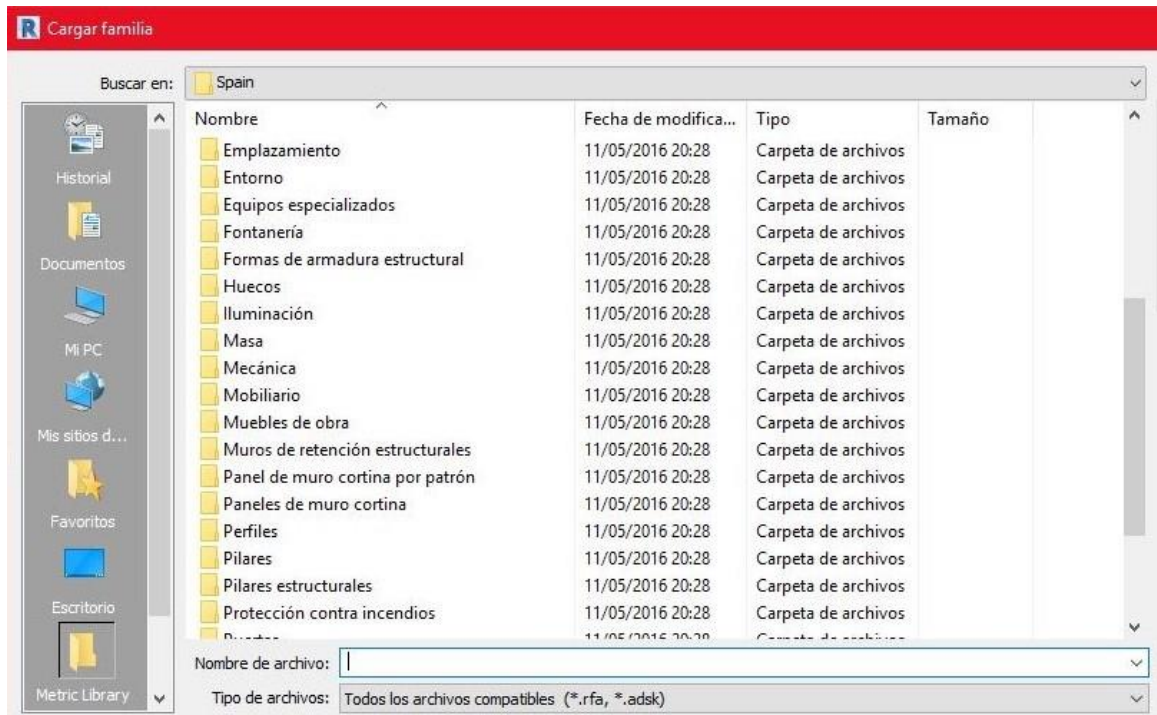
Por otro lado, el hecho de **generar el modelo directamente en 3D** y obtener todas las vistas necesarias desde el mismo nos **supone obtener la misma documentación que se obtenía antes en menos tiempo o bien**, lo que es lo mismo, **en el mismo tiempo obtener más documentación**. Aumentando productividad y por consiguiente reduciendo el tiempo y los costes debidos al proceso constructivo.



**Figura 3.10.**  
Productividad del diseño durante la implementación BIM. (elaboración propia)

Por último, en este apartado es importante recordar que todas las aplicaciones nativas BIM nos ofrecen una **amplia biblioteca de familias** ya realizadas para poder ubicar en nuestros proyectos. Esta característica también permite reducir el tiempo de realización de dichos proyectos y por consiguiente el aumento de nuestra productividad.

Esta opción se encuentra disponible al marcar la opción de “insertar” y seguidamente “cargar familia”. Hay varias carpetas dependiendo de aquello que se quiera añadir. Las imágenes a continuación muestran en concreto algunas de las tipologías generales de familias que Revit ofrece y seguidamente las familias de pilares estructurales de acero disponibles.



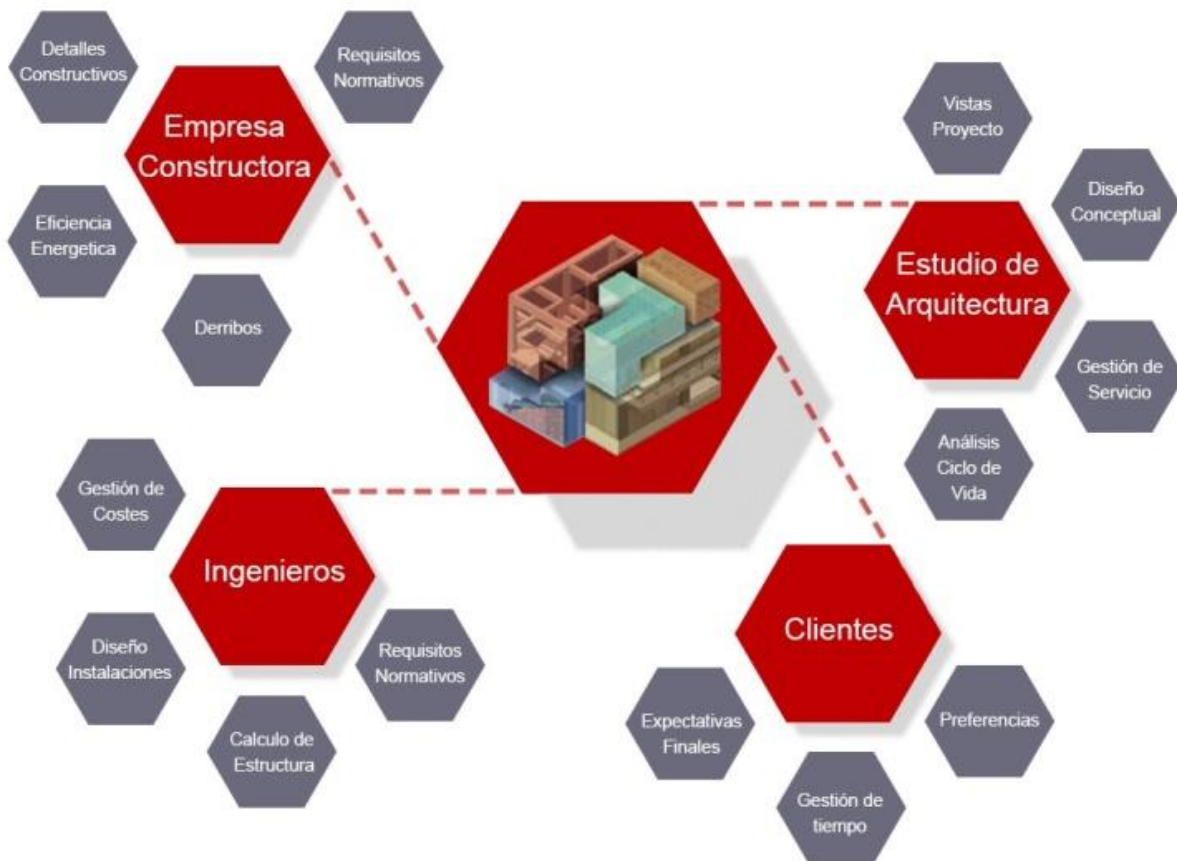
**Figura 3.11.**  
Famílias de objetos creadas que ofrece Revit. (elaboración propia).



**Figura 3.12.**  
Famílias de pilares estructurales de acero ya creadas que ofrece Revit. (elaboración propia)

### 3.2.4.6. PERMITE MÚLTIPLES FLUJOS DE INFORMACIÓN.

Los modelos, edificios en nuestro caso, realizados en BIM tienen múltiples flujos de información procedentes de diferentes agentes colaborativos que van a parar al mismo contenedor o proyecto. El desarrollo y el crecimiento del modelo va desde el nivel conceptual, hasta el de construcción y estructura, el cual avanza y se modifica a tiempo real, obteniéndose así un mejor seguimiento gracias a sus propiedades de interoperabilidad, interdisciplinar, globalización y bidireccionalidad de la información.



**Figura 3.13.**

*Todos los flujos de información llegan al mismo contenedor. (elaboración propia)*

### 3.2.4.7. PROMUEVE LA EFICIENCIA ENERGÉTICA

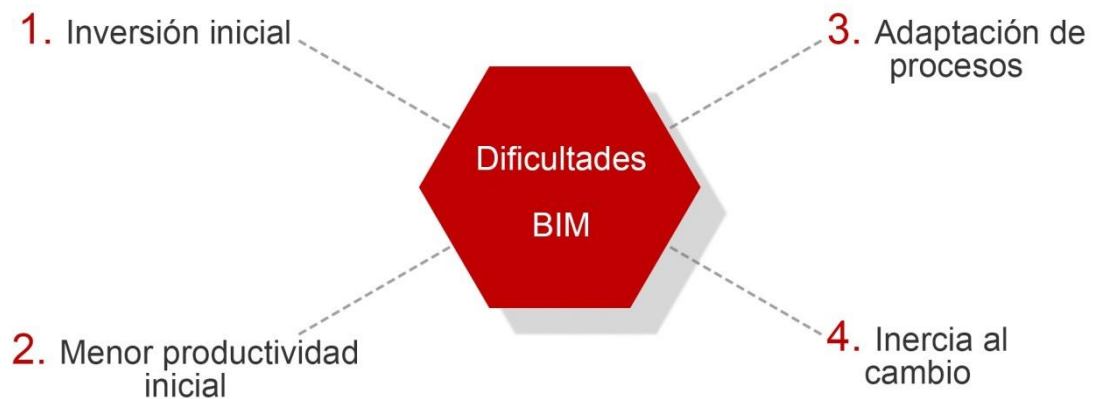
El BIM tiene la capacidad de añadir información suficiente para controlar la eficiencia energética y por lo tanto poder alcanzar el **confort ambiental**.

En el apartado *Análisis energético* de la *guía uBIM* se definen los requisitos para utilizar los BIM en el análisis energético durante las fases de diseño, construcción y uso. Ellos indican que para alcanzar el confort ambiental el BIM debería incluir:

- *Definición de recintos y su envolvente térmica.*
- *Consumo de la energía total.*
- *Certificado energético.*
- *Se recomienda: temperaturas de espacios durante el solsticio de invierno y de verano.*

### 3.2.5. DIFICULTADES DE LA METODOLOGÍA.

Todo cambio, cualquiera que sea su contexto, provoca beneficios e inevitablemente dificultades, en este caso son mucho menores que los beneficios, pero no por ello no deben nombrarse. Las dificultades de la metodología BIM son las siguientes:



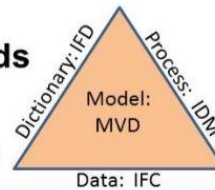
**Figura 3.14.**  
*Dificultades BIM. (elaboración propia)*

### 3.3. INDUSTRY FOUNDATION CLASS (IFC)

BuildingSMART creó el lenguaje digital que permite un intercambio abierto y libre de información entre los distintos actores resultando un eficiente y efectivo seguimiento del ciclo de vida del modelo proyectado.

#### Technical Principles: Basic Standards

There are five basic methodology standards



What it does	Name	Standard
Describes Processes	IDM Information Delivery Manual	ISO 29481-1 ISO 29481-2
Transports information / Data	IFC Industry Foundation Class	ISO 16739
Change Coordination	BCF BIM Collaboration Format	buildingSMART BCF
Mapping of Terms	IFD International Framework for Dictionaries	ISO 12006-3 buildingSMART Data Dictionary
Translates processes into technical requirements	MVD Model View Definitions	buildingSMART MVD

**Figura 3.15.**  
Estándares básicos. (<http://buildingSMART.org/standards/>)

BuildingSMART estableció 5 tipologías de estándares dependiendo del campo en el que se vayan a utilizar. En nuestro caso, de arquitectura y construcción haremos uso del IFC.

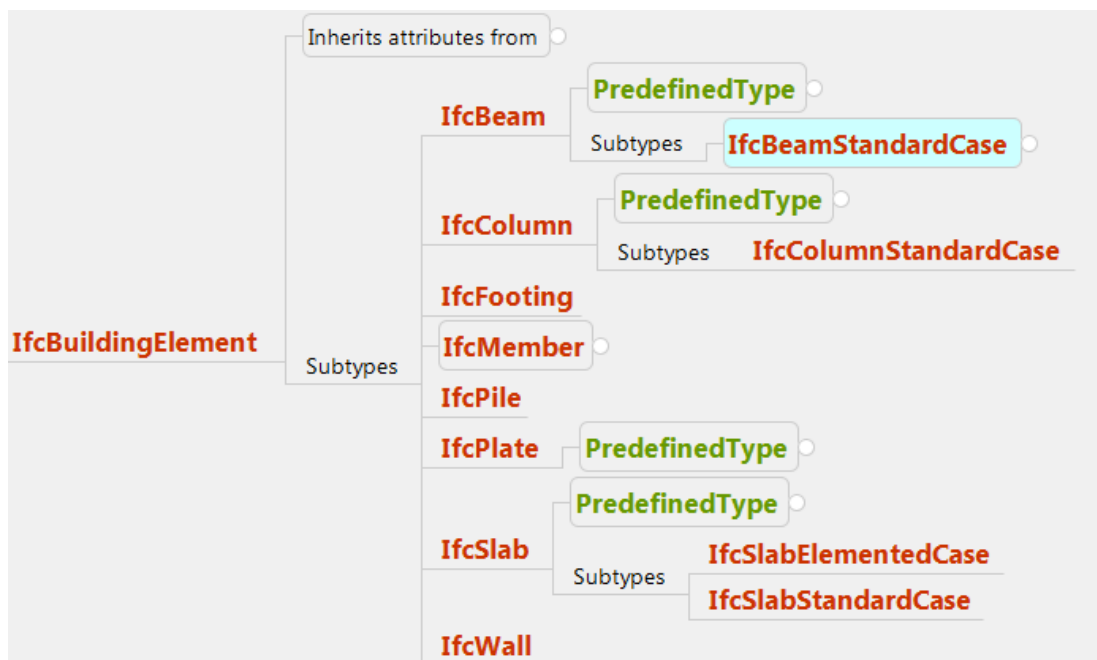
#### 3.3.1. DEFINICIÓN Y ORIGEN

*Industry Foundation Class* (IFC) es un **formato informático estándar, neutral y abierto** que permite con un único lenguaje la compartición e intercambio de datos, desarrollados por distintos fabricantes de software, de un edificio a lo largo de todo su ciclo de vida. IFC no es un lenguaje de programación ni pertenece a un único fabricante de software, es neutral e independiente, de manera que cada fabricante puede hacer compatible su software con IFC para que los datos e información

generada por su aplicación puedan ser compartidos con otros programas de otros fabricantes.

La norma y especificaciones de los modelos en IFC están disponibles con el lenguaje de programación informático de datos EXPRESS definido por la norma internacional **ISO 16739** (2013).

El IFC es un esquema de datos con **forma jerárquica**, en el cual se van desarrollando el conjunto de representaciones de datos significativos de la información del edificio, para facilitar el intercambio entre distintas aplicaciones de software en la industria AEC.



**Figura 3.16.**  
*Estructura jerárquica archivos IFC: funciones de elementos constructivos . (elaboración propia)*

Gracias a la capacidad de facilitar la interoperabilidad entre todas las plataformas software que lo permiten, muchos gobiernos están imponiendo el uso de archivos IFC para proyectos de la administración pública. En España, como anteriormente hemos comentado, ya hay unos objetivos fijados para los próximos años.

### 3.3.2. PRINCIPALES CARACTERÍSTICAS

El IFC se caracteriza por las siguientes cuestiones que a su vez resultan ventajosas:

- Permite **la interoperabilidad** entre **diferentes aplicaciones software**. El formato establece estándares internacionales para importar y exportar edificios y sus propiedades.
- Los formato BIM establece **estándares internacionales** para **importar y exportar** edificios y sus propiedades.
- Los archivos IFC **pesan un 20% menos** que los formatos de cualquier software nativo. **Aumentando la velocidad de gestión** y tratamiento de datos.
- Al ser el IFC una **tipología de archivo de texto** se puede abrir con cualquier programa de texto y **añadir y modificar sus parámetros**, a su vez los archivos de texto tienen la ventaja de que se pueden abrir con cualquier versión de los programas de texto ya que **nunca se queda “desfasado”** como pasaría con el formato de un software nativo. En cuanto a visualizarlo existen **visualizadores gratuitos** que permiten visualizar cualquier archivo IFC.
- Cada fabricante puede hacer **compatible su software con IFC** para que los datos e información generada por su aplicación puedan ser compartidos con otros programas.

En cuanto a la **desventaja** que ofrece este formato, es que al exportar un archivo desde un software nativo al formato IFC , este **podría perder algún tipo de información**, aunque con las actualizaciones que van haciéndose progresivamente en el BIM ésta desventaja cada vez será menor.



### 3.3.3. ORGANIZACIÓN DE LA INFORMACIÓN



El IFC está basado en **4 componentes principales** como bien se observa en su logo oficial. Este logo es un esquema de datos agregados a cada modelo participante en el proyecto de construcción y que tiene capacidad paramétrica. A continuación los explicaré individualmente (Kymmell2008):

**Figura 3.17.**

Logo IFC: reflejo de la interoperabilidad. (<http://buildingsmart.es>)

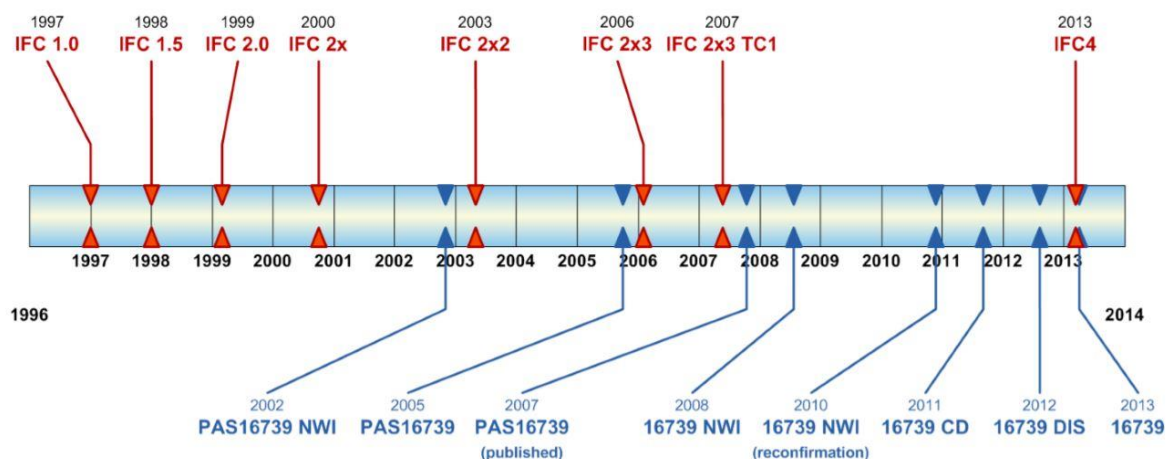
- 1. Design:** Hace referencia a la **geometría del modelo, dimensiones, área, volumen, ubicación espacial**, etc. Además, contendrá a su vez la relación del modelo con otros modelos del proyecto. Es la información básica que ha de contener cualquier elemento o modelo que pretendamos añadir como información de un proyecto constructivo.
- 2. Procure:** La **información se obtiene paraméricamente**. Es editable y contenida en el modelo de forma paramétrica, es decir, seleccionable de entre sus posibilidades. Es una información intrínseca al modelo, que lo define (el material que compone un muro: hormigón armado, ladrillo, bloque...), la calidad de dichos materiales (HA 25, ladrillo caravista...) Puede ser tanto información visual como de características definitorias del elemento (densidad, resistencia, mecánica...)
- 3. Operate:** El IFC **permite operar y funcionar con información externa**. Aquella información que no necesariamente va asociada o conectada con el modelo y que puede permanecer sin conectar o si fuera necesario enlazarla.
- 4. Assemble:** No contiene información del modelo en sí, sino de las **relaciones que éste establece con otros modelos del edificio** a través de enlaces visibles o no visibles.

### 3.3.4. CRONOLOGÍA DE VERSIONES.

La versión que apareció en primer lugar fue la IFC1, la cual se presentó en el año 1997, y desde entonces se fueron incorporando mejoras en el uso e intercambio de la información entre diferentes aplicaciones. En cuanto a la siguiente versión, IFC 2.0 se presentó en el 1999, siendo la versión IFC 2x del 2000 la cual supuso el establecimiento definitivo de una plataforma abierta que permitiera agregar nuevos elementos para intercambiar en el futuro. Seguidamente apareció la versión IFC2x Edition3, comúnmente denominado IFC2x3 publicada en el 2006, la cual constituye la versión más utilizada actualmente. Por último, la versión IFC4 apareció en el 2014 y promete tener ventajosas mejoras aunque aún no está del todo instaurado.

El modelo IFC 2x3 como ya he indicado anteriormente utiliza un tipo de lenguaje EXPRESS definido por la norma ISO 16739.

IFC1.0 / IFC 2.0	primeros prototipos	uso: 2000 – 2002
IFC2x / IFC 2x2	primeras adaptaciones	uso: 2002 – 2008
<b>IFC2X3</b>	<b>usado actualmente</b>	<b>uso: 2008 – 2016</b>
<b>IFC4</b>	<b>próximamente</b>	<b>desde el 2014</b>



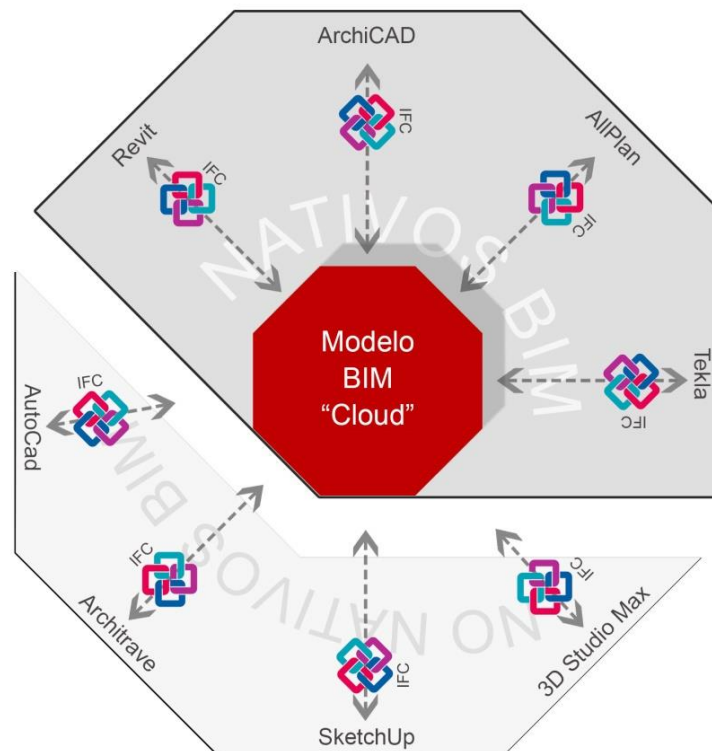
**Figura 3.18.**  
Cronología del desarrollo del IFC. (<http://buildingsmart.es>)

### 3.3.5. SOFTWARE COMPATIBLES

Actualmente, el BIM está en pleno crecimiento, de ahí que cada vez más empresas e instituciones estén desarrollando programas que corren bajo esta lógica o se adapten a ella. De esta manera, estarán actualizados en el ámbito de la construcción y arquitectura y podrán permitir la integración de los distintos actores dentro de un mismo flujo de trabajo tal y como establece la metodología BIM.

El arquitecto y Consultor BIM Héctor Miller clasificó los programas BIM como “no nativos BIM” y “Nativos BIM”: *“Las herramientas **no nativas** son aquellas que se desarrollaron con anterioridad al concepto y las denominadas como nativas son las que se crearon para dar respuesta al concepto”.*

Por otro lado, Miller establece que las herramientas “**nativas BIM**” pueden abarcar un gran abanico de disciplinas entre las que se destacan las vinculadas a la creación, el análisis, la comunicación y la gestión. Aunque nosotros nos centraremos en las vinculadas a la **creación** (**Revit**, **Archicad**, Allplan, Tekla y Dynamo), debido a que son las que se utilizan en el campo de la arquitectura.



**Figura 3.19.**  
Programas Nativos BIM y no Nativos BIM. (elaboración propia)

Las herramientas **nativas vinculadas a la creación** a diferencia de las *no nativas* tienen la posibilidad de trabajar en la nube o servicio “**Cloud**”, donde no solo se pueden almacenar archivos, sino que también se pueden correr parte de los procesos en servidores externos mediante una conexión a internet. De esta manera, se facilita y agiliza la comunicación y la fluidez entre las partes participantes. En cuanto a los **software nativos** más utilizados son:

- **Revit:** es un software BIM desarrollado por Autodesk, que aparte de poder **crear proyectos o modelos completos en 3D**, tiene la capacidad de **coordinación de datos automática y tareas necesarias para que la construcción sea eficiente y de alta calidad**. Puede además seguir al modelo a lo largo del ciclo de vida completo del edificio.
- **Archicad:** es un software BIM creado por Graphisoft, disponible para sistemas operativos Macintosh y Windows. Esta herramienta permite el **diseño paramétrico y contiene los datos de todo el ciclo de vida de la construcción** como en el caso de Revit, de hecho en cuanto a funcionalidad son muy similares. La diferencia más importante es que en este caso también **se puede utilizar con Macintosh** y en el caso de Revit no.
- **AllPlan:** es un software BIM para **Arquitectura, Ingeniería y Constructoras** para la **gestión de costes**. Éste es desarrollado por la empresa ALLPLAN del Grupo NEMETSCHek, el cual es disponible para el sistema operativo Windows, aunque nació bajo plataforma Unix. El software comenzó siendo una solución interna de un despacho de ingeniería de Múnich, Alemania, que posteriormente se fue mejorando y adaptando a lo que es hoy en día.
- **Tekla :** es un software **BIM** desarrollado por la empresa finlandesa TEKLA, disponible para sistema operativo Windows. **Permite el diseño en 3D detallado, del despiece, fabricación y montake de todo tipo de estructuras para la construcción.**



**Figura 3.20.**

*Logos aplicaciones Nativas BIM. (páginas oficiales de cada una de ellas).*

Por otro lado, se encuentran **los software no nativos**, es decir creadas antes de que apareciera la metodología BIM, pero que sin embargo están conectadas a ella, ya que se pueden convertir los archivos creados a partir de éstas a formato IFC para poder abrirlo con cualquiera de los programas nativos.

- **AutoCAD.**
- **3D Studio.**
- **Sketchup.**
- **Architrave**

Dentro de los software o programas *No Nativos BIM* se encontraría el programa “Architrave” (en el cual con posterioridad profundizaré) ,creado en nuestra universidad UPV, debido a que fue creado con anterioridad a los *nativos BIM*, pero que se podrá **conectar al flujo de los elementos BIM** mediante el protocolo de creación de elementos estructurales con objetos BIM que crearemos a continuación. Este protocolo nos permitirá **adaptarnos y convertir a un formato BIM**, es decir, IFC, **desde un programa No Nativo BIM, todo aquello que creemos y calculemos en las estructuras obtenidas desde Architrave**. De este modo actualizaríamos la aplicación permitiendo que ésta pueda permanecer en el **flujo de información único** proveniente de distintos actores, que establece el BIM en todo proyecto constructivo.



**Figura 3.21.**  
Logos aplicaciones No Nativas BIM. (páginas oficiales de cada una de ellas).



## 4.1. ARCHITRAVE Y SU RELACION CON EL BIM.

Architrave tiene su origen en mi universidad, la Politécnica de Valencia y es el resultado de trabajos de investigación de grupo CiD, del Departamento de Mecánica de los Medios Continuos y Teoría de Estructuras, y del área de Grid y Computación de Altas Prestaciones (GRyCAP) del Instituto de Instrumentación para Imagen Molecular. Sin embargo, personalmente mi acercamiento a este programa se dio en la asignatura de Estructuras1, con el cual hicimos un predimensionado de la estructura de acero de un edificio y lo calculamos y sacamos diversa información para entender el funcionamiento de las cargas y de la estructura del edificio a tratar. Me pareció un programa fácil e intuitivo en comparación con todo lo que puedes obtener de él a la hora de proyectar una estructura, diseñarla y realizar su predimensionado.

La definición oficial y la cual indica todas sus funciones y características se encuentra en la página oficial (<http://www.architrave.es/producto/producto.php>) y es la siguiente:

*“ **Architrave®**, compuesto por las aplicaciones **Architrave® Diseño** y **Architrave® Cálculo**, es un entorno informático orientado al diseño y al cálculo de estructuras de edificación y obra civil. **Architrave® Diseño** **permite**, mediante un sencillo e intuitivo interfaz gráfico de usuario, **generar y gestionar modelos estructurales utilizando el programa AutoCAD®**. Posteriormente, los **modelos son analizados y calculados por la aplicación Architrave® Cálculo**. Ambas aplicaciones son capaces de resolver el análisis estático y dinámico de estructuras en 3D mediante el método de los elementos finitos, así como el **dimensionado de los elementos estructurales de acero o de hormigón armado, gestionar su peritación y generar la documentación de proyecto –memorias de cálculo y planos– pertinente.**”*

### 4.1.1. ARCHITRAVE Y SUS CARACTERÍSTICAS

#### 1. Modelado de la Estructura.

Architrave permite la **generación de modelos estructurales desde el programa AutoCAD**. Tiene un panel de herramientas que ayuda ya que están realizados en su base de datos los prototipos de barras, elementos finitos en 2D, materiales, acciones, etc. más utilizados. Por otro lado, es muy versátil en cuanto a la tipología de estructuras ya que no hay restricciones de forma, tamaño o tipología estructural, pudiendo crear aquella que a cada uno mejor le convenga en su proyecto.

## 2. Cálculo de la estructura

*Architrave® Cálculo* permite la Computación de estructuras de Altas Prestaciones mediante un **análisis de la estructura lineal estático y dinámico por el método de los elementos finitos**. En el tenemos la posibilidad de combinar, en un mismo modelo, barras, elementos superficiales de membrana y de placa mediante triángulos y cuadriláteros, y elementos sólidos mediante tetraedros y hexaedros.

La gran ventaja de esta aplicación es su **cálculo altamente optimizado**, ya que aplica técnicas que pueden explotar eficientemente los procesadores de los ordenadores actuales. La combinación de dichas técnicas, permiten el poder calcular estructuras de grandes dimensiones en un tiempo reducido.

## 3. Visualización de Resultados

Una vez *Architrave® Cálculo* ha realizado el cálculo de la estructura, esta misma aplicación permite una **visualización de los mismos realista y eficiente**, con el fin de proporcionar un entorno avanzado para el análisis en 3D de estructuras de gran dimensión. Para un mejor resultado, también se permite **representar, visualizar y consultar resultados del análisis**, a la vez que permite la visualización realista de **esfuerzos, deformadas en las barras de la estructura y la distribución de tensiones calculadas mediante el método de elementos finitos**.

Por otro lado, en cuanto a la estructura *Architrave® Cálculo* permite mostrarla en 3D y seguidamente rotarla, trasladarla, modificar su tamaño de visualización y seleccionar elementos de la misma, todo ello dentro de los modos de visualización sólido y alámbrico, convirtiendo la tarea en una etapa mucho más rápida e intuitiva.

## 4. Dimensionado, Peritación y Planos de Ejecución

*Architrave® Cálculo* permite dimensionado de elementos de hormigón armado y de acero, teniendo en cuenta en todo momento las especificaciones del **Código Técnico de la Edificación**, de la **Instrucción EHE-08 de hormigón estructural** y de la **Normativa de Construcción Sismorresistente NCSE-02**. El dimensionado de las estructuras el programa lo realiza de forma automática a partir de las solicitaciones calculadas dependiendo de las cargas previamente asignadas a la ella. Por otro lado, con la utilidad de Peritación se pueden cambiar los resultados del dimensionado automático y es cuando el sistema comprueba su viabilidad. Por último, añadir que el formato con el cual genera los planos de ejecución este módulo es el DXF.



### 4.1.2. ARCHITRAVE, AutoLISP y BIM

Architrave como ya hemos comentado, se subdivide en dos aplicaciones; *Architrave® Diseño* y *Architrave® Cálculo*.

En concreto el Módulo de *Architrave® Diseño* que utiliza Architrave está insertado en el programa **AutoCAD®**. En su momento la elección de este entorno de Diseño Asistido por Ordenador se basó en los siguientes argumentos:

- Era uno de los sistemas CAD más conocidos y contaba con un gran número de usuarios.
- Perteneecía a la compañía que diseñó el formato de intercambio de ficheros de dibujos más extendido: el formato DXF.
- Permitía elaborar aplicaciones de usuario mediante los entornos de desarrollo que incorporaba; en especial el **lenguaje de programación AutoLISP** .

El Módulo de Diseño, como se acaba de explicar, se trata de una serie de funciones o rutinas programadas en **Visual LISP** que se cargan en el programa AutoCAD, a modo de barra de menús adicional con aspecto de árbol. Para realizar la modelización de una estructura, ésta se dibuja con entidades propias de AutoCAD, asignándole propiedades geométricas y mecánicas desde el árbol e insertando propiedades gráficas diversas que hacen referencia a las acciones y apoyos.

Una vez elaborado el modelo con *Architrave® Diseño*, los elementos gráficos que lo componen se exportan a un fichero en formato XML, que sirve de vínculo con *Architrave® Cálculo*, para poder proseguir con el análisis de la estructura. Este TFG tiene como **objetivo exportar la geometría de cualquier modelo estructural** desarrollado con *Architrave® Diseño* a cualquiera de los **entornos BIM** mencionados. Para ello se utiliza el formato IFC, con todas las ventajas que aporta.

Ya que que la base para utilizar Architrave es el programa AutoCAD y la base o lenguaje de programación para desarrollar aplicaciones en AutoCAD es AutoLISP, a la hora de implementar el protocolo de exportación que nos permita recrear los elementos estructurales utilizando objetos BIM utilizaremos el entorno de programación Visual LISP; para generar ficheros de texto codificados según las especificaciones IFC.

### 4.1.3. PROGRAMACIÓN EN Visual LISP

Visual LISP es la puesta al día de AutoLISP, esto es, el lenguaje de programación que ha permitido y lo sigue haciendo actualmente, a los usuarios de AutoCAD hacer de esta aplicación una herramienta adaptada a sus necesidades específicas. Visual LISP apareció por primera vez como una extensión de AutoCAD 14. Seguidamente se incorporó definitivamente en el interior de CAD.

El libro de Togores y Otero afirma que:

*AutoLISP es un lenguaje de programación diseñado para extender y personalizar la funcionalidad de AutoCAD. Se basa en el lenguaje de programación LISP, cuyos orígenes se remontan a fines de la década de los 50. Seguramente la elección, allá por 1985, de LISP como modelo del lenguaje de programación para AutoCAD pudo parecer por lo menos extraña. Se trataba de un lenguaje peculiar, de uso casi exclusivo en entornos tan elitistas como las investigaciones sobre inteligencia artificial. Pero según afirmaba entonces John Walker, uno de los creadores de AutoCAD,*

*“Ningún otro entre los principales lenguajes de programación es capaz de manipular tan fácilmente los tipos de objetos con que uno trabaja en CAD. Al contrario que la programación numérica, el CAD constantemente trabaja sobre colecciones de objetos heterogéneos en grupos de tamaño variable. LISP es excelente en ese aspecto”*

El entorno de desarrollo integrado Visual LISP (IDE VLISP), que se suministra con AutoCAD a partir de la versión 2000, incluye herramientas que ayudan en la redacción del código fuente, en su modificación y en la depuración de errores. Proporciona, además, el medio para empaquetar y distribuir aplicaciones en un archivo ejecutable único que incluye la interfaz de diálogo programada en DCL. Estas operaciones se realizan desde un entorno único, que permite tanto la edición del texto como la depuración del programa y la interacción con AutoCAD y otras aplicaciones.

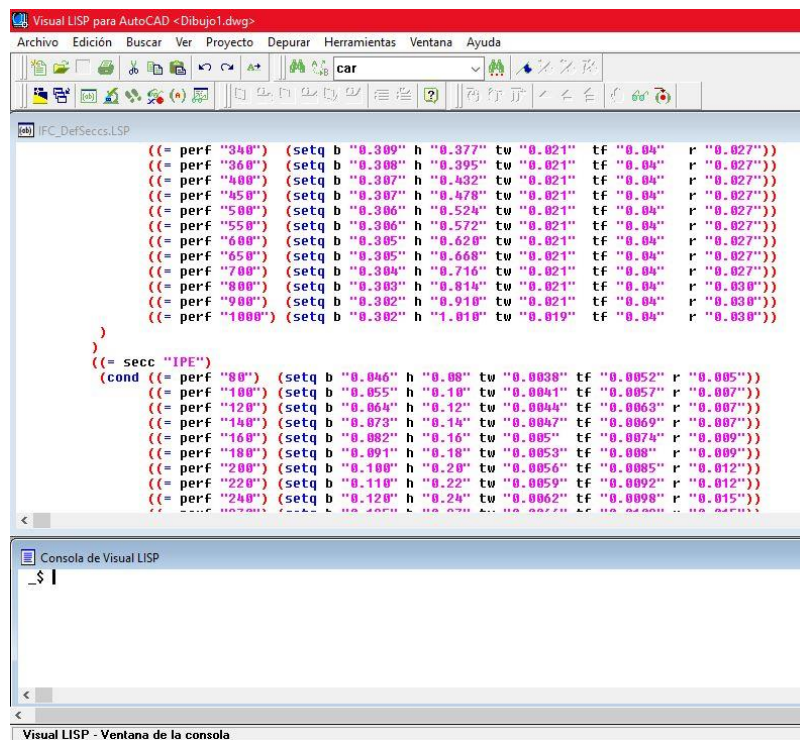
AutoLISP proporciona **tres formatos de archivo** para las aplicaciones:

- Lectura de un archivo **LSP** (.lsp), un archivo de texto ASCII que contiene el código de programa de AutoLISP.
- Lectura de un archivo **FAS** (.fas), versión binaria compilada de un único archivo de programa LSP.

- Lectura de un archivo **VLX** (.vlx), un conjunto compilado de uno o varios archivos LSP o DCL (Dialog Control Language, Lenguaje de control de diálogos. (Los archivos VLX y DCL no se admiten en AutoCAD para Mac).

Debido a que *el protocolo* será realizado mediante el programa Visual LISP considero importante hacer una pequeña **síntesis de su funcionamiento** general.

Una sesión de Visual LISP sólo puede iniciarse desde AutoCAD. Para iniciar la sesión se tecleará **vlide** o **vlisp** en la línea de comandos. Al activarse visual LISP se muestran los siguientes componentes:



**Figura 4.1.**  
Ventana con la consola de Visual LISP (elaboración propia).

### 4.1.3.1. LA CONSOLA Visual LISP

Las expresiones LISP se pueden ejecutar desde la línea de comandos de AutoCAD o desde la Consola. Aunque en esta ocasión lo haremos desde la Consola; donde ejecutaremos las expresiones LISP para comprobar los resultados y obtener de ahí el archivo de texto IFC como resultado. Como ayuda, en la ventana de la Consola van apareciendo mensajes en caso de producirse errores, así como los resultados de determinadas expresiones.

### Funcionamiento de la Consola:

- **Evalúa expresiones LISP :**  
El símbolo `_ $` indica que está en condiciones de recibir una expresión para evaluarla. El texto que se teclea en la Consola no es procesado hasta pulsar INTRO de manera expresa.
- **Repite y descarta expresiones LISP :**  
Para repetir la evaluación de una expresión ya utilizada en la Consola podemos valernos de la tecla `TAB`. El texto tecleado y los resultados de la ejecución de las expresiones se conservan en la ventana de la Consola. Sin embargo, en el caso de que al escribir una expresión en la Consola decidimos no continuar, pulsando la tecla `ESC` la eliminaremos y tampoco se guardará en el historial.
- **Menú contextual de la consola :**  
Las funciones `VLISP` más importantes para el trabajo en la Consola se incluyen en el menú contextual que aparece al pulsar el botón derecho del ratón en cualquier lugar de la consola.
- **Sintaxis coloreada :**  
Se resaltan con diversos colores las diferentes partes en las cuales se compone una expresión LISP. En cambio, una vez evaluada la expresión, desaparecen los colores.
- **La Consola y los dibujos AutoCAD :**  
`VLISP` cambia automáticamente de contexto al cambiar el dibujo activo, ya que en la Consola encontramos expresiones y comandos que han sido emitidos en el contexto de todos esos documentos. Por tanto, las expresiones que se teclearán en la Consola únicamente tendrán efecto en el *dibujo activo*.

### Evaluación de las expresiones.

- **Datos :**  
Entre las expresiones LISP más simples están los números. En este apartado es importante decidir si trabajaremos con enteros o con reales, ya que trabajando con enteros se truncarían los decimales. Los números enteros se representan en verde, en cambio si introducimos en la Consola

un número con decimales el color cambiaría a azul verdoso. Las letras, siempre dispuestas entre comillas, son de color magenta.

- **Expresiones :**

VLISP utiliza la notación “prefija”, ya que pone el operador al principio y aquello sobre lo que se va a operar seguidamente. Cuando LISP evalúa la llamada a una función, lo hace en dos pasos:

1. Evalúa los argumentos, de izquierda a derecha.
2. Los valores de los argumentos se pasan a la función representada por el operador.

- **Símbolos y asignación :**

Un símbolo representa *algo*; por ello el nombre de un símbolo LISP puede incluir números, pero no puede estar compuesto sólo por números, ya que los números ya se representan a sí mismos. Se deben incluir siempre letras o algunos otros caracteres no numéricos. Por el mismo motivo, un nombre de símbolo nunca incluirá comillas. Los símbolos que se colorean en azul tienen un significado ya asignado por el sistema. Los otros posibles símbolos no poseerán un valor hasta que el usuario se los asigne.

Cabe añadir, que LISP internamente convierte todos los nombres de símbolos en mayúsculas. Por esto es indiferente el uso de mayúsculas o minúsculas en ellos.

Algunos de los símbolos importantes en LISP y que aparecerán en el Protocolo creado son: **T**: positivo, con valor. **nil**: sin valor. 'o *quote*: función que devuelve su argumento sin evaluarlo. El carácter apóstrofe se coloreará en marrón para evitar confusiones.

- **Listas:**

Las listas son el tipo de dato más característico de LISP. Se expresan como cero o más elementos encerrados entre paréntesis. Los elementos de una lista pueden corresponder a cualquier tipo de dato, incluyendo otras listas. Es importante tener en cuenta que cuando empleamos una lista como dato, debemos precederla de *quote* para evitar que el sistema la confunda con una llamada a función , ya que éstas últimas toman la forma de listas.

#### 4.1.4. PROGRAMACIÓN PARA ARCHIVOS IFC.

En este apartado procederemos a la explicación general del procedimiento de generación del archivo IFC y del formato estándar de la creación de esta tipología de archivos.

Browsing documentation by:



##### *Alphabetical index*

- [Defined Types](#)
- [Enumerations](#)
- [Select Types](#)
- [Entities](#)

##### *Entities (653) :*

[Ifc2DCompositeCurve](#)  
[IfcActionRequest](#)  
[IfcActor](#)  
[IfcActorRole](#)  
[IfcActuatorType](#)  
[IfcAddress](#)  
[IfcAirTerminalBoxType](#)  
[IfcAirTerminalType](#)  
[IfcAirToAirHeatRecoveryType](#)  
[IfcAlarmType](#)  
[IfcAngularDimension](#)  
[IfcAnnotation](#)  
[IfcAnnotationCurveOccurrence](#)  
[IfcAnnotationFillArea](#)  
[IfcAnnotationFillAreaOccurrence](#)  
[IfcAnnotationOccurrence](#)  
[IfcAnnotationSurface](#)

En primer lugar explicaré el estándar general del archivo de texto IFC:

Todas las funciones IFC están precedidas del símbolo “#” seguida de un **número** el cual llamaremos “**IFCINDEX**”. Éste número hace referencia a la variable global que permite dar coherencia a la numeración de las etiquetas de todos los objetos que componen un modelo IFC. Se trata de un identificador único (para un modelo y una exportación concreta) que deja de tener utilidad una vez se ha realizado la importación y se ha incorporado el modelo a la base de datos de la aplicación informática que ha importado el fichero IFC.

Seguidamente **se indica la función** (toda ella escrita en mayúsculas), la cual siempre va precedida del prefijo “**IFC**” . Después de la función los parámetros de ésta se colocan entre paréntesis con el signo de coma entre ellos. Para terminar se coloca un espacio y el signo “;”.

Como apoyo para la realización de dichos archivos, he tomado la página oficial de BuildingSMART, la cual me ha permitido estudiar, analizar y aprender a utilizar y a clasificar adecuadamente las funciones IFC que podrían ser útiles para este TFG.

**Figura 4.2.**

Clasificación de las funciones en la página oficial.  
 (<http://www.buildingsmart-tech.org/ifc/IFC2x3/TC1/html/>)

Los pasos para la creación de este archivo son:

- 1.** Construcción de la **cabecera del fichero IFC** incluyendo información del proyecto, de la localización, del edificio, del autor...de acuerdo con el estándar IFC.
- 2.** **Apertura del fichero IFC** y escritura de la **cabecera**.
- 3.** **Selección de las entidades** de Architrave que se van a exportar al fichero IFC.
- 4.** **Clasificación de las entidades** en: barras (vigas, pilares, zunchos, correas, barras de celosías, ...) y mallas de elementos finitos (láminas, placas, muros y losas cada una con sus correspondientes huecos y zonas de refuerzo).
- 5.** **Detección de los tipos de secciones utilizadas** en el conjunto de todas las barras, creación de las entidades IFC que caracterizan cada una de las secciones y escritura en el fichero IFC.
- 6.** **Generación de las entidades IFC** que compondrán los diferentes niveles de detalle (LOD) de las barras y de las mallas de elementos finitos.
  - a.** **Barras.** Incluyen una representación alámbrica que caracteriza el eje de la barra y una representación sólida que describe su volumetría.
  - b.** **Mallas de elementos finitos.** Incluyen una representación alámbrica del contorno de la superficie mallada (losa, muro, ... ), de sus huecos y de sus zonas regruessadas así como una representación sólida que muestra su volumetría.
- 7.** **Ubicación y orientación de las entidades IFC** dentro del proyecto
- 8.** **Escritura sistemática**, dentro del fichero de exportación IFC, de la **descripción de cada una de las entidades** en ese formato.

Más adelante, se trata de explica con detalle la segunda parte del punto 6, esto es, el protocolo de generación de elementos BIM; el cual se trata de la creación de las entidades IFC que caracterizan cada una de las **barras**, sus secciones y su escritura en el fichero IFC.

## 4.2. DISEÑO DEL PROTOCOLO DE GENERACIÓN DE ELEMENTOS ESTRUCTURALES UTILIZANDO OBJETOS BIM.

La **generación** de un **archivo tipo IFC a partir de** un archivo de tipo **DWG** generado dentro de AutoCAD® con la aplicación *Architrave Diseño*® (ambas aplicaciones no nativas ni conectables BIM) **requiere conocer**, con más profundidad de lo habitual, **ambos entornos de desarrollo**. Sin embargo, una vez desarrollado el protocolo de exportación. Este cambio a IFC **obtendría múltiples ventajas** para el proyecto que deseemos calcular con Architrave, ya que de este modo se permitiría **la interoperabilidad y la compartición de datos del proyecto con los demás actores intervinientes en la obra**.

Con objeto de **ilustrar** el proceso de implementación del protocolo de exportación **se describen las funciones LISP** apropiadas para crear y concatenar los textos que componen cada una de las **líneas de un fichero IFC**.



## 4.2.2. FUNCIONES LISP UTILIZADAS.

En primer lugar, **antes de la creación del protocolo**, definiremos las funciones LISP utilizadas para ello, agrupándolas en diferentes tipologías según su función. Para la definición de éstas tomaré como ayuda el libro de *Programación en AutoCAD con Visual LISP*.

### Repetición de argumentos:

En muchos casos se repiten argumentos de la misma naturaleza. Con el propósito de evitar la repetición de explicaciones, a continuación se explican los nombres utilizados para esos argumentos, lo que significan y el tipo de dato requerido.

“**ANG**”: determina el ángulo que existe entre dos líneas. Los ángulos se miden en sentido contrario a las agujas del reloj respecto al eje x en 2D, o respecto a un eje especificado por el usuario en 3D. Se expresa en radianes (como número real).

“**CAD**”: argumento del tipo de cadena de caracteres.

### Funciones aritméticas:

Funciones AutoLISP aritméticas utilizadas en la creación del protocolo. Todas ellas tienen un límite de 10 parámetros, entre los que se incluye el nombre de la función.

<b>Función</b>	<b>+</b>
<b>Descripción</b>	Función que devuelve la suma de todos los números.
<i>(+ number number...)</i>	
<b>Parámetros</b>	number      Entero o Real      Números a sumar

<b>Función</b>	<b>*</b>
<b>Descripción</b>	Función que devuelve el producto de todos los números.
<i>(* number number...)</i>	
<b>Parámetros</b>	number      Entero o Real      Números a multiplicar

<b>Función</b>	<b>1+</b>
<b>Descripción</b>	Función que devuelve el argumento incrementado en uno.
	<i>(1+ number)</i>
<b>Parámetros</b>	number Entero o Real Número a incrementar

<b>Función</b>	<b>COS</b>
<b>Descripción</b>	Función que devuelve el coseno de un ángulo como número real expresado en radianes.
	<i>(cos ang)</i>
<b>Parámetros</b>	ang Entero o Real Ángulo en radianes

<b>Función</b>	<b>SIN</b>
<b>Descripción</b>	Función que devuelve el seno de un ángulo como número real expresado en radianes.
	<i>(sin ang)</i>
<b>Parámetros</b>	ang Entero o Real Ángulo en radianes

**Funciones de igualdad, condicionales y ciclos:**

<b>Función</b>	<b>COND</b>
<b>Descripción</b>	Función que evalúa una serie de condiciones de prueba secuencialmente. Sirve como la función primaria condicional para AutoLISP. Se utiliza para codificar la bifurcación multirrama.
	<i>(cond test expr)</i>
<b>Parámetros</b>	test Lista Condición de prueba a ser evaluada result Lista Los argumentos que se ejecutan si la condición de prueba es satisfactoria.
<b>Resultado</b>	T o nil El valor de la última expresión en la lista secundaria . Si sólo hay una expresión en la lista secundaria (es decir, si el resultado no se encuentra ), se devuelve el valor de la expresión de prueba . Si no se suministran argumentos , <i>cond</i> nos resultará como nil.

<b>Función</b>	<b>=</b>		
<b>Descripción</b>	Función que devuelve <i>T</i> si todos los argumentos son numéricamente iguales, y <i>nil</i> en caso contrario.		
<i>(= number number)</i>			
<b>Parámetros</b>	number	Entero o Real	Número a incrementar

### Funciones para definición y administración de funciones:

<b>Función</b>	<b>DEFUN</b>		
<b>Descripción</b>	Es una función la cual su primer argumento es el nombre que se le quiere dar a la función . A continuación, aparece la lista de parámetros y por último el cuerpo de la función .		
<i>(defun sym argumento/ variables...expr)</i>			
<b>Parámetros</b>	sym	Símbolo	Valor numérico
	argumentos	Entero, Real, String, Lista, T o nil.	Los nombres de los argumentos esperados por la función.
	/variables	Símbolo	Los nombres de una o más variables locales de la función. La barra que precede los nombres de las variables debe ser separada de la primera nombre local y desde el último argumento, en su caso, por lo menos un espacio.
	expr	Lista	Cualquier número de expresiones de AutoLISP que se evalúa cuando la función se ejecuta.
<b>Resultado</b>		Entero, Real, String, Lista, T o nil.	Resultado de la última expresión evaluada.

### Funciones de manipulación de listas:

<b>Función</b>	<b>NTH</b>		
<b>Descripción</b>	Función que devuelve el elemento enésimo de la lista. Empieza a contar a partir de cero.		
<i>(nth n lst)</i>			
<b>Parámetros</b>	n	Entero	Número que indica el orden del elemento de la lista que se pretende mostrar (cero es el primer número).
	lst	Lista	Lista formada por uno o varios elementos
<b>Resultado</b>		Entero, Real, String, Archivo, Nombre, T, o nil	El enésimo elemento de LST . Si n es mayor que el número de elemento más alto de lst , dará como resultado <i>nil</i> .

**Operaciones con cadenas:**

<b>Función</b>	<b>STRCAT</b>		
<b>Descripción</b>	Función que devuelve una cadena que es la concatenación de múltiples cadenas de caracteres.		
<i>(strcat cadena1 cadena2...)</i>			
<b>Parámetros</b>	cadena	String	Valores de texto para concatenar
<b>Resultado</b>		String	Concatenación de cadenas de texto . Si no se suministran argumentos , <i>strcat</i> devuelve una cadena de longitud cero .

<b>Función</b>	<b>STD-STRSPLIT</b>		
<b>Descripción</b>	Función que descompone una cadena de caracteres utilizando como delimitador un cierto carácter.		
<i>(std-strsplit cadena cadena2...)</i>			
<b>Parámetros</b>	cadena	String	Valores de texto para descomponer
<b>Resultado</b>		String	Concatenación de cadenas de texto . Si no se suministran argumentos , <i>strcat</i> devuelve una cadena de longitud cero .

**Funciones de Administración de Símbolos:**

<b>Función</b>	<b>SETQ</b>		
<b>Descripción</b>	Función utilizada para asignar un determinado valor a un símbolo.		
<i>(setq sym1 expr1 sym2 expr2...)</i>			
<b>Parámetros</b>	sym	símbolo	Variable definida por el usuario para asignar a <i>expr</i> . Este argumento no se evalúa.
	expr	entero/real/string/ lista/archivo/ nombre/T/nil	Expresión genérica
<b>Resultado</b>		Entero, Real, String, Lista, Archivo, Nombre, T, o nil	Resultado de la última <i>expr</i> evaluada.

## Funciones de conversión de datos:

<b>Función</b>	<b>ATOF</b>		
<b>Descripción</b>	Función que devuelve la conversión de una cadena en un número real.		
<i>(atof str)</i>			
<b>Parámetros</b>	str	String	Cadena a convertir en nº real
<b>Resultado</b>		Real	Valor numérico

<b>Función</b>	<b>atoi</b>		
<b>Descripción</b>	Función que devuelve la conversión de una cadena en un número entero.		
<i>(atoi str)</i>			
<b>Parámetros</b>	str	String	Cadena a convertir en nº entero
<b>Resultado</b>		Entero	Valor numérico

<b>Función</b>	<b>itoa</b>		
<b>Descripción</b>	Función que devuelve la conversión de un entero en una cadena.		
<i>(itoa int)</i>			
<b>Parámetros</b>	int	Entero	Número entero a convertir
<b>Resultado</b>		String	Cadena obtenida

<b>Función</b>	<b>rtos</b>		
<b>Descripción</b>	Función que permite controlar el formato y la cantidad de decimales: devuelve la conversión de un número en una cadena, permitiendo controlar el número de decimales visibles, ya que si no fuera de esta forma la línea de comandos de AutoCAD truncaría los números reales a cinco decimales máximo.		
<i>(RTOS value mode precision)</i>			
<b>Parámetros</b>	value	Entero o Real	Valor numérico
			El modo corresponde a los valores permitidos para la variable del sistema de AutoCAD LUNITS. El modo puede ser uno de los siguientes números:
	mode	Entero	1 - Científico. 2 - Decimal. 3 - Ingeniería (pies y pulgadas decimales) 4 - Arquitectura (pies y fracciones de pulgada) 5 - Fraccional
	precision	Entero	Valor numérico
<b>Resultado</b>		String	Valor numérico formateado

**Funciones de conversión grados-radianes:**

Puesto que AutoLISP trabaja con radianes mientras que AutoCAD lo hace por defecto en ángulos sexagesimales, es necesario tener a mano siempre las funciones de conversión grados-radianes.

<b>Función</b>	<b>RAD</b>		
<b>Descripción</b>	recibe ángulos en radianes y los devuelve en grados.		
<i>(rad número)</i>			
<b>Parámetros</b>	número	Entero o Real	Número en radianes
<b>Resultado</b>		Entero o Real	Número en grados

<b>Función</b>	<b>GRD</b>		
<b>Descripción</b>	Función que recibe ángulos en grados y los devuelve en radianes.		
<i>(grd número)</i>			
<b>Parámetros</b>	número	Entero o Real	Número en grados
<b>Resultado</b>		Entero o Real	Número en radianes

**Función de manejo de archivos**

<b>Función</b>	<b>WRITE-LINE</b>		
<b>Descripción</b>	función que se emplea para escribir una línea o cadena en la línea de comandos o en un archivo de texto abierto.		
<i>(write-line cadena id-archivo)</i>			
<b>Parámetros</b>	cadena	String	Valor textual
	id-archivo	Archivo	Direccionado a un archivo abierto

### 4.2.1. FUNCIONES IFC UTILIZADAS

A continuación explicaremos algunas de las funciones de la librería de *Architrave* de las que nos hemos servido para poder generar posteriormente **ficheros IFC de la geometría de un modelo estructural genérico**.

<b>Función</b>	<b>-&gt;IFC</b>		
<b>Descripción</b>	Define un sólido según el estándar IFC a partir de la extrusión de una superficie (abierta o alveolar) generada también según el estándar IFC		
	<i>(-&gt;IFC camino fichero modoSeleccion)</i>		
<b>Parámetros</b>	camino	String	Ubicación donde se guardará el fichero IFC
	fichero	String	Nombre del fichero donde se creará el modelo en IFC
	modoSeleccion	String	Procedimiento para seleccionar (por ventana o por captura) las entidades que formarán parte del fichero IFC
<b>Resultado</b>			

<b>Función</b>	<b>getIfcGuid</b>		
<b>Descripción</b>	Genera un identificador de entidades único tipo GUID		
	<i>(getIfcGuid)</i>		
<b>Parámetros</b>			
<b>Resultado</b>	String	Identificador GUID de 22 caracteres	

<b>Función</b>	<b>b_datos-&gt;IFC</b>		
<b>Descripción</b>	Incorpora el bloque datos genéricos al fichero IFC		
	<i>(b_datos-&gt;IFC fich)</i>		
<b>Parámetros</b>	fich	Descriptor	Fichero IFC
<b>Resultado</b>			

<b>Función</b>	<b>IfcCartesianPoint</b>		
<b>Descripción</b>	Define un punto por sus coordenadas cartesianas según el estándar IFC		
	<i>(IfcCartesianPoint fich c1 c2 c3)</i>		
<b>Parámetros</b>	fich	Descriptor	Fichero IFC
	c1	Real	Coordenada X del punto
	c2	Real	Coordenada Y del punto
	c3	Real	Coordenada Z del punto
<b>Resultado</b>	String	Etiqueta del punto dentro del fichero IFC	

<b>Función</b>	<b>IfcDirection</b>		
<b>Descripción</b>	Define una dirección mediante un vector unitario según el estándar IFC		
<i>(IfcDirection fich c1 c2 c3)</i>			
<b>Parámetros</b>	fich	Descriptor	Fichero IFC
	c1	Real	Coseno director del vector unitario respecto eje X
	c2	Real	Coseno director del vector unitario respecto eje Y
	c3	Real	Coseno director del vector unitario respecto eje Z
<b>Resultado</b>	String	Etiqueta de la dirección dentro del fichero IFC	

<b>Función</b>	<b>IfcPolyline</b>		
<b>Descripción</b>	Define una polilínea según el estándar IFC		
<i>(IfcPolyline fich lpt)</i>			
<b>Parámetros</b>	fich	Descriptor	Fichero IFC
	lpt	Lista	Lista de listas con las coordenadas de los vértices de la polilínea
<b>Resultado</b>	String	Etiqueta de la polilínea dentro del fichero IFC	

### **Definición de funciones para la creación de BARRAS estructurales.**

<b>Función</b>	<b>IfcIShapeProfileDef</b>		
<b>Descripción</b>	Define un perfil de sección I según el estándar IFC		
<i>(IfcIShapeProfileDef fich secc perf giro / b h tw tf r res orig ang dir lst)</i>			
<b>Parámetros</b>	fich	Descriptor	Fichero IFC
	secc	Real	Tipo de perfil
	perf	Real	Designación del perfil
	giro	String	Ángulo de giro del perfil una vez generado
	b	Real	Longitud del perfil en dirección x
	h	Real	Longitud del perfil en dirección y
	tw	Real	Espesor del alma
	tf	Real	Espesor del ala
	r	Real	Radio de acuerdo ala-alma
	<b>Resultado</b>	String	Etiqueta de la dirección dentro del fichero IFC



<b>Función</b>	<b>IfcRectangleProfileDef</b>		
<b>Descripción</b>	Define un perfil de sección rectangular según el estándar IFC		
<i>(IfcRectangleProfileDef fich dim giro / dimens x y res orig ang dir lst)</i>			
<b>Parámetros</b>	fich	Descriptor	Fichero IFC
	dim	Real	Dimensiones de la sección rectangular
	giro	String	Ángulo de giro del perfil una vez generado
	x	Real	Longitud de la sección en la dirección x
	y	Real	Longitud de la sección en la dirección y
<b>Resultado</b>	String	Etiqueta de la dirección dentro del fichero IFC	

<b>Función</b>	<b>IfcUShapeProfileDef</b>		
<b>Descripción</b>	Define un perfil de sección en U según el estándar IFC		
<i>(IfcUShapeProfileDef fich secc perf / h b tw tf r1 r2 res orig ang dir lst)</i>			
<b>Parámetros</b>	fich	Descriptor	Fichero IFC
	secc	Real	Tipo de perfil
	perf	Real	Designación del perfil
	h	Real	Longitud del perfil en dirección y
	b	Real	Longitud del perfil en dirección x
	tw	Real	Espesor del alma
	tf	Real	Espesor del ala
	r1	Real	Radio de acuerdo ala-ala
	r2	Real	Radio exterior del ala
<b>Resultado</b>	String	Etiqueta de la dirección dentro del fichero IFC	

<b>Función</b>	<b>IfcCircleProfileDef</b>		
<b>Descripción</b>	Define un perfil de sección circular según el estándar IFC		
<i>(IfcCircleProfileDef fich diam / rad res)</i>			
<b>Parámetros</b>	fich	Descriptor	Fichero IFC
	diam	Real	Dimensiones de la sección circular
	rad	Real	Radio de la sección circular
<b>Resultado</b>	String	Etiqueta de la dirección dentro del fichero IFC	

<b>Función</b>	<b>IfcRectangleHollowProfileDef</b>		
<b>Descripción</b>	Define un perfil de sección rectangular hueca según el estándar IFC		
	<i>(IfcRectangleHollowProfileDef fich secc perf giro / b h e r2 r1 res orig ang dir lst)</i>		
<b>Parámetros</b>	fich	Descriptor	Fichero IFC
	secc	Real	Tipo de perfil
	perf	Real	Designación del perfil
	giro	String	Ángulo de giro del perfil una vez generado
	b	Real	Longitud del perfil en dirección x
	h	Real	Longitud del perfil en dirección y
	tw	Real	Espesor perfil
	r2	Real	Radio interno del perfil
	r1	Real	Radio externo del perfil
<b>Resultado</b>	String	Etiqueta de la dirección dentro del fichero IFC	

<b>Función</b>	<b>IfcCircleHollowProfileDef</b>		
<b>Descripción</b>	Define un perfil de sección circular según el estándar IFC		
	<i>(IfcCircleHollowProfileDef fich perf / dimens rad th res)</i>		
<b>Parámetros</b>	fich	Descriptor	Fichero IFC
	perf	Real	Designación del perfil
	dimens	¿	¿
	rad	Real	Radio de la sección circular hueca
	th	Real	Espesor perfil
<b>Resultado</b>	String	Etiqueta de la dirección dentro del fichero IFC	

### Perfil de usuario

<b>Función</b>	<b>contPerfilUsuario</b>		
<b>Descripción</b>	Genera una lista con los vértices del contorno de una sección de usuario según se define en los ficheros de bases de datos de Architrave		
	<i>(contPerfilUsuario dim closed)</i>		
<b>Parámetros</b>	dim	String	Descripción de las coordenadas de los vértices de la sección definida por el usuario de Architrave
	closed	Boolean	Indicador de contorno cerrado o abierto
<b>Resultado</b>		Lista	Lista de listas con las coordenadas de los vértices del contorno exterior de la sección de usuario y los vértices de los contornos de los huecos si existen

### ***Perfil a partir de una polilínea***

<b>Función</b>	<b>IfcArbitraryClosedProfileDef</b>		
<b>Descripción</b>	Define el contorno de un perfil a partir de una IfcPolyline según el estándar IFC		
<i>(IfcArbitraryClosedProfileDef fich lpt giro)</i>			
<b>Parámetros</b>	fich	Descriptor	Fichero IFC
	lpt	Lista	Lista de listas con las coordenadas de los vértices del contorno del perfil
	giro	String	Angulo de giro del perfil una vez generado
<b>Resultado</b>		String	Etiqueta del perfil dentro del fichero IFC

### ***Perfil alveolar a partir de una polilínea***

<b>Función</b>	<b>IfcArbitraryProfileDefWithVoids</b>		
<b>Descripción</b>	Define el contorno de un perfil alveolar a partir de una IfcPolyline de contorno y una o más IfcPolyline de huecos según el estándar IFC		
<i>(IfcArbitraryProfileDefWithVoids fich lpt lsth giro)</i>			
<b>Parámetros</b>	fich	Descriptor	Fichero IFC
	lpt	Lista	Lista de listas con las coordenadas de los vértices del contorno del perfil
	lsth	Lista	Lista de listas de listas con las coordenadas de los vértices del contorno de los huecos del perfil
	giro	String	Angulo de giro del perfil una vez generado
<b>Resultado</b>		String	Etiqueta del perfil dentro del fichero IFC

**Sólido a partir de la extrusión de una superficie**

<b>Función</b>	<b>IfcExtrudedAreaSolid</b>		
<b>Descripción</b>	Define un sólido según el estándar IFC a partir de la extrusión de una superficie (abierta o alveolar) generada también según el estándar IFC		
<i>(IfcExtrudedAreaSolid fich profile sysref dir thick)</i>			
<b>Parámetros</b>	fich	Descriptor	Fichero IFC
	profile	String	Etiqueta del perfil definida dentro del fichero IFC como IfcArbitraryProfileDefWithVoids, IfcArbitraryClosedProfileDef, IfcRectangleProfileDef, IfcLShapeProfileDef, ...
	sysref	String	Etiqueta del sistema de referencia definida dentro del fichero IFC como IfcAxis2Placement3D y según el cual se generará el perfil que se va a extruir. Si no indica se toma el sistema de referencia global
	dir	String	Etiqueta de la dirección definida como IfcDirection a lo largo de la que se extruirá el perfil
	thick	Real	Longitud del sólido extruido
<b>Resultado</b>		String	Etiqueta del sólido dentro del fichero IFC

**Barra**

<b>Función</b>	<b>barra-&gt;IFC</b>		
<b>Descripción</b>	Incorpora una barra al fichero IFC		
<i>(barra-&gt;IFC fich cap lab ibar tl puntos)</i>			
<b>Parámetros</b>	fich	Descriptor	Fichero IFC
	cap	String	Capa de la barra
	lab	String	Etiqueta de la barra
	ibar	Entero	Índice de la barra en el listado de barras
	tl	String	Nombre del tipo de línea de la barra. Se utiliza para localizar el tipo de sección dentro de la lista de secciones del modelo y generar el sólido apropiado según el estándar IFC
	puntos	Lista	Punto inicial y final como listas de coordenadas
<b>Resultado</b>		String	Etiqueta de la barra dentro del fichero IFC

## Contorno 2D

<b>Función</b>	<b>contornoEF-&gt;IFC</b>		
<b>Descripción</b>	Incorpora un contorno de malla de EF2D al fichero IFC como IfcArbitraryClosedProfileDef o como IfcArbitraryProfileDefWithVoids		
<i>(contornoEF-&gt;IFC fich cap lab icon t p0 v01 n puntos puntosLoc tl lstHuecos lstHuecosLoc shift)</i>			
<b>Parámetros</b>	fich	Descriptor	Fichero IFC
	cap	String	Capa de la barra
	lab	String	Etiqueta de la polilínea de contorno de mallado
	icont	Entero	Índice del contorno en el listado de contornos
	p0	Lista	Coordenadas del primer punto del contorno exterior
	v01	Lista	Vector en el plano del contorno
	n	Lista	Vector perpendicular al plano del contorno
	puntos	Lista	Puntos del contorno en coordenadas universales
	puntosLoc	Lista	Puntos del contorno en coordenadas locales
	tl	String	Nombre del tipo de línea de la barra
	lstHuecos	Lista	Lista de listas de puntos para cada hueco. Incluye los contornos de los reguesados para rellenarlos posteriormente
	lstHuecosLoc	Lista	Lista de listas de puntos para cada hueco en coordenadas locales. Incluye los contornos de los reguesados para rellenarlos posteriormente
	shift	Real	Desplazamiento del sólido para ajustar a caras
<b>Resultado</b>		String	Etiqueta del contorno dentro del fichero IFC

## Área de reparto de cargas

<b>Función</b>	<b>placa-&gt;IFC</b>		
<b>Descripción</b>	Incorpora una 3DCara (EF2D o Area de Reparto de cargas) al fichero IFC como IfcArbitraryClosedProfileDef		
<i>(placa-&gt;IFC fich cap lab ifor puntos puntosLoc depth)</i>			
<b>Parámetros</b>	fich	Descriptor	Fichero IFC
	cap	String	Capa de la 3DCara
	lab	String	Etiqueta de la 3DCara
	ifor	Entero	Índice de la 3DCara en el listado de 3DCaras
	puntos	Lista	Vértices de la 3DCara en coordenadas universales
	puntosLoc	Lista	Vértices de la 3DCara en coordenadas locales
	depth	Real	Espesor del EF o del Area de Reparto
<b>Resultado</b>		String	Etiqueta del EF o el Area de Reparto dentro del fichero IFC

### 4.2.3. EXPLICACIÓN POR FASES DEL PROTOCOLO.

Dado que el código de *Architrave Diseño* no es un código abierto, el equipo de desarrollo del programa ha incorporado funciones básicas de comunicación que nos han permitido crear y gestionar la base de datos de secciones así como implementar las funciones que permiten describir las barras y los mallados de elementos finitos que constituyen las losas y muros.

En este apartado se explica una parte del proceso de diseño del protocolo de generación de elementos estructurales. En concreto, **todas las tipologías de barras** mediante el programa Visual LISP, de tal modo que se entienda la metodología del proceso que nos permita obtener el resultado que buscamos: convertir el modelo estructural de *Architrave Diseño* al formato intercambiable IFC.

Los códigos de color utilizados serán los siguientes:

#### Exento al protocolo:

**Granate:** Explicaciones exentas al protocolo que sirven únicamente para comprender mejor aquello que vayamos realizando.

#### Forma parte del protocolo:

**Magenta:** Expresiones genéricas asignadas por símbolos, aquello que está en este color sí que se evalúa.

**Negro:** Símbolos a los que se les asigna una expresión, aquello que está en este color no se evalúa.

**Rojo:** Paréntesis utilizados para las diversas funciones.

**Azul:** Funciones LISP.

### IFCIShapeProfileDef

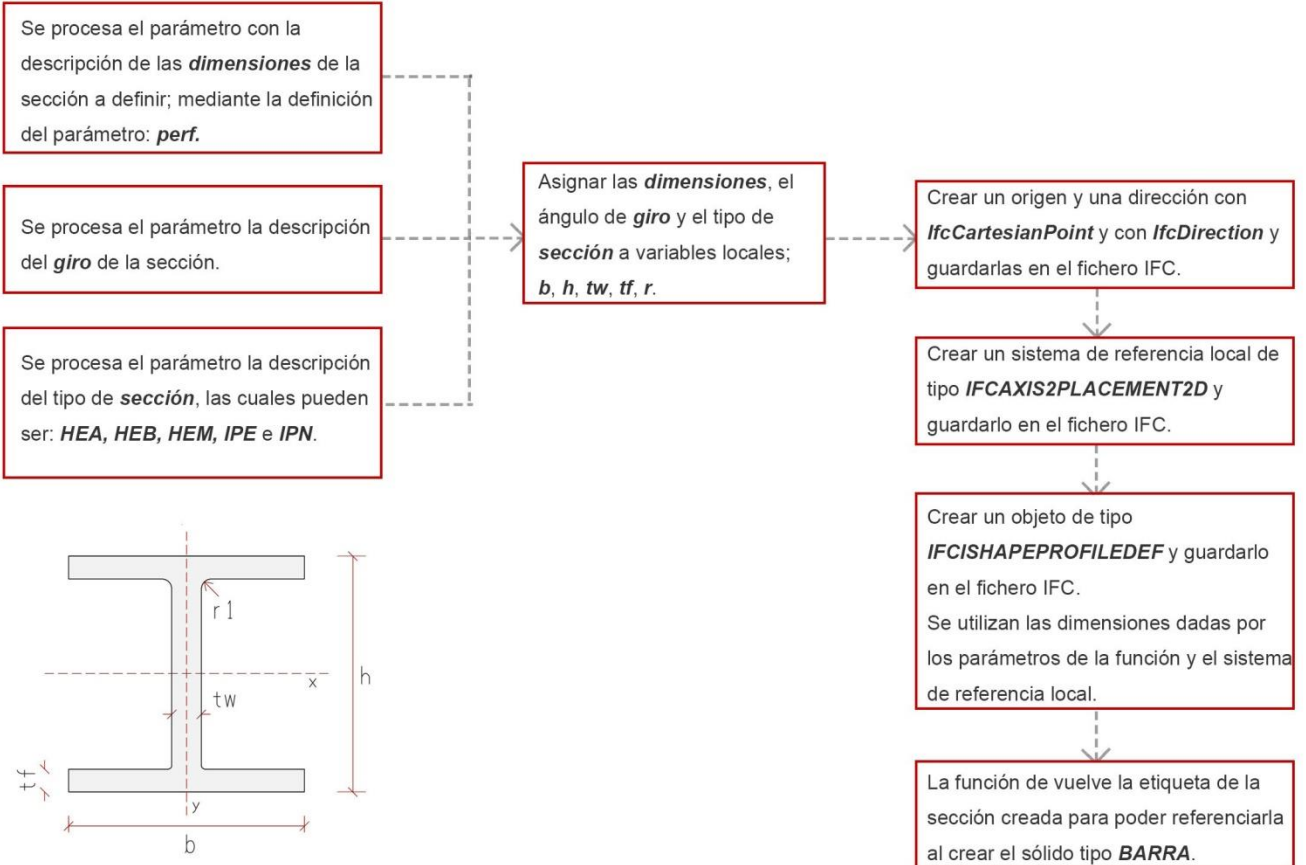


Figura 4.3.

Esquema explicativo de funcionamiento de la función IfcIShapeProfileDef. (elaboración propia)

```

(defun IfcIShapeProfileDef (fich secc perf giro / b h tw tf r res orig ang dir lst)
  Genera un IfcIShapeProfileDef a partir de las dimensiones de una sección en I.
  (cond ((= secc "HEA")
    (cond ((= perf "100") (setq b "0.10" h "0.096" tw "0.005" tf "0.008" r "0.012"))
      ((= perf "120") (setq b "0.12" h "0.114" tw "0.005" tf "0.008" r "0.012"))
      ((= perf "140") (setq b "0.14" h "0.133" tw "0.0055" tf "0.0085" r "0.012"))
      ((= perf "160") (setq b "0.16" h "0.152" tw "0.006" tf "0.009" r "0.015"))
      ((= perf "180") (setq b "0.18" h "0.171" tw "0.006" tf "0.0095" r "0.015"))
      ((= perf "200") (setq b "0.20" h "0.190" tw "0.0065" tf "0.01" r "0.018"))
      ((= perf "220") (setq b "0.22" h "0.210" tw "0.007" tf "0.011" r "0.018"))
      ((= perf "240") (setq b "0.24" h "0.230" tw "0.0075" tf "0.012" r "0.021"))
      ((= perf "260") (setq b "0.26" h "0.250" tw "0.0075" tf "0.0125" r "0.024"))
      ((= perf "280") (setq b "0.28" h "0.270" tw "0.008" tf "0.013" r "0.024"))
      ((= perf "300") (setq b "0.30" h "0.290" tw "0.0085" tf "0.014" r "0.027"))
      ((= perf "320") (setq b "0.30" h "0.310" tw "0.009" tf "0.0155" r "0.027"))
      ((= perf "340") (setq b "0.30" h "0.330" tw "0.0095" tf "0.0165" r "0.027"))
      ((= perf "360") (setq b "0.30" h "0.350" tw "0.01" tf "0.0175" r "0.027"))
      ((= perf "400") (setq b "0.30" h "0.390" tw "0.011" tf "0.019" r "0.027"))
      ((= perf "450") (setq b "0.30" h "0.440" tw "0.0115" tf "0.021" r "0.027"))
      ((= perf "500") (setq b "0.30" h "0.490" tw "0.012" tf "0.023" r "0.027"))
      ((= perf "550") (setq b "0.30" h "0.540" tw "0.0125" tf "0.024" r "0.027"))
      ((= perf "600") (setq b "0.30" h "0.590" tw "0.013" tf "0.025" r "0.027"))
      ((= perf "650") (setq b "0.30" h "0.640" tw "0.0135" tf "0.026" r "0.027"))
      ((= perf "700") (setq b "0.30" h "0.690" tw "0.0145" tf "0.027" r "0.027"))
      ((= perf "800") (setq b "0.30" h "0.790" tw "0.015" tf "0.028" r "0.03"))
      ((= perf "900") (setq b "0.30" h "0.890" tw "0.016" tf "0.030" r "0.03"))
      ((= perf "1000") (setq b "0.30" h "0.990" tw "0.0165" tf "0.031" r "0.03"))
    )
  )
  Descripción de la variable "perf"
  Descripción de las variables locales

```

```

((= secc "HEB")
  (cond ((= perf "100") (setq b "0.10" h "0.100" tw "0.006" tf "0.01" r "0.012"))
        ((= perf "120") (setq b "0.12" h "0.120" tw "0.0065" tf "0.011" r "0.012"))
        ((= perf "140") (setq b "0.14" h "0.140" tw "0.007" tf "0.012" r "0.012"))
        ((= perf "160") (setq b "0.16" h "0.160" tw "0.008" tf "0.013" r "0.015"))
        ((= perf "180") (setq b "0.18" h "0.180" tw "0.0085" tf "0.014" r "0.015"))
        ((= perf "200") (setq b "0.20" h "0.200" tw "0.009" tf "0.015" r "0.018"))
        ((= perf "220") (setq b "0.22" h "0.220" tw "0.0095" tf "0.016" r "0.018"))
        ((= perf "240") (setq b "0.24" h "0.240" tw "0.010" tf "0.017" r "0.021"))
        ((= perf "260") (setq b "0.26" h "0.260" tw "0.01" tf "0.0175" r "0.024"))
        ((= perf "280") (setq b "0.28" h "0.280" tw "0.0105" tf "0.018" r "0.024"))
        ((= perf "300") (setq b "0.30" h "0.300" tw "0.011" tf "0.019" r "0.027"))
        ((= perf "320") (setq b "0.30" h "0.320" tw "0.0115" tf "0.0205" r "0.027"))
        ((= perf "340") (setq b "0.30" h "0.340" tw "0.012" tf "0.0215" r "0.027"))
        ((= perf "360") (setq b "0.30" h "0.360" tw "0.0125" tf "0.0225" r "0.027"))
        ((= perf "400") (setq b "0.30" h "0.400" tw "0.0135" tf "0.024" r "0.027"))
        ((= perf "450") (setq b "0.30" h "0.450" tw "0.014" tf "0.026" r "0.027"))
        ((= perf "500") (setq b "0.30" h "0.500" tw "0.0145" tf "0.028" r "0.027"))
        ((= perf "550") (setq b "0.30" h "0.550" tw "0.015" tf "0.029" r "0.027"))
        ((= perf "600") (setq b "0.30" h "0.600" tw "0.0155" tf "0.03" r "0.027"))
        ((= perf "650") (setq b "0.30" h "0.650" tw "0.016" tf "0.031" r "0.027"))
        ((= perf "700") (setq b "0.30" h "0.700" tw "0.017" tf "0.032" r "0.027"))
        ((= perf "800") (setq b "0.30" h "0.800" tw "0.0175" tf "0.033" r "0.030"))
        ((= perf "900") (setq b "0.30" h "0.900" tw "0.0185" tf "0.035" r "0.030"))
        ((= perf "1000") (setq b "0.30" h "1.000" tw "0.019" tf "0.036" r "0.030"))
  )
)

```

```

((= secc "HEM")
  (cond ((= perf "100") (setq b "0.106" h "0.120" tw "0.012" tf "0.02" r "0.012"))
        ((= perf "120") (setq b "0.126" h "0.140" tw "0.0125" tf "0.021" r "0.012"))
        ((= perf "140") (setq b "0.146" h "0.160" tw "0.013" tf "0.022" r "0.012"))
        ((= perf "160") (setq b "0.166" h "0.180" tw "0.014" tf "0.023" r "0.015"))
        ((= perf "180") (setq b "0.186" h "0.200" tw "0.0145" tf "0.024" r "0.015"))
        ((= perf "200") (setq b "0.206" h "0.220" tw "0.015" tf "0.025" r "0.018"))
        ((= perf "220") (setq b "0.226" h "0.240" tw "0.0155" tf "0.026" r "0.018"))
        ((= perf "240") (setq b "0.248" h "0.270" tw "0.018" tf "0.032" r "0.021"))
        ((= perf "260") (setq b "0.268" h "0.290" tw "0.018" tf "0.0325" r "0.024"))
        ((= perf "280") (setq b "0.288" h "0.310" tw "0.0185" tf "0.033" r "0.024"))
        ((= perf "300") (setq b "0.310" h "0.340" tw "0.021" tf "0.039" r "0.027"))
        ((= perf "320") (setq b "0.309" h "0.359" tw "0.021" tf "0.04" r "0.027"))
        ((= perf "340") (setq b "0.309" h "0.377" tw "0.021" tf "0.04" r "0.027"))
        ((= perf "360") (setq b "0.308" h "0.395" tw "0.021" tf "0.04" r "0.027"))
        ((= perf "400") (setq b "0.307" h "0.432" tw "0.021" tf "0.04" r "0.027"))
        ((= perf "450") (setq b "0.307" h "0.478" tw "0.021" tf "0.04" r "0.027"))
        ((= perf "500") (setq b "0.306" h "0.524" tw "0.021" tf "0.04" r "0.027"))
        ((= perf "550") (setq b "0.306" h "0.572" tw "0.021" tf "0.04" r "0.027"))
        ((= perf "600") (setq b "0.305" h "0.620" tw "0.021" tf "0.04" r "0.027"))
        ((= perf "650") (setq b "0.305" h "0.668" tw "0.021" tf "0.04" r "0.027"))
        ((= perf "700") (setq b "0.304" h "0.716" tw "0.021" tf "0.04" r "0.027"))
        ((= perf "800") (setq b "0.303" h "0.814" tw "0.021" tf "0.04" r "0.030"))
        ((= perf "900") (setq b "0.302" h "0.910" tw "0.021" tf "0.04" r "0.030"))
        ((= perf "1000") (setq b "0.302" h "1.010" tw "0.019" tf "0.04" r "0.030"))
  )
)

```



```

(= secc "IPE")
(cond ((= perf "80") (setq b "0.046" h "0.08" tw "0.0038" tf "0.0052" r "0.005"))
      ((= perf "100") (setq b "0.055" h "0.10" tw "0.0041" tf "0.0057" r "0.007"))
      ((= perf "120") (setq b "0.064" h "0.12" tw "0.0044" tf "0.0063" r "0.007"))
      ((= perf "140") (setq b "0.073" h "0.14" tw "0.0047" tf "0.0069" r "0.007"))
      ((= perf "160") (setq b "0.082" h "0.16" tw "0.005" tf "0.0074" r "0.009"))
      ((= perf "180") (setq b "0.091" h "0.18" tw "0.0053" tf "0.008" r "0.009"))
      ((= perf "200") (setq b "0.100" h "0.20" tw "0.0056" tf "0.0085" r "0.012"))
      ((= perf "220") (setq b "0.110" h "0.22" tw "0.0059" tf "0.0092" r "0.012"))
      ((= perf "240") (setq b "0.120" h "0.24" tw "0.0062" tf "0.0098" r "0.015"))
      ((= perf "270") (setq b "0.135" h "0.27" tw "0.0066" tf "0.0102" r "0.015"))
      ((= perf "300") (setq b "0.150" h "0.30" tw "0.0071" tf "0.0107" r "0.015"))
      ((= perf "330") (setq b "0.160" h "0.33" tw "0.0075" tf "0.0115" r "0.018"))
      ((= perf "360") (setq b "0.170" h "0.36" tw "0.008" tf "0.0127" r "0.018"))
      ((= perf "400") (setq b "0.180" h "0.40" tw "0.0086" tf "0.0135" r "0.021"))
      ((= perf "450") (setq b "0.190" h "0.45" tw "0.0094" tf "0.0146" r "0.021"))
      ((= perf "500") (setq b "0.200" h "0.50" tw "0.0102" tf "0.016" r "0.021"))
      ((= perf "550") (setq b "0.210" h "0.55" tw "0.0111" tf "0.0172" r "0.024"))
      ((= perf "600") (setq b "0.220" h "0.60" tw "0.012" tf "0.019" r "0.024"))
)
)

(= secc "IPN")
(cond ((= perf "80") (setq b "0.042" h "0.08" tw "0.0039" tf "0.0059" r "0.0059" r2 "0.0023"))
      ((= perf "100") (setq b "0.050" h "0.10" tw "0.0045" tf "0.0068" r "0.0068" r2 "0.0027"))
      ((= perf "120") (setq b "0.058" h "0.12" tw "0.0051" tf "0.0077" r "0.0077" r2 "0.0031"))
      ((= perf "140") (setq b "0.066" h "0.14" tw "0.0057" tf "0.0086" r "0.0086" r2 "0.0034"))
      ((= perf "160") (setq b "0.074" h "0.16" tw "0.0063" tf "0.0095" r "0.0095" r2 "0.0038"))
      ((= perf "180") (setq b "0.082" h "0.18" tw "0.0069" tf "0.0104" r "0.0104" r2 "0.0041"))
      ((= perf "200") (setq b "0.090" h "0.20" tw "0.0075" tf "0.0113" r "0.0113" r2 "0.0045"))
      ((= perf "220") (setq b "0.098" h "0.22" tw "0.0081" tf "0.0122" r "0.0122" r2 "0.0049"))
      ((= perf "240") (setq b "0.106" h "0.24" tw "0.0087" tf "0.0131" r "0.0131" r2 "0.0052"))
      ((= perf "260") (setq b "0.113" h "0.26" tw "0.0094" tf "0.0141" r "0.0141" r2 "0.0056"))
      ((= perf "280") (setq b "0.119" h "0.28" tw "0.0101" tf "0.0152" r "0.0152" r2 "0.0061"))
      ((= perf "300") (setq b "0.125" h "0.30" tw "0.0108" tf "0.0162" r "0.0162" r2 "0.0065"))
      ((= perf "320") (setq b "0.131" h "0.32" tw "0.0115" tf "0.0173" r "0.0173" r2 "0.0069"))
      ((= perf "340") (setq b "0.137" h "0.34" tw "0.0122" tf "0.0183" r "0.0183" r2 "0.0073"))
      ((= perf "360") (setq b "0.143" h "0.36" tw "0.0130" tf "0.0195" r "0.0195" r2 "0.0078"))
      ((= perf "380") (setq b "0.149" h "0.38" tw "0.0137" tf "0.0205" r "0.0205" r2 "0.0082"))
      ((= perf "400") (setq b "0.155" h "0.40" tw "0.0144" tf "0.0216" r "0.0216" r2 "0.0086"))
      ((= perf "450") (setq b "0.170" h "0.45" tw "0.0162" tf "0.0243" r "0.0243" r2 "0.0097"))
      ((= perf "500") (setq b "0.185" h "0.50" tw "0.0180" tf "0.0270" r "0.0270" r2 "0.0108"))
      ((= perf "550") (setq b "0.200" h "0.55" tw "0.0190" tf "0.0300" r "0.0300" r2 "0.0119"))
      ((= perf "600") (setq b "0.215" h "0.60" tw "0.0216" tf "0.0324" r "0.0324" r2 "0.0130"))
)
)
)

```

(setq ang (grd->rad (atof giro)) *Se procesa el parámetro con la descripción del giro de la sección.*

dir (IfcDirection fich (cos ang) (sin ang) nil)

orig (IfcCartesianPoint fich 0.0 0.0 nil)

)

(setq IfcIndex (1+ IfcIndex)) *Se crea un sistema de referencia local y se guarda en el fichero IFC.*

(write-line (strcat "#" (itoa IfcIndex) "= IFCAXIS2PLACEMENT2D( " orig " , " dir " );") fich)

(setq orientacion (strcat "#" (itoa IfcIndex)))

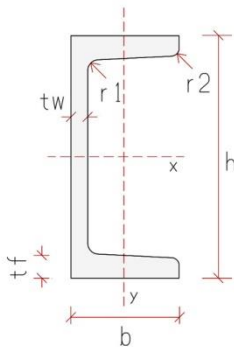
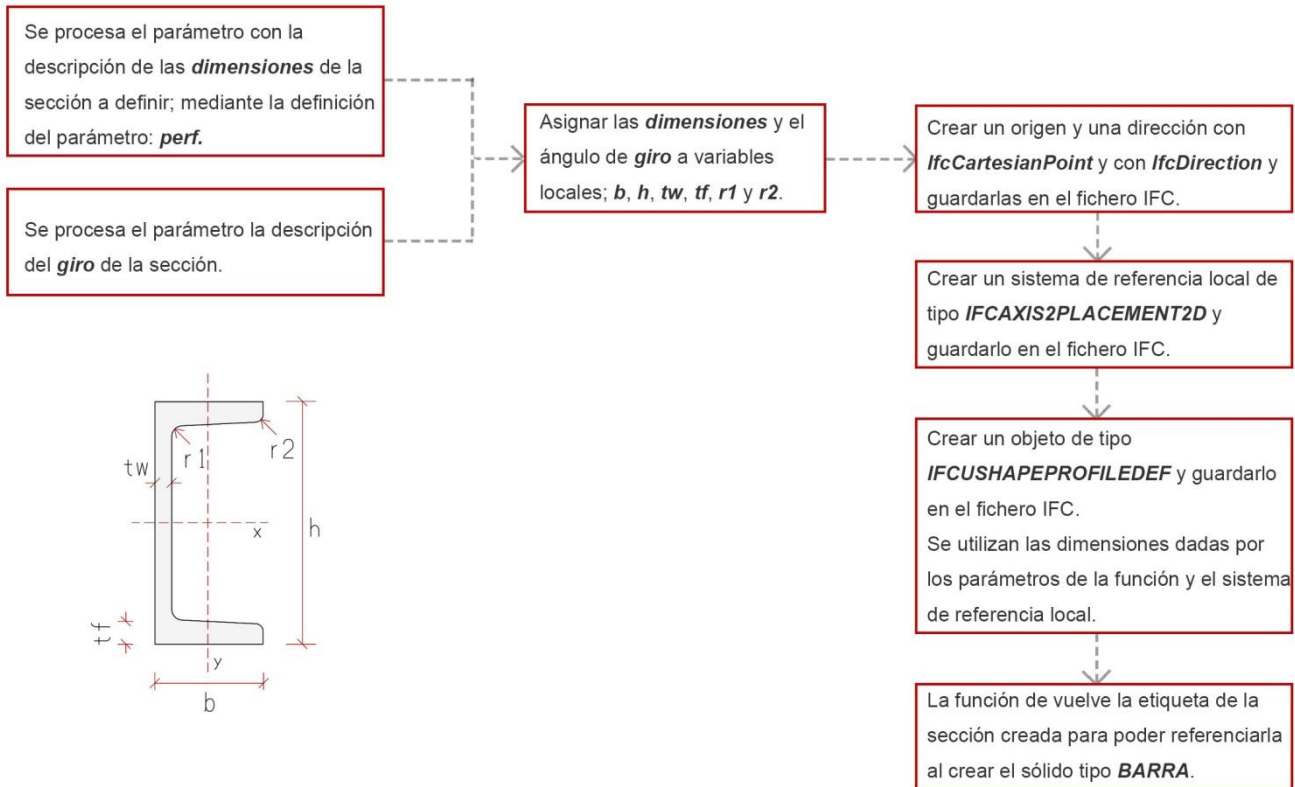
(setq IfcIndex (1+ IfcIndex))  
 res (strcat "#" (itoa IfcIndex) "= IFCISHAPEPROFILEDEF(.AREA., " secc perf " ,  
 " orientacion " , " b " , " h " , " tw " , " tf " , " r ");")

(write-line res fich) *Se crea un objeto de tipo IFCISHAPEPROFILEDEF y se guarda en el fichero IFC.*

(strcat "#" (itoa IfcIndex)) *Esta función devuelve la etiqueta de la sección creada para poder referenciarla.*

)

**IFCUSHapeProfileDef**

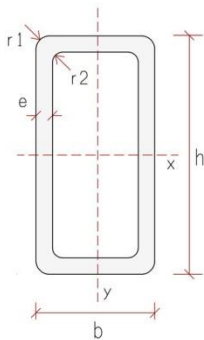
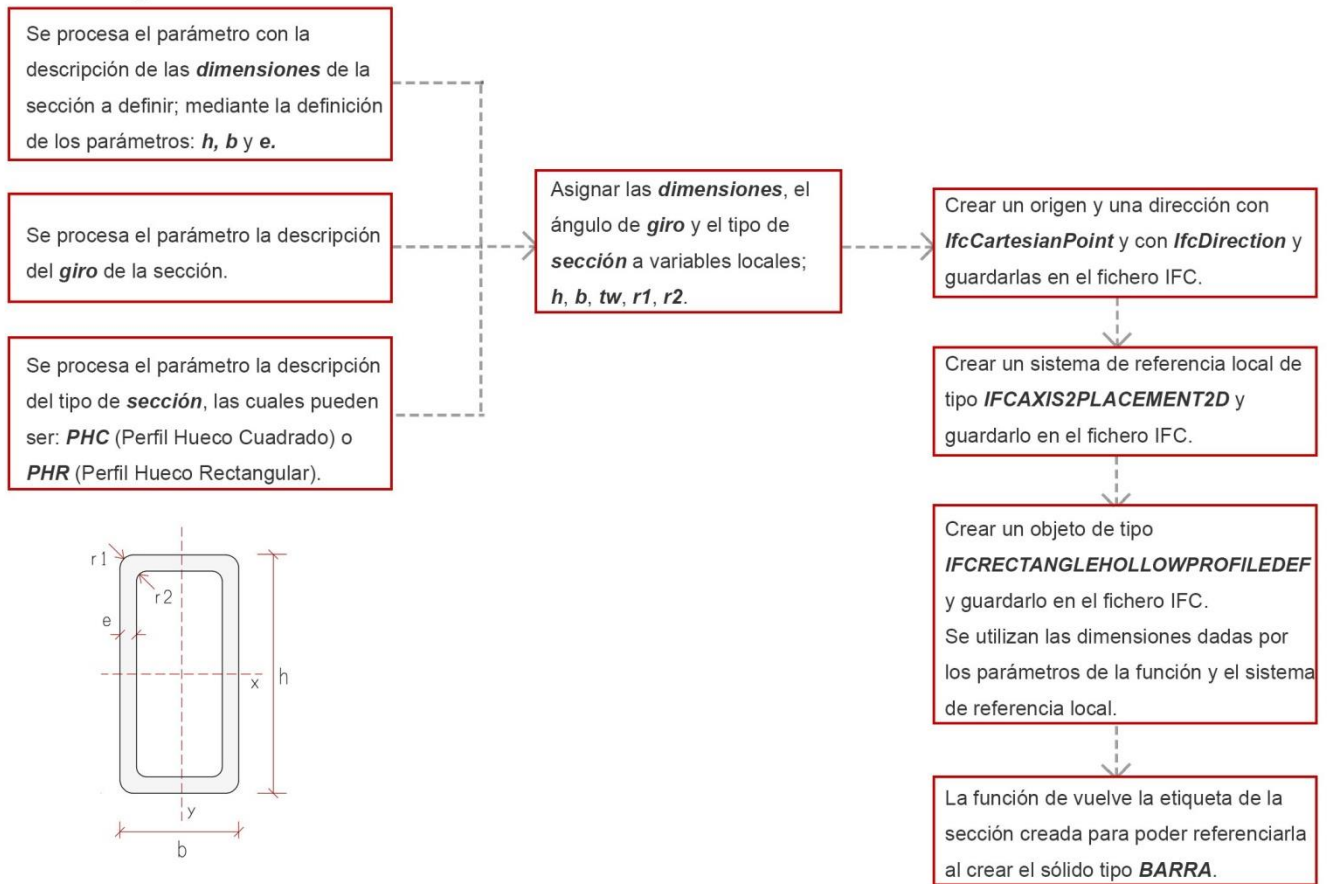


**Figura 4.4.**

Esquema explicativo de funcionamiento de la función IfcUShapeProfileDef. (elaboración propia)

```
(defun IfcUShapeProfileDef (fich perf giro / b h tw tf r1 r2 res orig ang dir lst)
  Genera un IfcUShapeProfileDef a partir de las dimensiones de una sección en U.
  (cond ((= perf "80") (setq h "0.08" b "0.045" tw "0.006" tf "0.008" r1 "0.008" r2 "0.004"))
        ((= perf "100") (setq h "0.10" b "0.050" tw "0.006" tf "0.0085" r1 "0.0085" r2 "0.0045"))
        ((= perf "120") (setq h "0.12" b "0.055" tw "0.007" tf "0.009" r1 "0.009" r2 "0.0045"))
        ((= perf "140") (setq h "0.14" b "0.060" tw "0.007" tf "0.010" r1 "0.010" r2 "0.005"))
        ((= perf "160") (setq h "0.16" b "0.065" tw "0.0075" tf "0.0105" r1 "0.0105" r2 "0.0055"))
        ((= perf "180") (setq h "0.18" b "0.070" tw "0.008" tf "0.011" r1 "0.011" r2 "0.0055"))
        ((= perf "200") (setq h "0.20" b "0.075" tw "0.0085" tf "0.0115" r1 "0.0115" r2 "0.006"))
        ((= perf "220") (setq h "0.22" b "0.080" tw "0.009" tf "0.0125" r1 "0.0125" r2 "0.0065"))
        ((= perf "240") (setq h "0.24" b "0.085" tw "0.0095" tf "0.013" r1 "0.013" r2 "0.0065"))
        ((= perf "260") (setq h "0.26" b "0.090" tw "0.010" tf "0.014" r1 "0.014" r2 "0.007"))
        ((= perf "280") (setq h "0.28" b "0.095" tw "0.010" tf "0.015" r1 "0.015" r2 "0.0075"))
        ((= perf "300") (setq h "0.30" b "0.100" tw "0.010" tf "0.016" r1 "0.016" r2 "0.008"))
        ((= perf "320") (setq h "0.32" b "0.100" tw "0.014" tf "0.0175" r1 "0.0175" r2 "0.00875"))
        ((= perf "350") (setq h "0.35" b "0.100" tw "0.014" tf "0.016" r1 "0.016" r2 "0.008"))
        ((= perf "380") (setq h "0.38" b "0.102" tw "0.0135" tf "0.016" r1 "0.016" r2 "0.008"))
        ((= perf "400") (setq h "0.40" b "0.110" tw "0.014" tf "0.018" r1 "0.018" r2 "0.009"))
  )
  Definición de la variable "perf"
  Definición de las variables locales.
)
```

### IFCRectangleHollowProfileDef



**Figura 4.5.**

Esquema explicativo de funcionamiento de la función `IfcRectangleHollowProfileDef`. (elaboración propia)

```

(setq ang (grd->rad (atof giro)) — Se procesa el parámetro con la descripción del giro de la sección.
  dir (IfcDirection fich (cos ang) (sin ang) nil)
  orig (IfcCartesianPoint fich 0.0 0.0 nil)
)

(setq IfcIndex (1+ IfcIndex)) — Se crea un sistema de referencia local y se guarda en el fichero IFC.
(write-line (strcat "#" (itoa IfcIndex) "= IFCAXIS2PLACEMENT2D( " orig ", " dir ");") fich)
(setq orientacion (strcat "#" (itoa IfcIndex)))

(setq IfcIndex (1+ IfcIndex)
  res (strcat "#" (itoa IfcIndex) "= IFCUSHAPEPROFILEDEF(.AREA., "" secc perf "" ,
    " orientacion ", " h ", " b ", " tw ", " tf ", " r1 ", " r2 ", $, $);")
)
(write-line res fich) — Se crea un objeto de tipo IFCUSHAPEPROFILEDEF y se guarda en el fichero IFC.
(strcat "#" (itoa IfcIndex)) — Esta función devuelve la etiqueta de la sección creada para poder referenciarla.
)
  
```

```

(defun IfcRectangleHollowProfileDef (fich secc perf giro / b h e r2 r1 res orig ang dir lst)
  Genera un IfcRectangleHollowProfileDef a partir de las dimensiones de rectángulo.
  (setq dimens (std-strsplit perf "x") Se procesa el parámetro con la descripción de las dimensiones a definir.
    h (nth 0 dimens)
    b (nth 1 dimens)
    e (nth 2 dimens)
  )

  (cond ((= secc "PHC") Se procesa el parámetro con la descripción del tipo de sección.
    (cond ((= h "40")
      (cond ((= b "40")
        (cond ((= e "2.6") (setq h "0.04" b "0.04" tw "0.0026" r1 "0.0039" r2 "0.0026"))
              ((= e "3.2") (setq h "0.04" b "0.04" tw "0.0032" r1 "0.0045" r2 "0.0032"))
              ((= e "4") (setq h "0.04" b "0.04" tw "0.004" r1 "0.0060" r2 "0.0040"))
              ((= e "5") (setq h "0.04" b "0.04" tw "0.005" r1 "0.0075" r2 "0.0050"))
            )
        )
      )
    )
  )
  )

  ((= h "50")
    (cond ((= b "50")
      (cond ((= e "2.6") (setq h "0.05" b "0.05" tw "0.0026" r1 "0.0039" r2 "0.0026"))
            ((= e "3.2") (setq h "0.05" b "0.05" tw "0.0032" r1 "0.0048" r2 "0.0032"))
            ((= e "4") (setq h "0.05" b "0.05" tw "0.004" r1 "0.0060" r2 "0.0040"))
            ((= e "5") (setq h "0.05" b "0.05" tw "0.005" r1 "0.0075" r2 "0.0050"))
            ((= e "6.3") (setq h "0.05" b "0.05" tw "0.0063" r1 "0.0095" r2 "0.0063"))
          )
      )
    )
  )

  ((= h "60")
    (cond ((= b "60")
      (cond ((= e "2.6") (setq h "0.06" b "0.06" tw "0.0026" r1 "0.0039" r2 "0.0026"))
            ((= e "3.2") (setq h "0.06" b "0.06" tw "0.0032" r1 "0.0048" r2 "0.0032"))
            ((= e "4") (setq h "0.06" b "0.06" tw "0.004" r1 "0.0060" r2 "0.0040"))
            ((= e "5") (setq h "0.06" b "0.06" tw "0.005" r1 "0.0075" r2 "0.0050"))
            ((= e "6.3") (setq h "0.06" b "0.06" tw "0.0063" r1 "0.0095" r2 "0.0063"))
            ((= e "8") (setq h "0.06" b "0.06" tw "0.0080" r1 "0.0120" r2 "0.0080"))
          )
      )
    )
  )

  ((= h "70")
    (cond ((= b "70")
      (cond ((= e "3.2") (setq h "0.07" b "0.07" tw "0.0032" r1 "0.0048" r2 "0.0032"))
            ((= e "4") (setq h "0.07" b "0.07" tw "0.004" r1 "0.0060" r2 "0.0040"))
            ((= e "5") (setq h "0.07" b "0.07" tw "0.005" r1 "0.0075" r2 "0.0050"))
            ((= e "6.3") (setq h "0.07" b "0.07" tw "0.0063" r1 "0.0095" r2 "0.0063"))
            ((= e "8") (setq h "0.07" b "0.07" tw "0.0080" r1 "0.0120" r2 "0.0080"))
          )
      )
    )
  )
)

```

*Definición de variables locales para esa sección, espesor y dimensiones concretas.*

```

((= h "80")
  (cond ((= b "80")
    (cond ((= e "3.2") (setq h "0.08" b "0.08" tw "0.0032" r1 "0.0048" r2 "0.0032"))
      ((= e "4") (setq h "0.08" b "0.08" tw "0.004" r1 "0.0060" r2 "0.0040"))
      ((= e "5") (setq h "0.08" b "0.08" tw "0.005" r1 "0.0075" r2 "0.0050"))
      ((= e "6.3") (setq h "0.08" b "0.08" tw "0.0063" r1 "0.0095" r2 "0.0063"))
      ((= e "8") (setq h "0.08" b "0.08" tw "0.008" r1 "0.0120" r2 "0.0080"))
    )
  )
)
)
)
((= h "90")
  (cond ((= b "90")
    (cond ((= e "4") (setq h "0.09" b "0.09" tw "0.004" r1 "0.0060" r2 "0.004"))
      ((= e "5") (setq h "0.09" b "0.09" tw "0.005" r1 "0.0075" r2 "0.005"))
      ((= e "6.3") (setq h "0.09" b "0.09" tw "0.0063" r1 "0.0095" r2 "0.0063"))
      ((= e "8") (setq h "0.09" b "0.09" tw "0.008" r1 "0.0120" r2 "0.0080"))
    )
  )
)
)
)
((= h "100")
  (cond ((= b "100")
    (cond ((= e "4") (setq h "0.1" b "0.1" tw "0.0040" r1 "0.0060" r2 "0.0040"))
      ((= e "5") (setq h "0.1" b "0.1" tw "0.0050" r1 "0.0075" r2 "0.0050"))
      ((= e "6.3") (setq h "0.1" b "0.1" tw "0.0063" r1 "0.0095" r2 "0.0063"))
      ((= e "8") (setq h "0.1" b "0.1" tw "0.008" r1 "0.0120" r2 "0.0080"))
      ((= e "10") (setq h "0.1" b "0.1" tw "0.01" r1 "0.0150" r2 "0.01"))
    )
  )
)
)
)
((= h "120")
  (cond ((= b "120")
    (cond ((= e "5") (setq h "0.12" b "0.12" tw "0.005" r1 "0.0075" r2 "0.005"))
      ((= e "6.3") (setq h "0.12" b "0.12" tw "0.0063" r1 "0.0095" r2 "0.0063"))
      ((= e "8") (setq h "0.12" b "0.12" tw "0.008" r1 "0.0120" r2 "0.0080"))
      ((= e "10") (setq h "0.12" b "0.12" tw "0.01" r1 "0.0150" r2 "0.01"))
      ((= e "12.5") (setq h "0.12" b "0.12" tw "0.0125" r1 "0.0188" r2 "0.0125"))
    )
  )
)
)
)
((= h "140")
  (cond ((= b "140")
    (cond ((= e "5") (setq h "0.14" b "0.14" tw "0.005" r1 "0.0075" r2 "0.005"))
      ((= e "6.3") (setq h "0.14" b "0.14" tw "0.0063" r1 "0.0095" r2 "0.0063"))
      ((= e "8") (setq h "0.14" b "0.14" tw "0.008" r1 "0.0120" r2 "0.0080"))
      ((= e "10") (setq h "0.14" b "0.14" tw "0.01" r1 "0.0150" r2 "0.01"))
      ((= e "12.5") (setq h "0.14" b "0.14" tw "0.0125" r1 "0.0188" r2 "0.0125"))
    )
  )
)
)
)
((= h "150")
  (cond ((= b "150")
    (cond ((= e "5") (setq h "0.15" b "0.15" tw "0.005" r1 "0.0075" r2 "0.005"))
      ((= e "6.3") (setq h "0.15" b "0.15" tw "0.0063" r1 "0.0095" r2 "0.0063"))
      ((= e "8") (setq h "0.15" b "0.15" tw "0.008" r1 "0.0120" r2 "0.0080"))
      ((= e "10") (setq h "0.15" b "0.15" tw "0.01" r1 "0.0150" r2 "0.01"))
      ((= e "12.5") (setq h "0.15" b "0.15" tw "0.0125" r1 "0.0188" r2 "0.0125"))
      ((= e "14.2") (setq h "0.15" b "0.15" tw "0.0142" r1 "0.0213" r2 "0.0142"))
      ((= e "16") (setq h "0.15" b "0.15" tw "0.016" r1 "0.0240" r2 "0.016"))
    )
  )
)
)
)
)

```

```

((= h "160")
  (cond ((= b "160")
    (cond ((= e "5") (setq h "0.16" b "0.16" tw "0.005" r1 "0.0075" r2 "0.005"))
      ((= e "6.3") (setq h "0.16" b "0.16" tw "0.0063" r1 "0.0095" r2 "0.0063"))
      ((= e "8") (setq h "0.16" b "0.16" tw "0.008" r1 "0.0120" r2 "0.0080"))
      ((= e "10") (setq h "0.16" b "0.16" tw "0.01" r1 "0.0150" r2 "0.01"))
      ((= e "12.5") (setq h "0.16" b "0.16" tw "0.0125" r1 "0.0188" r2 "0.0125"))
      ((= e "14.2") (setq h "0.16" b "0.16" tw "0.0142" r1 "0.0213" r2 "0.0142"))
      ((= e "16") (setq h "0.16" b "0.16" tw "0.016" r1 "0.0240" r2 "0.016"))
    )
  )
)
)
)
((= h "180")
  (cond ((= b "180")
    (cond ((= e "5") (setq h "0.18" b "0.18" tw "0.005" r1 "0.0075" r2 "0.005"))
      ((= e "6.3") (setq h "0.18" b "0.18" tw "0.0063" r1 "0.0095" r2 "0.0063"))
      ((= e "8") (setq h "0.18" b "0.18" tw "0.008" r1 "0.0120" r2 "0.0080"))
      ((= e "10") (setq h "0.18" b "0.18" tw "0.01" r1 "0.0150" r2 "0.01"))
      ((= e "12.5") (setq h "0.18" b "0.18" tw "0.0125" r1 "0.0188" r2 "0.0125"))
      ((= e "14.2") (setq h "0.18" b "0.18" tw "0.0142" r1 "0.0213" r2 "0.0142"))
      ((= e "16") (setq h "0.18" b "0.18" tw "0.016" r1 "0.0240" r2 "0.016"))
    )
  )
)
)
)
((= h "200")
  (cond ((= b "200")
    (cond ((= e "5") (setq h "0.2" b "0.2" tw "0.005" r1 "0.0075" r2 "0.005"))
      ((= e "6.3") (setq h "0.2" b "0.2" tw "0.0063" r1 "0.0095" r2 "0.0063"))
      ((= e "8") (setq h "0.2" b "0.2" tw "0.008" r1 "0.0120" r2 "0.0080"))
      ((= e "10") (setq h "0.2" b "0.2" tw "0.01" r1 "0.0150" r2 "0.01"))
      ((= e "12.5") (setq h "0.2" b "0.2" tw "0.0125" r1 "0.0188" r2 "0.0125"))
      ((= e "14.2") (setq h "0.2" b "0.2" tw "0.0142" r1 "0.0213" r2 "0.0142"))
      ((= e "16") (setq h "0.2" b "0.2" tw "0.016" r1 "0.0240" r2 "0.016"))
    )
  )
)
)
)
((= h "220")
  (cond ((= b "220")
    (cond ((= e "6.3") (setq h "0.22" b "0.22" tw "0.0063" r1 "0.0095" r2 "0.0063"))
      ((= e "8") (setq h "0.22" b "0.22" tw "0.008" r1 "0.0120" r2 "0.0080"))
      ((= e "10") (setq h "0.22" b "0.22" tw "0.01" r1 "0.0150" r2 "0.01"))
      ((= e "12.5") (setq h "0.22" b "0.22" tw "0.0125" r1 "0.0188" r2 "0.0125"))
      ((= e "14.2") (setq h "0.22" b "0.22" tw "0.0142" r1 "0.0213" r2 "0.0142"))
      ((= e "16") (setq h "0.22" b "0.22" tw "0.016" r1 "0.0240" r2 "0.016"))
    )
  )
)
)
)
((= h "250")
  (cond ((= b "250")
    (cond ((= e "6.3") (setq h "0.25" b "0.25" tw "0.0063" r1 "0.0095" r2 "0.0063"))
      ((= e "8") (setq h "0.25" b "0.25" tw "0.008" r1 "0.0120" r2 "0.0080"))
      ((= e "10") (setq h "0.25" b "0.25" tw "0.01" r1 "0.0150" r2 "0.01"))
      ((= e "12.5") (setq h "0.25" b "0.25" tw "0.0125" r1 "0.0188" r2 "0.0125"))
      ((= e "14.2") (setq h "0.25" b "0.25" tw "0.0142" r1 "0.0213" r2 "0.0142"))
      ((= e "16") (setq h "0.25" b "0.25" tw "0.016" r1 "0.0240" r2 "0.016"))
    )
  )
)
)
)
)

```

```

(= h "260")
  (cond ((= b "260")
    (cond ((= e "6.3") (setq h "0.26" b "0.26" tw "0.0063" r1 "0.0095" r2 "0.0063"))
      ((= e "8") (setq h "0.26" b "0.26" tw "0.008" r1 "0.0120" r2 "0.0080"))
      ((= e "10") (setq h "0.26" b "0.26" tw "0.01" r1 "0.0150" r2 "0.01"))
      ((= e "12.5") (setq h "0.26" b "0.26" tw "0.0125" r1 "0.0180" r2 "0.0125"))
      ((= e "14.2") (setq h "0.26" b "0.26" tw "0.0142" r1 "0.0213" r2 "0.0142"))
      ((= e "16") (setq h "0.26" b "0.26" tw "0.016" r1 "0.0240" r2 "0.016"))
    )
  )
)
)
(= h "300")
  (cond ((= b "300")
    (cond ((= e "6.3") (setq h "0.3" b "0.3" tw "0.0063" r1 "0.0095" r2 "0.0063"))
      ((= e "8") (setq h "0.3" b "0.3" tw "0.008" r1 "0.0120" r2 "0.0080"))
      ((= e "10") (setq h "0.3" b "0.3" tw "0.01" r1 "0.0150" r2 "0.01"))
      ((= e "12.5") (setq h "0.3" b "0.3" tw "0.0125" r1 "0.0180" r2 "0.0125"))
      ((= e "14.2") (setq h "0.3" b "0.3" tw "0.0142" r1 "0.0213" r2 "0.0142"))
      ((= e "16") (setq h "0.3" b "0.3" tw "0.016" r1 "0.0240" r2 "0.016"))
    )
  )
)
)
(= h "350")
  (cond ((= b "350")
    (cond ((= e "8") (setq h "0.35" b "0.35" tw "0.008" r1 "0.0120" r2 "0.0080"))
      ((= e "10") (setq h "0.35" b "0.35" tw "0.01" r1 "0.0150" r2 "0.01"))
      ((= e "12.5") (setq h "0.35" b "0.35" tw "0.0125" r1 "0.0180" r2 "0.0125"))
      ((= e "14.2") (setq h "0.35" b "0.35" tw "0.0142" r1 "0.0213" r2 "0.0142"))
      ((= e "16") (setq h "0.35" b "0.35" tw "0.016" r1 "0.0240" r2 "0.016"))
    )
  )
)
)
(= h "400")
  (cond ((= b "400")
    (cond ((= e "10") (setq h "0.4" b "0.4" tw "0.01" r1 "0.0150" r2 "0.01"))
      ((= e "12.5") (setq h "0.4" b "0.4" tw "0.0125" r1 "0.0180" r2 "0.0125"))
      ((= e "14.2") (setq h "0.4" b "0.4" tw "0.0142" r1 "0.0213" r2 "0.0142"))
      ((= e "16") (setq h "0.4" b "0.4" tw "0.016" r1 "0.0240" r2 "0.016"))
      ((= e "20") (setq h "0.4" b "0.4" tw "0.020" r1 "0.0300" r2 "0.020"))
    )
  )
)
)
)
(= secc "PHR")
  (cond ((= h "50")
    (cond ((= b "30")
      (cond ((= e "2.6") (setq h "0.05" b "0.03" tw "0.0026" r1 "0.0039" r2 "0.0026"))
        ((= e "3.2") (setq h "0.05" b "0.03" tw "0.0032" r1 "0.0048" r2 "0.0032"))
        ((= e "4") (setq h "0.05" b "0.03" tw "0.004" r1 "0.0060" r2 "0.0040"))
        ((= e "5") (setq h "0.05" b "0.03" tw "0.005" r1 "0.0075" r2 "0.0050"))
      )
    )
  )
)
)
(= h "60")
  (cond ((= b "40")
    (cond ((= e "2.6") (setq h "0.06" b "0.04" tw "0.0026" r1 "0.0039" r2 "0.0026"))
      ((= e "3.2") (setq h "0.06" b "0.04" tw "0.0032" r1 "0.0048" r2 "0.0032"))
      ((= e "4") (setq h "0.06" b "0.04" tw "0.004" r1 "0.0060" r2 "0.0040"))
      ((= e "5") (setq h "0.06" b "0.04" tw "0.005" r1 "0.0075" r2 "0.0050"))
      ((= e "6.3") (setq h "0.06" b "0.04" tw "0.0063" r1 "0.0095" r2 "0.0063"))
    )
  )
)
)
(= h "80")
  (cond ((= b "40")
    (cond ((= e "3.2") (setq h "0.08" b "0.04" tw "0.0032" r1 "0.0048" r2 "0.0032"))
      ((= e "4") (setq h "0.08" b "0.04" tw "0.004" r1 "0.0060" r2 "0.0040"))
      ((= e "5") (setq h "0.08" b "0.04" tw "0.005" r1 "0.0075" r2 "0.0050"))
      ((= e "6.3") (setq h "0.08" b "0.04" tw "0.0063" r1 "0.0095" r2 "0.0063"))
      ((= e "8") (setq h "0.08" b "0.04" tw "0.008" r1 "0.012" r2 "0.0080"))
    )
  )
)
)
)

```

```

((= h "90")
  (cond ((= b "50")
    (cond ((= e "3.2") (setq h "0.09" b "0.05" tw "0.0032" r1 "0.0048" r2 "0.0032"))
          ((= e "4") (setq h "0.09" b "0.05" tw "0.004" r1 "0.0060" r2 "0.0040"))
          ((= e "5") (setq h "0.09" b "0.05" tw "0.005" r1 "0.0075" r2 "0.0050"))
          ((= e "6.3") (setq h "0.09" b "0.05" tw "0.0063" r1 "0.0095" r2 "0.0063"))
          ((= e "8") (setq h "0.09" b "0.05" tw "0.008" r1 "0.012" r2 "0.0080"))
        )
      )
    )
  )
)
((= h "100")
  (cond ((= b "50")
    (cond ((= e "3.2") (setq h "0.1" b "0.05" tw "0.0032" r1 "0.0048" r2 "0.0032"))
          ((= e "4") (setq h "0.1" b "0.05" tw "0.004" r1 "0.0060" r2 "0.0040"))
          ((= e "5") (setq h "0.1" b "0.05" tw "0.005" r1 "0.0075" r2 "0.0050"))
          ((= e "6.3") (setq h "0.1" b "0.05" tw "0.0063" r1 "0.0095" r2 "0.0063"))
          ((= e "8") (setq h "0.1" b "0.05" tw "0.008" r1 "0.012" r2 "0.0080"))
        )
      )
    )
  )
  (cond ((= b "60")
    (cond ((= e "3.2") (setq h "0.1" b "0.06" tw "0.0032" r1 "0.0048" r2 "0.0032"))
          ((= e "4") (setq h "0.1" b "0.06" tw "0.004" r1 "0.0060" r2 "0.0040"))
          ((= e "5") (setq h "0.1" b "0.06" tw "0.005" r1 "0.0075" r2 "0.0050"))
          ((= e "6.3") (setq h "0.1" b "0.06" tw "0.0063" r1 "0.0095" r2 "0.0063"))
          ((= e "8") (setq h "0.1" b "0.06" tw "0.008" r1 "0.012" r2 "0.0080"))
        )
      )
    )
  )
)
)
)

```

```

((= h "120")
  (cond ((= b "60")
    (cond ((= e "4") (setq h "0.12" b "0.06" tw "0.004" r1 "0.0060" r2 "0.0040"))
          ((= e "5") (setq h "0.12" b "0.06" tw "0.005" r1 "0.0075" r2 "0.0050"))
          ((= e "6.3") (setq h "0.12" b "0.06" tw "0.0063" r1 "0.0095" r2 "0.0063"))
          ((= e "8") (setq h "0.12" b "0.06" tw "0.008" r1 "0.012" r2 "0.0080"))
          ((= e "10") (setq h "0.12" b "0.06" tw "0.010" r1 "0.015" r2 "0.010"))
        )
      )
    )
  )
  (cond ((= b "80")
    (cond ((= e "4") (setq h "0.12" b "0.08" tw "0.004" r1 "0.0060" r2 "0.0040"))
          ((= e "5") (setq h "0.12" b "0.08" tw "0.005" r1 "0.0075" r2 "0.0050"))
          ((= e "6.3") (setq h "0.12" b "0.08" tw "0.0063" r1 "0.0095" r2 "0.0063"))
          ((= e "8") (setq h "0.12" b "0.08" tw "0.008" r1 "0.012" r2 "0.0080"))
          ((= e "10") (setq h "0.12" b "0.08" tw "0.010" r1 "0.015" r2 "0.010"))
        )
      )
    )
  )
)
)
((= h "140")
  (cond ((= b "80")
    (cond ((= e "4") (setq h "0.14" b "0.08" tw "0.004" r1 "0.0060" r2 "0.0040"))
          ((= e "5") (setq h "0.14" b "0.08" tw "0.005" r1 "0.0075" r2 "0.0050"))
          ((= e "6.3") (setq h "0.14" b "0.08" tw "0.0063" r1 "0.0095" r2 "0.0063"))
          ((= e "8") (setq h "0.14" b "0.08" tw "0.008" r1 "0.012" r2 "0.0080"))
          ((= e "10") (setq h "0.14" b "0.08" tw "0.010" r1 "0.015" r2 "0.010"))
        )
      )
    )
  )
)
)
)

```



```

((= h "150")
  (cond ((= b "100")
    (cond ((= e "4") (setq h "0.15" b "0.1" tw "0.004" r1 "0.0060" r2 "0.0040"))
          ((= e "5") (setq h "0.15" b "0.1" tw "0.005" r1 "0.0075" r2 "0.0050"))
          ((= e "6.3") (setq h "0.15" b "0.1" tw "0.0063" r1 "0.0095" r2 "0.0063"))
          ((= e "8") (setq h "0.15" b "0.1" tw "0.008" r1 "0.012" r2 "0.0080"))
          ((= e "10") (setq h "0.15" b "0.1" tw "0.010" r1 "0.015" r2 "0.010"))
          ((= e "12.5") (setq h "0.15" b "0.1" tw "0.0125" r1 "0.0188" r2 "0.0125"))
        )
    )
  )
)
((= h "160")
  (cond ((= b "80")
    (cond ((= e "4") (setq h "0.16" b "0.08" tw "0.004" r1 "0.0060" r2 "0.0040"))
          ((= e "5") (setq h "0.16" b "0.08" tw "0.005" r1 "0.0075" r2 "0.0050"))
          ((= e "6.3") (setq h "0.16" b "0.08" tw "0.0063" r1 "0.0095" r2 "0.0063"))
          ((= e "8") (setq h "0.16" b "0.08" tw "0.008" r1 "0.012" r2 "0.0080"))
          ((= e "10") (setq h "0.16" b "0.08" tw "0.010" r1 "0.015" r2 "0.010"))
          ((= e "12.5") (setq h "0.16" b "0.08" tw "0.0125" r1 "0.0188" r2 "0.0125"))
        )
    )
  )
)
((= h "180")
  (cond ((= b "100")
    (cond ((= e "4") (setq h "0.18" b "0.1" tw "0.004" r1 "0.0060" r2 "0.0040"))
          ((= e "5") (setq h "0.18" b "0.1" tw "0.005" r1 "0.0075" r2 "0.0050"))
          ((= e "6.3") (setq h "0.18" b "0.1" tw "0.0063" r1 "0.0095" r2 "0.0063"))
          ((= e "8") (setq h "0.18" b "0.1" tw "0.008" r1 "0.012" r2 "0.0080"))
          ((= e "10") (setq h "0.18" b "0.1" tw "0.010" r1 "0.015" r2 "0.010"))
          ((= e "12.5") (setq h "0.18" b "0.1" tw "0.0125" r1 "0.0188" r2 "0.0125"))
        )
    )
  )
)
((= h "200")
  (cond ((= b "100")
    (cond ((= e "4") (setq h "0.2" b "0.1" tw "0.004" r1 "0.0060" r2 "0.0040"))
          ((= e "5") (setq h "0.2" b "0.1" tw "0.005" r1 "0.0075" r2 "0.0050"))
          ((= e "6.3") (setq h "0.2" b "0.1" tw "0.0063" r1 "0.0095" r2 "0.0063"))
          ((= e "8") (setq h "0.2" b "0.1" tw "0.008" r1 "0.012" r2 "0.0080"))
          ((= e "10") (setq h "0.2" b "0.1" tw "0.010" r1 "0.015" r2 "0.010"))
          ((= e "12.5") (setq h "0.2" b "0.1" tw "0.0125" r1 "0.015" r2 "0.0125"))
          ((= e "16") (setq h "0.2" b "0.1" tw "0.016" r1 "0.015" r2 "0.016"))
        )
    )
    ((= b "120")
    (cond ((= e "6.3") (setq h "0.2" b "0.12" tw "0.0063" r1 "0.0095" r2 "0.0063"))
          ((= e "8") (setq h "0.2" b "0.12" tw "0.008" r1 "0.012" r2 "0.0080"))
          ((= e "10") (setq h "0.2" b "0.12" tw "0.010" r1 "0.015" r2 "0.010"))
          ((= e "12.5") (setq h "0.2" b "0.12" tw "0.0125" r1 "0.015" r2 "0.0125"))
        )
    )
  )
)
((= h "250")
  (cond ((= b "150")
    (cond ((= e "6.3") (setq h "0.25" b "0.15" tw "0.0063" r1 "0.0095" r2 "0.0063"))
          ((= e "8") (setq h "0.25" b "0.15" tw "0.008" r1 "0.012" r2 "0.0080"))
          ((= e "10") (setq h "0.25" b "0.15" tw "0.010" r1 "0.015" r2 "0.010"))
          ((= e "12.5") (setq h "0.25" b "0.15" tw "0.0125" r1 "0.0188" r2 "0.0125"))
          ((= e "14.2") (setq h "0.25" b "0.15" tw "0.0142" r1 "0.0213" r2 "0.0142"))
          ((= e "16") (setq h "0.25" b "0.15" tw "0.0160" r1 "0.0240" r2 "0.0160"))
        )
    )
  )
)

```



```

(setq ang (grd->rad (atof giro)) Se procesa el parámetro con la descripción del giro de la sección.
  dir (IfcDirection fich (cos ang) (sin ang) nil)
  orig (IfcCartesianPoint fich 0.0 0.0 nil)
)

(setq IfcIndex (1+ IfcIndex)) Se crea un sistema de referencia local y se guarda en el fichero IFC.
(write-line (strcat "#" (itoa IfcIndex) "= IFCAxis2Placement2D( " orig ", " dir ");") fich)
(setq orientacion (strcat "#" (itoa IfcIndex)))

(setq IfcIndex (1+ IfcIndex)
  res (strcat "#" (itoa IfcIndex) "= IFCRECTANGLEHOLLOWPROFILEDEF(.AREA., "" secc perf "" ,
    " orientacion ", " b ", " h ", " tw ", " r2 ", " r1 ");")
)
(write-line res fich) Se crea un objeto de tipo IFCRECTANGLEHOLLOWPROFILEDEF y se guarda en fichero IFC.
(strcat "#" (itoa IfcIndex)) Esta función devuelve la etiqueta de la sección creada para poder referenciarla.
)

```

### IFCCircleHollowProfileDef

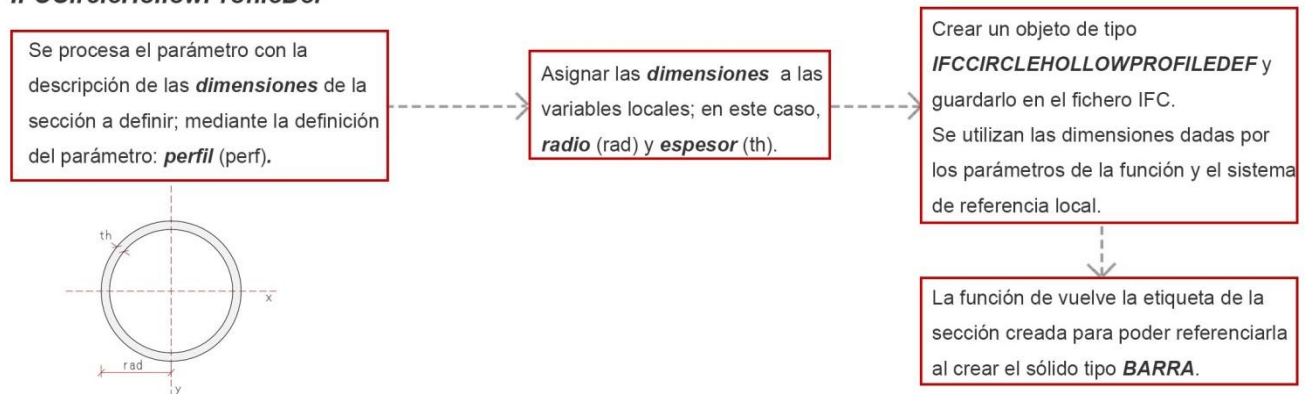


Figura 4.6.  
Esquema explicativo de funcionamiento de la función IfcCircleHollowProfileDef. (elaboración propia)

```

(defun IfcCircleHollowProfileDef (fich perf / dimens rad th res)
  Se genera un IfcCircleHollowProfileDef a partir de las dimensiones de un círculo.
  (setq dimens (std-strsplit perf "x") Se procesa el parámetro con la descripción de las dimensiones de la sección a definir.
    rad (rtos (* 0.5 (* 0.001 (atoi (nth 0 dimens)))) 2 5) Se asignan las dimensiones a las variables locales; radio (rad) y espesor (th).
    th (rtos (* 0.001 (atoi (nth 1 dimens)))) 2 5)
    IfcIndex (1+ IfcIndex)
    res (strcat "#" (itoa IfcIndex) "= IFCCIRCLEHOLLOWPROFILEDEF(.AREA., 'CIR-H" perf "' , $,
      " rad ", " th ");")
  )
  Se crea un objeto de tipo IFCCIRCLEHOLLOWPROFILEDEF y se guarda en el fichero IFC.
  (write-line res fich)
  (strcat "#" (itoa IfcIndex)) Esta función devuelve la etiqueta de la sección creada para poder referenciarla.
)

```

**IFCRectangleProfileDef**

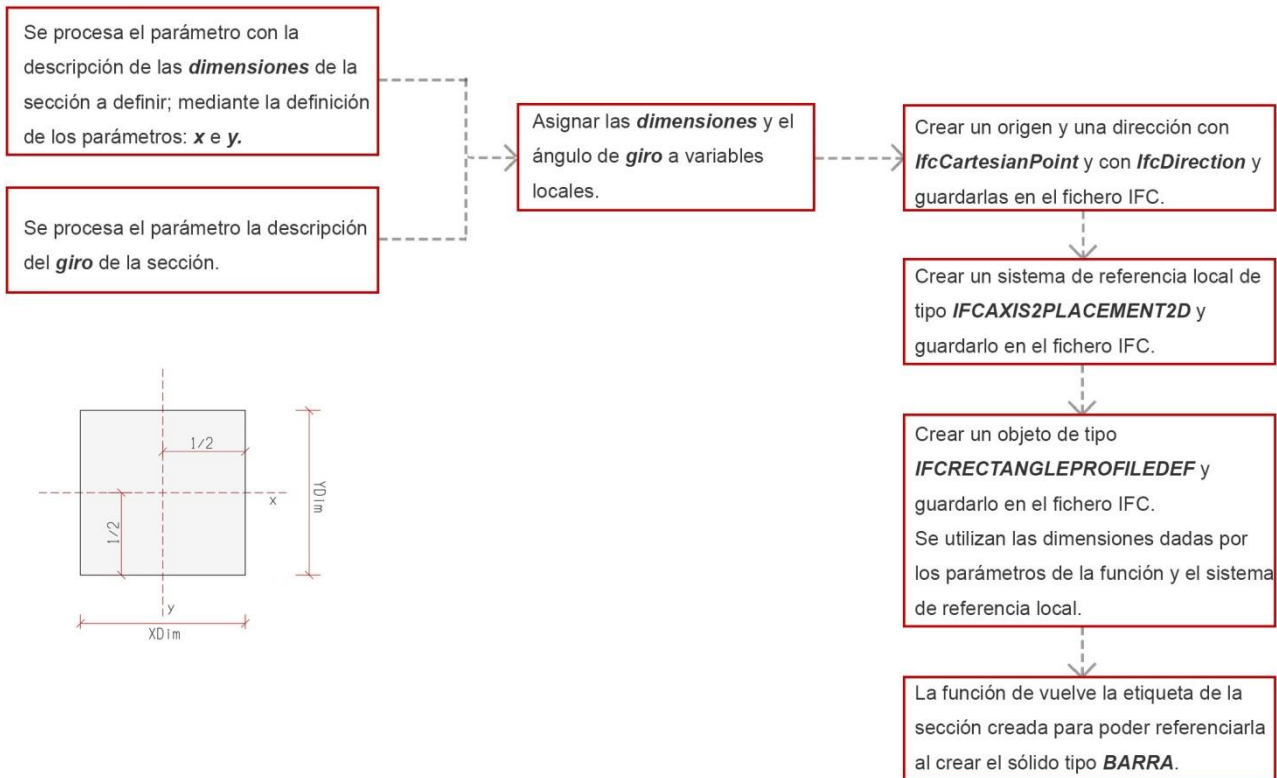


Figura 4.7.

Esquema explicativo de funcionamiento de la función `IfcRectangleProfileDef`. (elaboración propia)

```

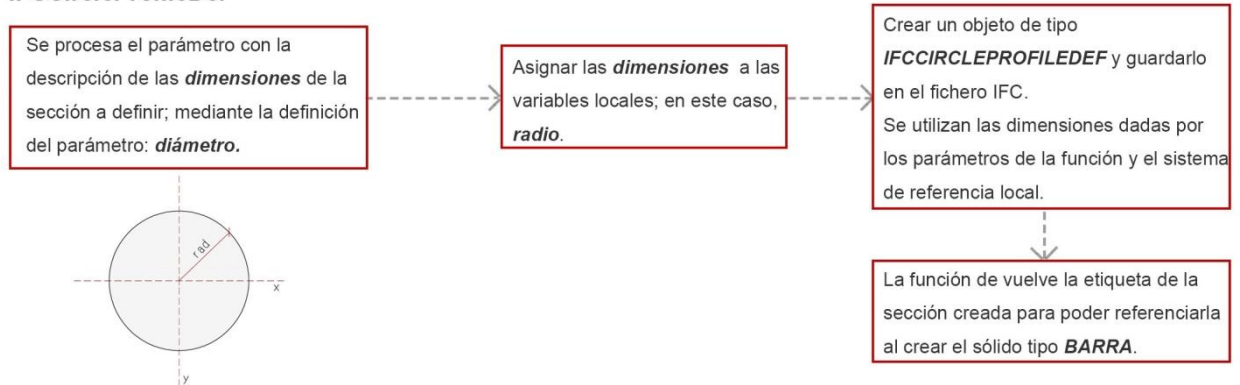
(defun IfcRectangleProfileDef (fich dim giro / dimens x y res orig ang dir lst)
  Se genera un IfcRectangleProfileDef a partir de las dimensiones de un rectángulo.
  (setq dimens (std-strsplit dim "x") Se procesa el parámetro con la descripción de las dimensiones de la sección a definir.
    x (rtos (* 0.001 (atoi (nth 0 dimens))) 2 5) Se asignan las dimensiones a las variables locales; x e y.
    y (rtos (* 0.001 (atoi (nth 1 dimens))) 2 5)
  )

  (setq ang (grd->rad (atof giro)) Se procesa el parámetro de la descripción del giro de la sección.
    dir (IfcDirection fich (cos ang) (sin ang) nil)
    orig (IfcCartesianPoint fich 0.0 0.0 nil)
  )

  (setq IfcIndex (1+ IfcIndex)) Se crea un sistema de referencia local y se guarda en el fichero IFC.
  (write-line (strcat "#" (itoa IfcIndex) "= IFCAXIS2PLACEMENT2D( " orig ", " dir ");") fich)
  (setq orientacion (strcat "#" (itoa IfcIndex)))

  (setq IfcIndex (1+ IfcIndex))
  res (strcat "#" (itoa IfcIndex) "= IFCRECTANGLEPROFILEDEF(.AREA., 'REC " dim "' ,
    " orientacion ", " x ", " y ");")
  ) Se crea un objeto de tipo IFCRECTANGLEPROFILEDEF y se guarda en el fichero IFC.
  (write-line res fich) Esta función devuelve la etiqueta de la sección creada para poder referenciarla.
  (strcat "#" (itoa IfcIndex))
)
  
```

### IFCCircleProfileDef



**Figura 4.8.**

Esquema explicativo de funcionamiento de la función *IfcCircleProfileDef*. (elaboración propia)

```
(defun IfcCircleProfileDef (fich diam / rad res)
  (setq rad (rtos (* 0.5 (* 0.001 (atoi (nth 0 dim)))) 2 5))
  (IfcIndex (1+ IfcIndex))
  res (strcat "#" (itoa IfcIndex) "= IFCCIRCLEPROFILEDEF(.AREA., 'CIRC " dim "' , $, " rad ");")
)
(write-line res fich)
(strcat "#" (itoa IfcIndex))
)
```

Se genera un *IfcCircleProfileDef* a partir de las dimensiones de un círculo.

Se asignan la dimensión a la variable local: *rad*.

Se crea un objeto de tipo *IFCCIRCLEPROFILEDEF* y se guarda en el fichero IFC.

Esta función devuelve la etiqueta de la sección creada para poder referenciarla.

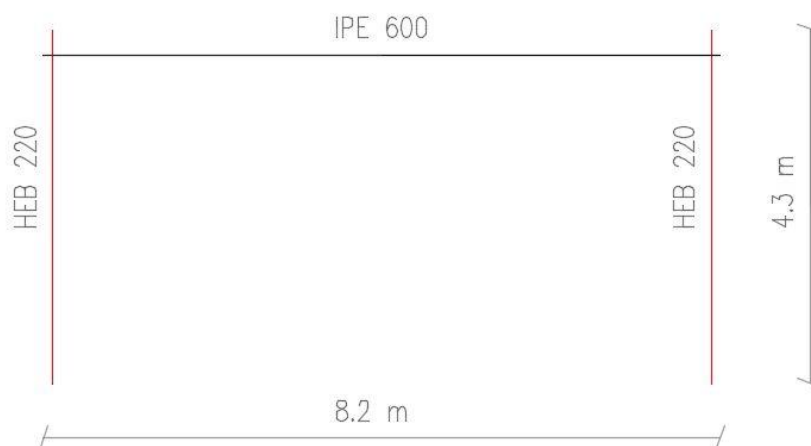
### 4.3. APLICACIÓN DEL PROTOCOLO CREADO

Anteriormente se ha mostrado la metodología de generación del protocolo de los elementos estructurales, seguidamente para que este proceso pueda realizarse de manera acertada, se ha debido introducir este protocolo en la base de datos del programa Architrave (mediante la intervención del equipo creador de dicho programa) para que éste pueda evaluar los documentos y proyectos de formato AutoLISP al teclear mediante éste la función “BIM” en la barra de comandos de AutoCAD, y generar a partir de ellos el documento IFC correspondiente.

Para ilustrar el funcionamiento del protocolo de exportación se ha aplicado en este TFG a dos casos. El primero, es un **pórtico simple** para el que se utilizan las secciones creadas en la parte del protocolo explicada en el apartado anterior. El segundo, es un proyecto más complejo y conocido por todos: el “**Crown Hall**” de Mies Van Der Rohe. En este modelo se utilizan, además, **secciones de usuario** (sustancialmente más complejas) y se incluye la **definición de los contornos de mallado de elementos finitos** que representan los muros y forjados también anteriormente definidos.

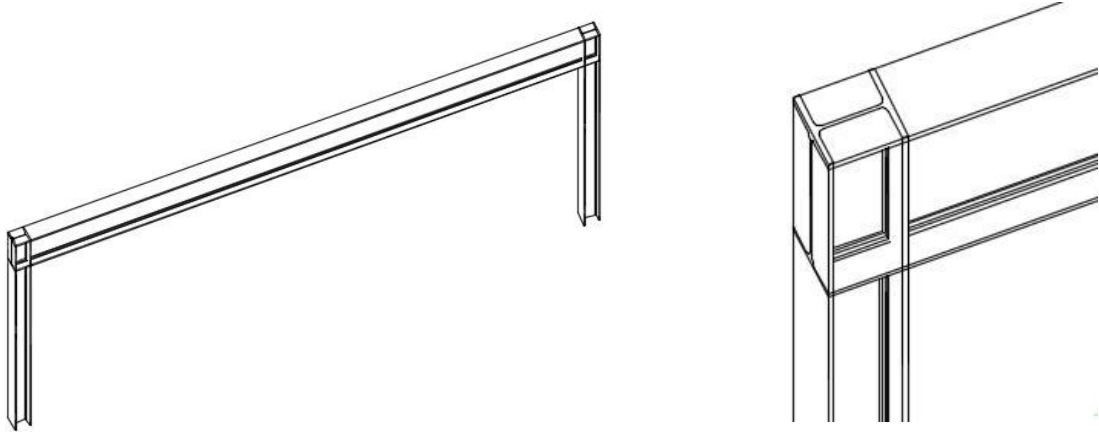
#### 4.3.2. CREACIÓN DE UN PÓRTICO EN ARCHITRAVE.

El pórtico creado se ha realizado con algunas de las secciones realizadas en el protocolo y tiene la siguiente composición y medidas a la hora de realizarlo en Architrave (en el cual se añaden únicamente las dimensiones de los ejes de los elementos estructurales y se les asigna la tipología de sección a cada uno de ellos):



**Figura 4.9.**  
Ejes del pórtico creado. (elaboración propia)

Para la realización de los ejes se han tenido en cuenta los espesores de la viga y los pilares para que cuando éstos tengan volumetría se ajusten correctamente a los tamaños:



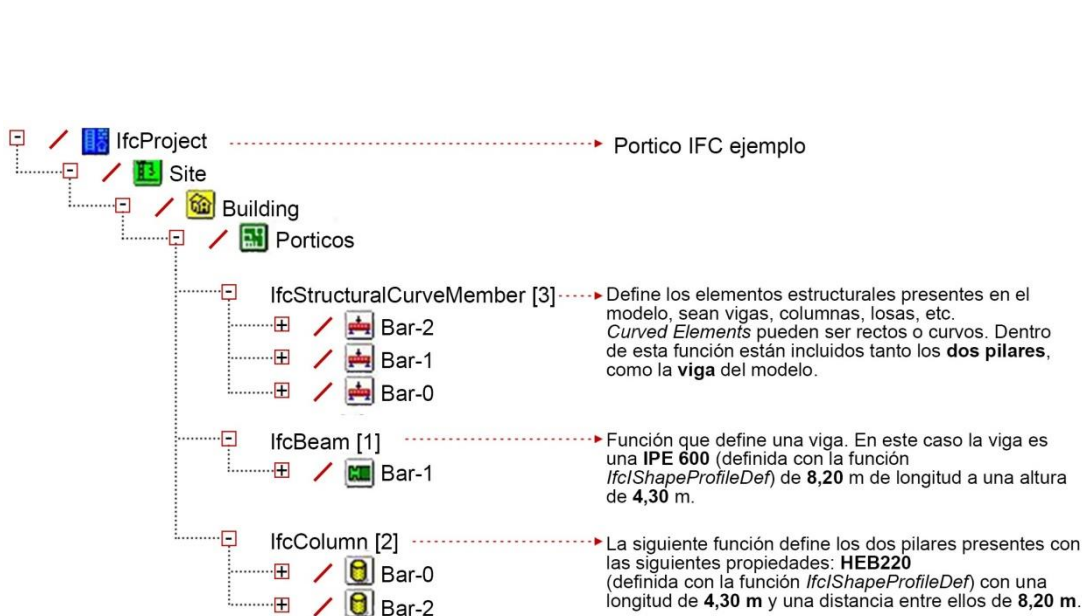
**Figura 4.10.**  
*Volumetría y detalle esquina pórtico (elaboración propia).*

Al finalizar la creación del pórtico, se teclea en la barra de comandos la palabra “BIM” a continuación te pide qué es aquello que queremos convertir a IFC, seleccionamos el pórtico y aceptamos. Mediante esta palabra el programa accede al protocolo creado para poder convertir el texto de programación generado automáticamente en Visual LISP mediante Architrave, a un texto de programación IFC, haciendo así el documento accesible por todos los programas BIM.

Para saber aquello que queríamos obtener, es decir, el documento IFC que contuviera la información necesaria para poder obtener el pórtico explicado, en primer lugar creamos y analizamos aquello que contendría un archivo IFC del dicho pórtico, que es lo que se detalla a continuación.

### 4.3.1.1. ANÁLISIS DEL DOCUMENTO IFC OBTENIDO

El documento como tal estará escrito en negro y los comentarios adjuntos en color granate para diferenciarlo. Como anteriormente hemos comentado, todo archivo IFC tiene una estructura definida, a continuación explicaremos por partes cada una de ellas y aquello que se va definiendo. El documento IFC como tal estará escrito en negro y los comentarios adjuntos en color granate para ir comprendiendo aquello que se va definiendo. Antes de analizar el documento IFC, se muestra un esquema de las funciones IFC utilizadas y aquello que van realizando.



**Figura 4.11.**  
Funciones IFC utilizadas para la creación del pórtico. (elaboración propia)



ISO-10303-21; Estándar del modelo de datos para intercambio de productos utilizado para el documento

HEADER;FILE\_DESCRIPTION(('ViewDefinition [CoordinationView\_V2.0]', Tipología d definiciones de vista del modelo utilizada entre las definidas por BuildingSMART.

Se seleccionan las entidades del modelo de Architrave que hemos creado que se quieran exportar al fichero IFC.

```
'Option [Drawing Scale: 100.000000]',  
'Option [Global Unique Identifiers (GUID): Keep existing]',  
'Option [Elements to export: Visible elements (on all stories)]',  
'Option [Partial Structure Display: Entire Model]',  
'Option [IFC Domain: All]', 'Option [Structural Function: All Elements]',  
'Option [Convert Grid elements: On]',  
'Option [Export geometries that Participates in Collision Detection only: On]',  
'Option [Multi-skin complex geometries: Building element parts]',  
'Option [Elements in Solid Element Operations: Extruded/revolved]',  
'Option [Elements with junctions: Extruded/revolved without junctions]',  
'Option [Slabs with slanted edge(s): Extruded]',  
'Option [IFC Site Geometry: As boundary representation (BRep)]',  
'Option [Properties To Export: All properties]',  
) , '2;1');
```

A continuación se da la información del nombre del archivo , la fecha, las iniciales de quien lo ha creado, el programa y la tipología de programación que utiliza dicho programa. Por otro lado, informa de la versión IFC en la cual se basa el presente documento.

```
FILE_NAME('pórtico.ifc', '2016-09-01T13:15:58', ('DV'), ('Architrave'), 'VLisp', 'VLisp', '');
```

```
FILE_SCHEMA(('IFC2X3'));
```

```
ENDSEC;
```

```
DATA;
```

Construcción de la cabecera del fichero IFC incluyendo : el código para identificar el proyecto, propietario de quien ha realizado el cambio, ha añadido o modificado el proyecto, la organización, la persona, la aplicación mediante se ha realizado

El código para identificar el proyecto:

```
#1= IFCPROJECT('3$z59C4WzEHBmIHem5Xtl', #2, 'IfcProject', $, $, $, $, (#3), #70);
```

La empresa, persona, acción que haya hecho el autor en el proyecto; añadir, modificar, borrar, etc. y fecha de la modificación.

#2= **IFCOWNERHISTORY**(#5,#6,\$,.ADDED.,\$,,\$,2016090113160067);

La empresa y persona que haya realizado el proyecto:

#5= **IFCPERSONANDORGANIZATION**(#7,#8,\$);

Actor del proyecto o modificación:

#7= **IFCPERSON**(\$,'Desire',\$,,\$,\$,\$);

Programas y versiones con las cuales se ha creado el documento:

#8= **IFCORGANIZATION**(\$,'Architrave',\$,,\$);

#6= **IFCAPPLICATION**(#9,'v.15','Architrave','IFC 2X3 Architrave');

#9= **IFCORGANIZATION**(\$,'Architrave',\$,,\$);

La siguiente función define el contexto geométrico en el cual se realiza el proyecto, definiendo ejes, direcciones, etc.

#3= **IFCGEOMETRICREPRESENTATIONCONTEXT**(\$,'Model',3,1.000000000000E-5,#10,#11);

Definición de ejes 3D para cuando se precise utilizarlos:

#10= **IFCAXIS2PLACEMENT3D**(#12,#13,#14);

#12= **IFCCARTESIANPOINT**((0.,0.,0.));

Definición de direcciones;

#13= **IFCDIRECTION**((0.,0.,1.));

#14= **IFCDIRECTION**((1.,0.,0.));

#11= **IFCDIRECTION**((0.,1.));

Definición de la ubicación del proyecto;

#20= **IFCSITE**('0veYtoED58QB2tuxAKPb0S',#2,'Site',\$,,\$,#21,\$,\$,.ELEMENT.,\$,,\$,0.,\$,,\$);

#21= **IFCLOCALPLACEMENT**(\$,#10);

#30= **IFCBUILDING**('3mXJRuOSr2shTKwjLY7L4k',#2,'Building',\$,,\$,#31,\$,\$,.ELEMENT.,\$,,\$);

#31= **IFCLOCALPLACEMENT**(#21,#10);

Definición de aquello que se va a representar del proyecto; ejes, volúmenes, plantas...

#50= **IFCGEOMETRICREPRESENTATIONSUBCONTEXT**('Axis','Model',\*,\*,\*,#3,\$,.MODEL\_VIEW.,\$);

#51= **IFCGEOMETRICREPRESENTATIONSUBCONTEXT**('Body','Model',\*,\*,\*,#3,\$,.MODEL\_VIEW.,\$);

#52= **IFCGEOMETRICREPRESENTATIONSUBCONTEXT**('Profile','Model',\*,\*,\*,#3,\$,.MODEL\_VIEW.,\$);



#112= **IFCCOLOURRGB**(\$,0.658823529412,0.662745098039,0.619607843137);

Definición de tipologías de renderizado que se han utilizado:

#120=  
**IFCSURFACESTYLERENDERING**(#110,0.,IFCNORMALISEDRA  
TIO MEASURE(0.85),\$,,\$,IFCNORMALISEDRA  
TIO MEASURE(0.1),\$,,NOTDEFINED.);

#121=  
**IFCSURFACESTYLERENDERING**(#111,0.,IFCNORMALISEDRA  
TIO MEASURE(0.85),\$,,\$,IFCNORMALISEDRA  
TIO MEASURE(0.1),\$,,NOTDEFINED.);

#122=  
**IFCSURFACESTYLERENDERING**(#112,0.,IFCNORMALISEDRA  
TIO MEASURE(0.85),\$,,\$,IFCNORMALISEDRA  
TIO MEASURE(0.1),\$,,NOTDEFINED.);

Definición de tipologías de renderizado de los materiales ya definidos:

#130= **IFCSURFACESTYLE**('Acero',.BOTH.,(#120));

#131= **IFCSURFACESTYLE**('Madera',.BOTH.,(#121));

#132= **IFCSURFACESTYLE**('Hormigon',.BOTH.,(#122));

Definición del estilo de representación asignado a cada material.

#140= **IFCPRESENTATIONSTYLEASSIGNMENT**((#130));

#141= **IFCPRESENTATIONSTYLEASSIGNMENT**((#131));

#142= **IFCPRESENTATIONSTYLEASSIGNMENT**((#132));

Definición de presentación de un elemento de representación geométrica.

#150= **IFCSTYLEDITEM**(\$,(#140),\$);

#151= **IFCSTYLEDITEM**(\$,(#141),\$);

#152= **IFCSTYLEDITEM**(\$,(#142),\$);

#160= **IFCSTYLEDREPRESENTATION**(#51,\$,\$,(#150));

#161= **IFCSTYLEDREPRESENTATION**(#51,\$,\$,(#151));

#162= **IFCSTYLEDREPRESENTATION**(#51,\$,\$,(#152));

#170= **IFCMATERIALDEFINITIONREPRESENTATION**(\$,\$,(#160),#100);

#171= **IFCMATERIALDEFINITIONREPRESENTATION**(\$,\$,(#161),#101);

#172= **IFCMATERIALDEFINITIONREPRESENTATION**(\$,\$,(#162),#102);

Definición de una dirección (la de los pilares):

#1001= **IFCDIRECTION**((1.00000, 0.00000));

Definición de un punto :

#1002= **IFCCARTESIANPOINT**((0.000000, 0.000000));

Definición de un eje:

#1003= **IFCAXIS2PLACEMENT2D**( #1002, #1001);

Creación de la sección transversal del primer pilar en HEB 220, el cual se encuentra en el eje anteriormente definido y en el cual se definen las dimensiones de éste, en el orden que establece BuildingSMART.

#1004= **IFCISHAPEPROFILEDEF**(.AREA., 'HEB220', #1003, 0.22, 0.220, 0.0095, 0.016, 0.018);

Se define otra dirección (la de la viga):

#1005= **IFCDIRECTION**((1.00000, 0.00000));

#1006= **IFCCARTESIANPOINT**((0.000000, 0.000000));

#1007= **IFCAXIS2PLACEMENT2D**( #1006, #1005);

Creación de otra sección transversal de barra, esta vez es una viga ya que tiene la dirección perpendicular al pilar, y creamos mediante esta función la viga IPE 600, la cual se encuentra en los ejes definidos anteriormente.

#1008= **IFCISHAPEPROFILEDEF**(.AREA., 'IPE600', #1007, 0.220, 0.60, 0.012, 0.019, 0.024);

#1009= **IFCDIRECTION**((0.00000, 0.00000, 1.00000));

#1010= **IFCEXTRUDEDAREASOLID**(#1004, #10, #1009, 4.3000);

Se definen 2 puntos:

#1011= **IFCCARTESIANPOINT**((0.000000, 0.000000, 0.000000));

#1012= **IFCCARTESIANPOINT**((0.000000, 0.000000, 4.300000));

Y se crea una polilínea de esos puntos, de la cual resulta una línea de ahí que solo se defina el punto inicial y el final.

#1013= **IFCPOLYLINE**((#1011, #1012));

#1014= **IFCSHAPEREPRESENTATION**(#50, 'Axis', 'GeometricCurveSet', (#1013));

#1015= **IFCSHAPEREPRESENTATION**(#51, 'Body', 'SweptSolid', (#1010));

#1016= **IFCPRODUCTDEFINITIONSHAPE**(\$, \$, (#1014, #1015));

#1017= **IFCCARTESIANPOINT**((138.968610, 28.737569, 4.300000));

#1018= **IFCDIRECTION**((0.00000, 0.00000, -4.30000));

#1019= **IFCDIRECTION**((0.00000, 0.00000, 0.00000));

#1020= **IFCDIRECTION**((1.00000, 0.00000, 0.00000));

#1021= **IFCAXIS2PLACEMENT3D**( #1017, #1018,#1020);

#1022= **IFCLOCALPLACEMENT**(\$, #1021);

Clasificación de columna o pilar a la sección con forma de HEB 220 anteriormente creada, indicando su material, la persona que lo ha realizado y la ubicación de éste mediante los ejes.

#1023= **IFCCOLUMN**('3SbpHVfKjCAvf1X8PNIm\_d', #2, 'Bar-0', \$, 'Acero', #1022, #1016, \$);

#1024= **IFCSTRUCTURALCURVEMEMBER**('3\_44nZSHTEmexi9dmiM5UG', #2, 'Bar-0', \$, \$, #1022, #1016, .RIGID\_JOINED\_MEMBER.);

#1025= **IFCRELASSOCIATESMATERIAL**('0\$qt5jbPX0veEoI8bi1F4', #2, \$, \$, (#1024,#1023), #100);

Indica la capa en la que se ha creado el edificio, y quien lo ha creado o modificado.

#200= **IFCBUILDINGSTOREY**('38vy5SYW91V9U2IVfPJfnZ', #2, 'porticos', \$, \$, #400, \$, \$, .ELEMENT., 0.0);

#400= **IFCLOCALPLACEMENT**(#31, #10);

#600= **IFCRELAGGREGATES**('1PowzF7hj9RfabV9UIV4Z0', #2, \$, \$, #30, (#200));

#1026= **IFCRELCONTAINEDINSPATIALSTRUCTURE**('14P3iXGRbA5QZDkspG9ov4', #2, \$, \$, (#1024,#1023), #200);

#1027= **IFCDIRECTION**((0.00000, 0.00000, 1.00000));

#1028= **IFCEXTRUDEDAREASOLID**(#1008, #10, #1027, 8.22000);

Definición de los dos puntos que definen la trayectoria de la viga, viendo que la diferencia entre ellos es la longitud que se pretende asignar a ésta.

#1029= **IFCCARTESIANPOINT**((0.000000, 0.000000, 0.000000));

#1030= **IFCCARTESIANPOINT**((0.000000, 0.000000, 8.220000));

Definición de la polilínea mediante la ubicación del punto inicial y final que hace referencia al eje que sigue la trayectoria de la viga .

#1031= **IFCPOLYLINE**((#1029, #1030));

#1032= **IFCSHAPEREPRESENTATION**(#50, 'Axis', 'GeometricCurveSet', (#1031));

#1033= **IFCSHAPEREPRESENTATION**(#51, 'Body', 'SweptSolid', (#1028));

#1034= **IFCPRODUCTDEFINITIONSHAPE**(\$, \$, (#1032, #1033));

#1035= **IFCCARTESIANPOINT**((138.968610, 20.627569, 4.000000));

#1036= **IFCDIRECTION**((0.00000, 8.22000, 0.00000));

#1037= **IFCDIRECTION**((-8.22000, 0.00000, 0.00000));

#1038= **IFCAXIS2PLACEMENT3D**(#1035, #1036, #1037);

#1039= **IFCLOCALPLACEMENT**(\$, #1038);

Clasificación de viga o elemento horizontal estructural a la sección con forma de IPE 600 anteriormente definida, indicando su material, la persona que lo ha realizado y la ubicación mediante los ejes.

```
#1040= IFCBEAM('2veAAzKWf3UBelw$1V_zoz', #2 , 'Bar-1', $, 'Acero', #1039, #1034, $);

#1041= IFCSTRUCTURALCURVEMEMBER('22EPdikHr9EAI4g_7UjqRM', #2 , 'Bar-1', $, $, #1039, #1034,
.RIGID_JOINED_MEMBER.);

#1042= IFCREASSOCIATESMATERIAL('22LEHUQwH6fLDnmOF3AkI', #2 , $, $, (#1041,#1040), #100);

#1043= IFCRELCONTAINEDINSPATIALSTRUCTURE('2nIRKScT1AfhIS4XnYMi1W', #2, $, $, (#1041,#1040),
#200);

#1044= IFCDIRECTION((0.00000, 0.00000, 1.00000));

#1045= IFCEXTRUDEDAREASOLID(#1004, #10, #1044, 4.3000);

#1046= IFCCARTESIANPOINT((0.000000, 0.000000, 0.000000));

#1047= IFCCARTESIANPOINT((0.000000, 0.000000, 4.300000));

#1048= IFCPOLYLINE((#1046, #1047));

#1049= IFCSHAPEREPRESENTATION(#50, 'Axis', 'GeometricCurveSet', (#1048));

#1050= IFCSHAPEREPRESENTATION(#51, 'Body', 'SweptSolid', (#1045));

#1051= IFCPRODUCTDEFINITIONSHAPE($, $, (#1049, #1050));

#1052= IFCCARTESIANPOINT((138.968610, 20.737569, 0.000000));

#1053= IFCDIRECTION((0.00000, 0.00000, 4.30000));

#1054= IFCDIRECTION((0.00000, 0.00000, 0.00000));

#1055= IFCDIRECTION((1.00000, 0.00000, 0.00000));

#1056= IFCAXIS2PLACEMENT3D( #1052, #1053,#1055);

#1057= IFCLocalPLACEMENT($, #1056);

#1058= IFCCOLUMN('0MgLS4xVH9mxEbnoV11KI$', #2 , 'Bar-2', $, 'Acero', #1057, #1051, $);

#1059= IFCSTRUCTURALCURVEMEMBER('10ZgeyCub0kB$D7vyIT_iK', #2 , 'Bar-2', $, $, #1057, #1051,
.RIGID_JOINED_MEMBER.);

#1060= IFCREASSOCIATESMATERIAL('0dizqjX9P2M8LV4Zy64RK_', #2 , $, $, (#1059,#1058), #100);

#1061= IFCRELCONTAINEDINSPATIALSTRUCTURE('2BAstnwi57Rw$O$_2J0Mlx', #2, $, $, (#1059,#1058),
#200);

ENDSEC;

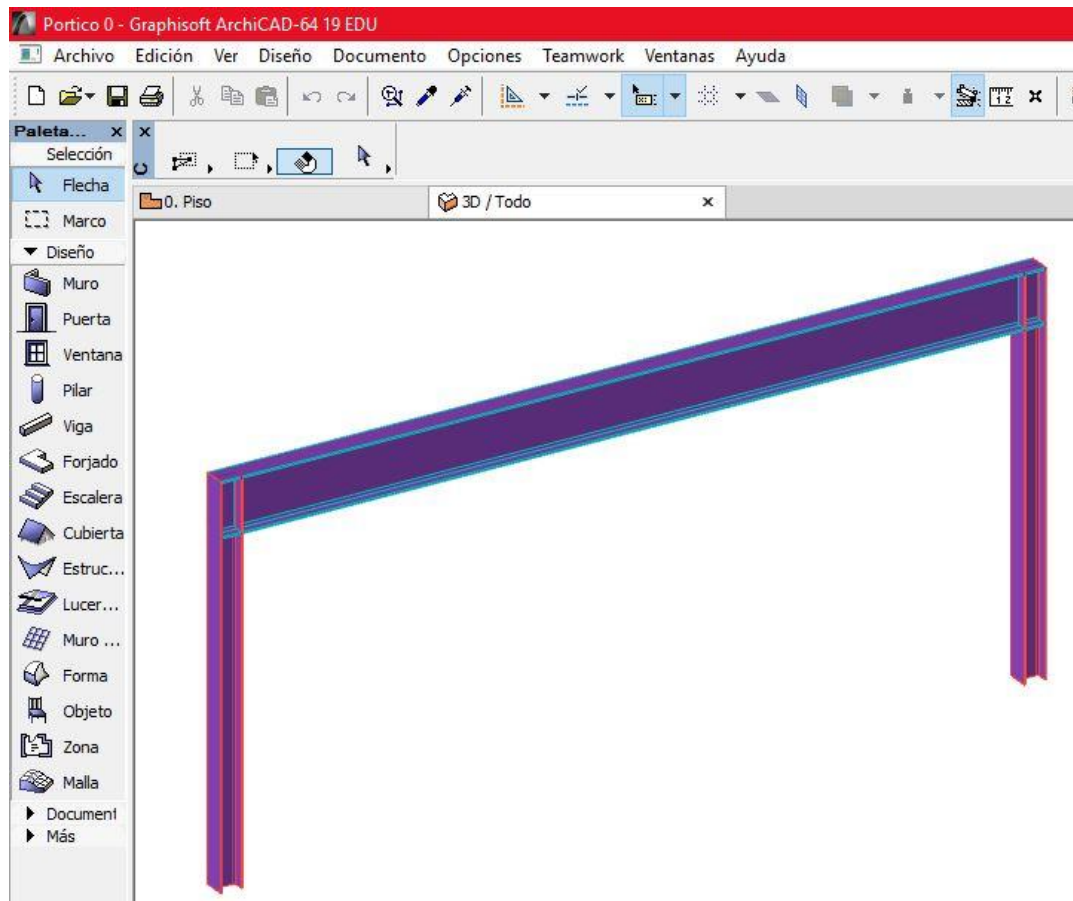
END-ISO-10303-21;
```

### 4.3.1.2. VISUALIZACIONES EN PROGRAMAS BIM.

El documento IFC obtenido permite la visualización del pórtico en diferentes aplicaciones nativas BIM, convirtiendo así al programa Architrave en un programa No Nativo BIM, pero sí intercambiable y conectable con aquellos que sí lo son.

#### a. Programa Nativo BIM: ArchiCAD.

Al abrir el archivo IFC creado, nos aparece la imagen siguiente, en la que comprobamos que la viga tiene un contorno de color distinto que el de los pilares, entendiéndose así que está estableciendo diferencias en las características de la viga y los pilares.



**Figura 4.12.**  
*Pórtico en ArchiCAD: volumetría. (elaboración propia)*



Para comprobar que las características importadas del IFC que ha establecido están correctas seleccionamos cada uno de ellos y podemos ver que en ambos casos coincide con las características que hemos establecido previamente en el archivo IFC.

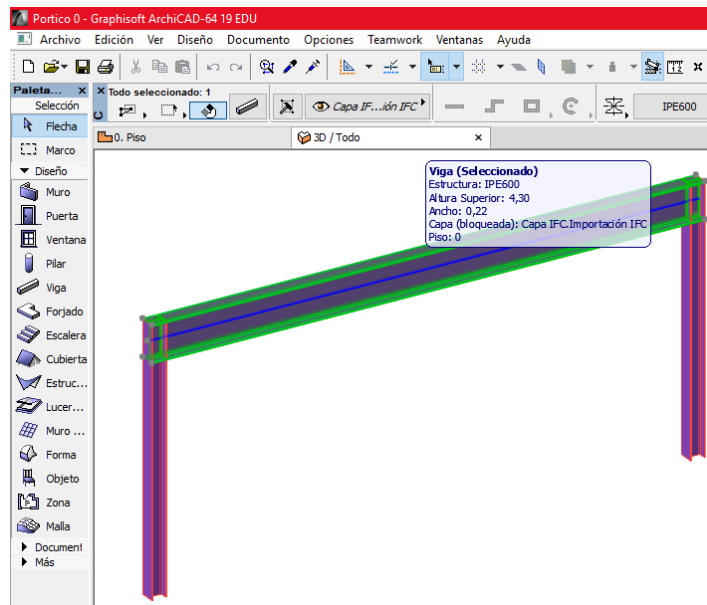
La viga tiene las mismas características definidas en el archivo IFC:

Tipología: **viga**, propiedad definida mediante la función *lfcBeam*.

Perfil utilizado: **IPE 600**, propiedad definida mediante la función *lfcIShapeProfileDef*,

Altura superior: **4,30m**, definidos mediante *lfcCartesianPoint*, *lfcDirection*, *lfcLocalPlacement*, *lfcAxis2Placement3D*.

Ancho: 0.22m, característica propia del perfil IPE 600.



**Figura 4.13.**  
Pórtico en ArchiCAD: Propiedades viga. (elaboración propia)

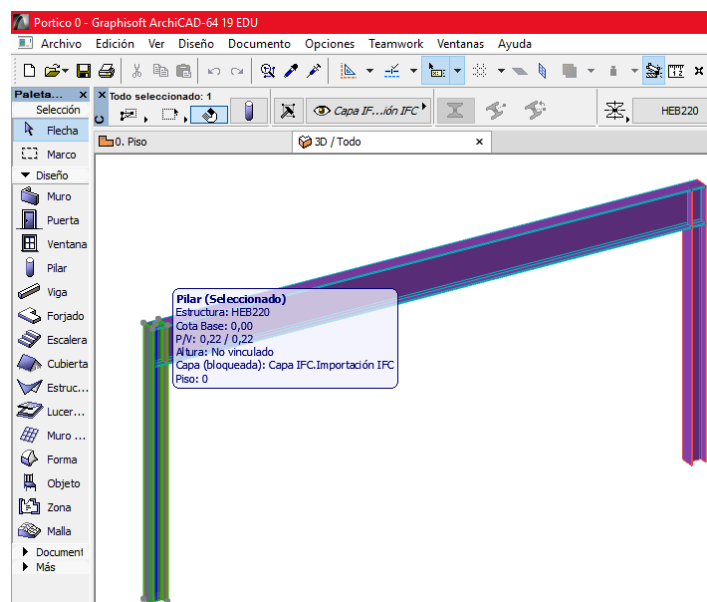
Los pilares también tienen las mismas características definidas en el archivo IFC:

Tipología: **pilar**, propiedad definida mediante la función *lfcColumn*.

Perfil utilizado: **HEB220**, propiedad definida mediante la función *lfcIShapeProfileDef*,

Altura superior: **4,30m**, definidos mediante *lfcCartesianPoint*, *lfcDirection*, *lfcLocalPlacement*, *lfcAxis2Placement3D*.

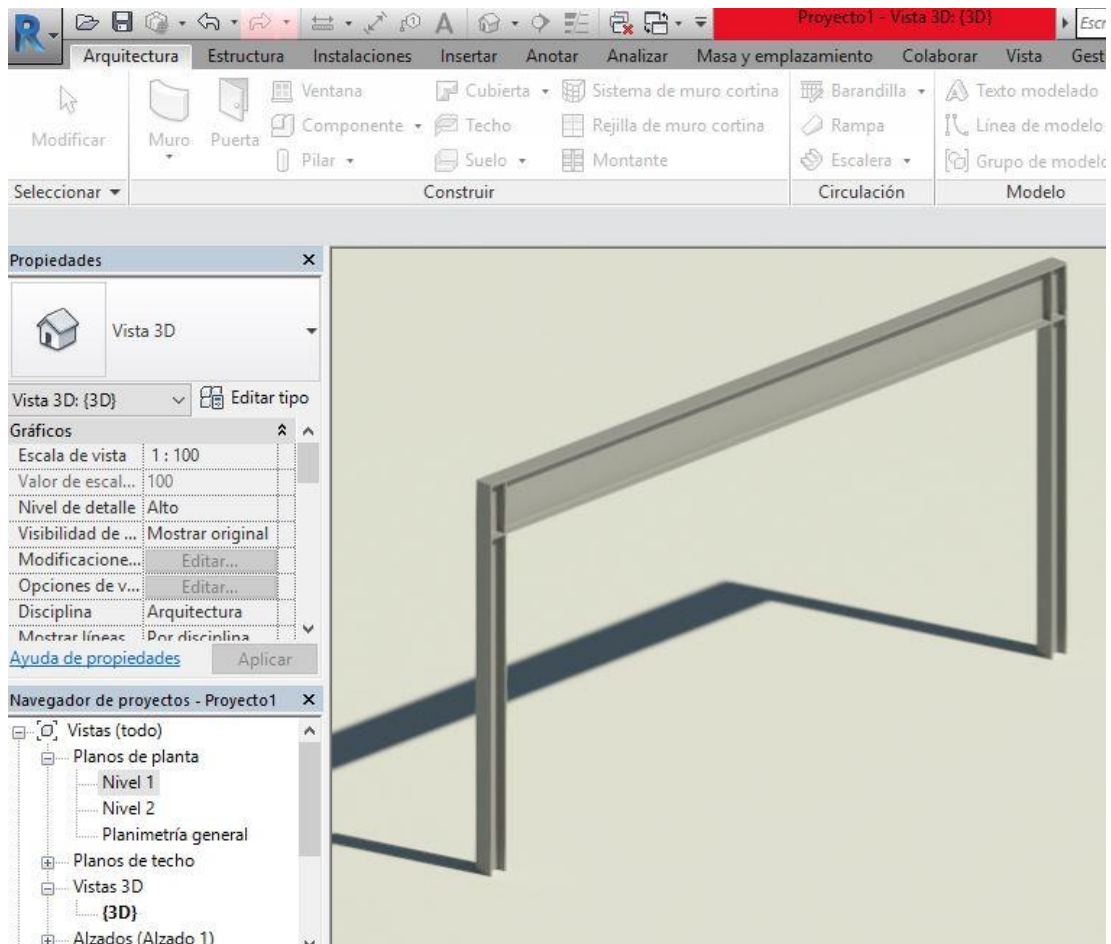
Bxh perfil: 0.22x0.22m, característica propia del perfil HEB220.



**Figura 4.14.**  
Pórtico en ArchiCAD: Propiedades pilar. (elaboración propia)

### b. Programa Nativo BIM: Revit.

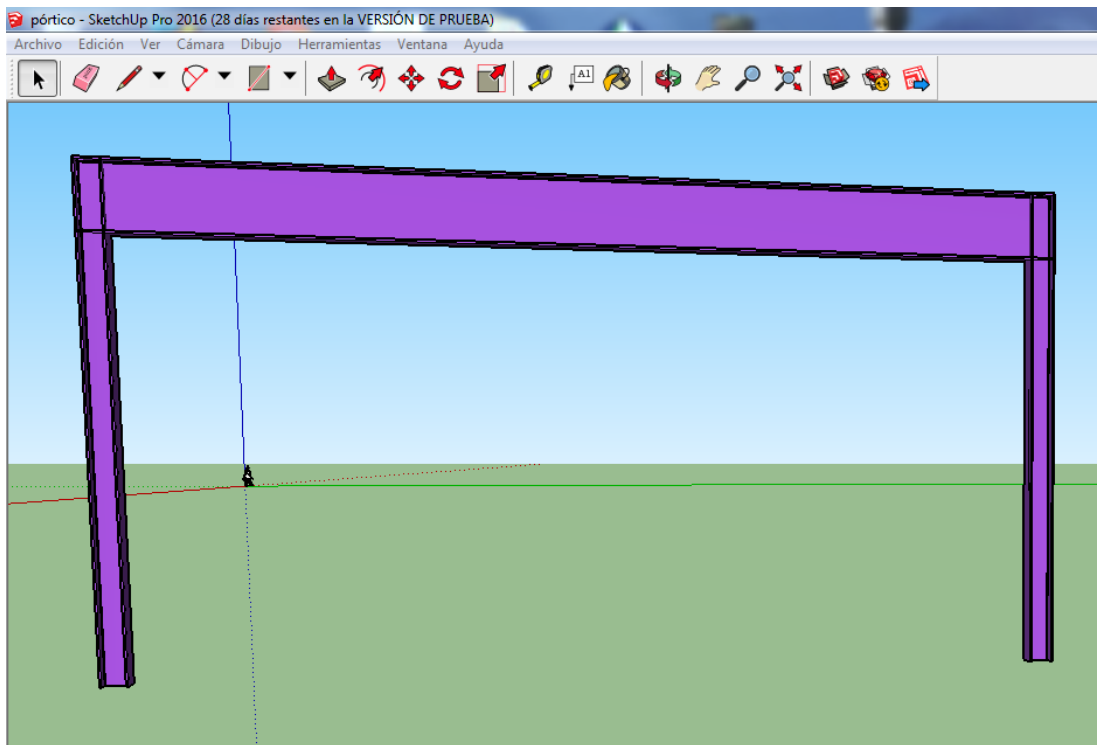
También en el caso del programa Nativo BIM, Revit comprobamos que se consigue importar y exportar el archivo IFC creado del pórtico.



**Figura 4.15.**  
Pórtico en Revit: volumetría renderizada (elaboración propia)

### c. Programa No Nativo BIM: SketchUp Pro.

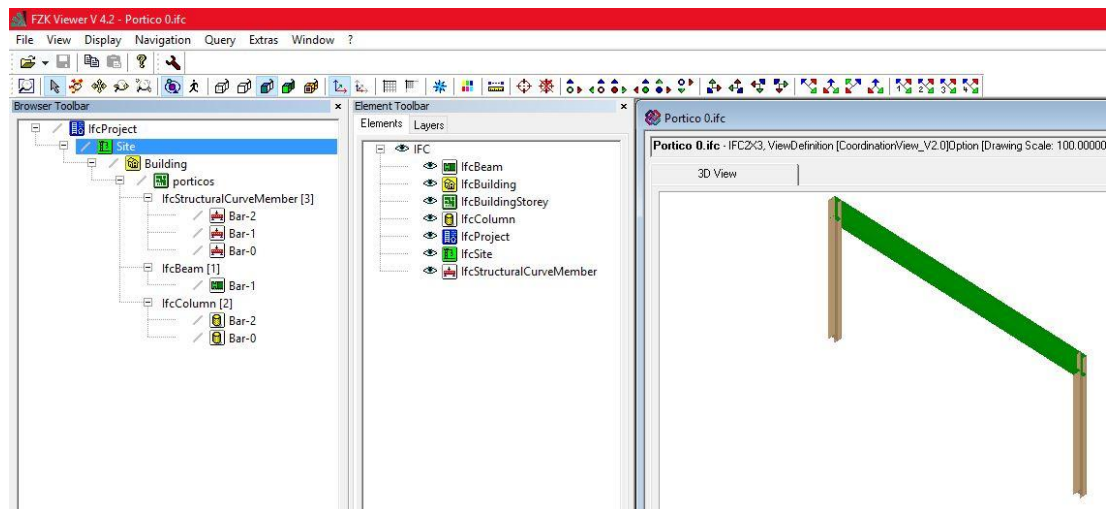
En el caso del programa no nativo BIM SketchUp Pro se tiene la opción de importar y exportar archivos IFC, y en este caso también comprobamos que se consigue leer adecuadamente el archivo generado anteriormente del pórtico. El inconveniente de esta tipología de importación (desde un programa no Nativo BIM) es que las propiedades de los IFC no se exportan todas, es decir, no reconoce qué tipología de perfiles ni el material que son, únicamente permite la visualización de la geometría de la figura que contiene dicho IFC y esto se consigue también en esta ocasión con éxito.



**Figura 4.16.**  
*Pórtico en SketchUp: volumetría (elaboración propia)*

#### d. Programa visualizador de IFCs gratuito en la web: FZK Viewer.

Se han realizado las visualizaciones también en este programa para demostrar que no es preciso tener una licencia concreta para poder visualizar aquello que contiene un IFC, es decir, los proyectos pueden visualizarse en cualquier situación y desde cualquier dispositivo si son convertidos a formato IFC.

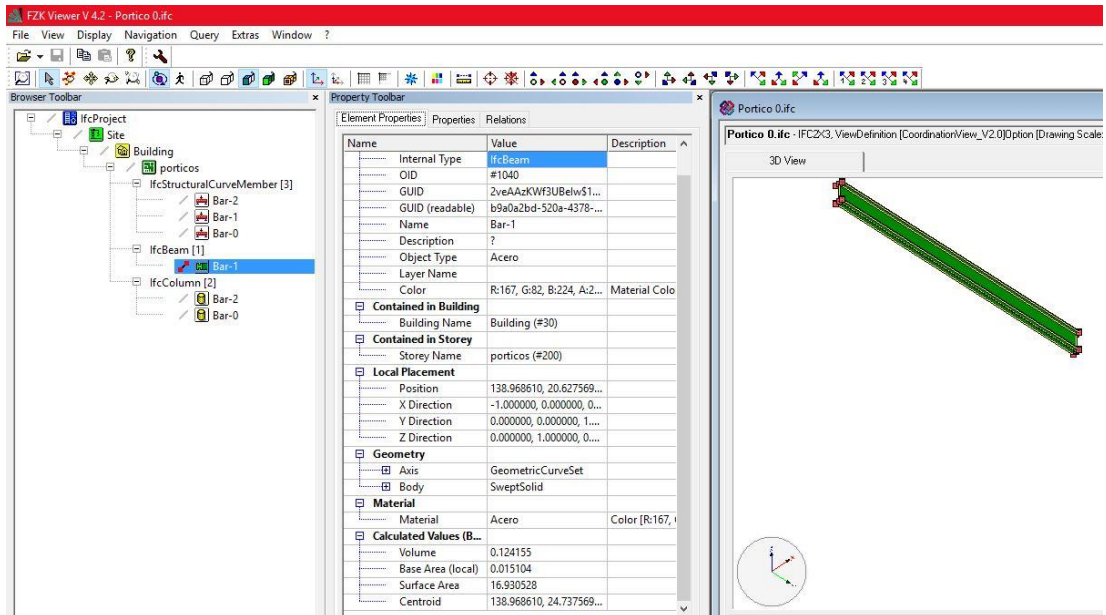


**Figura 4.17.**

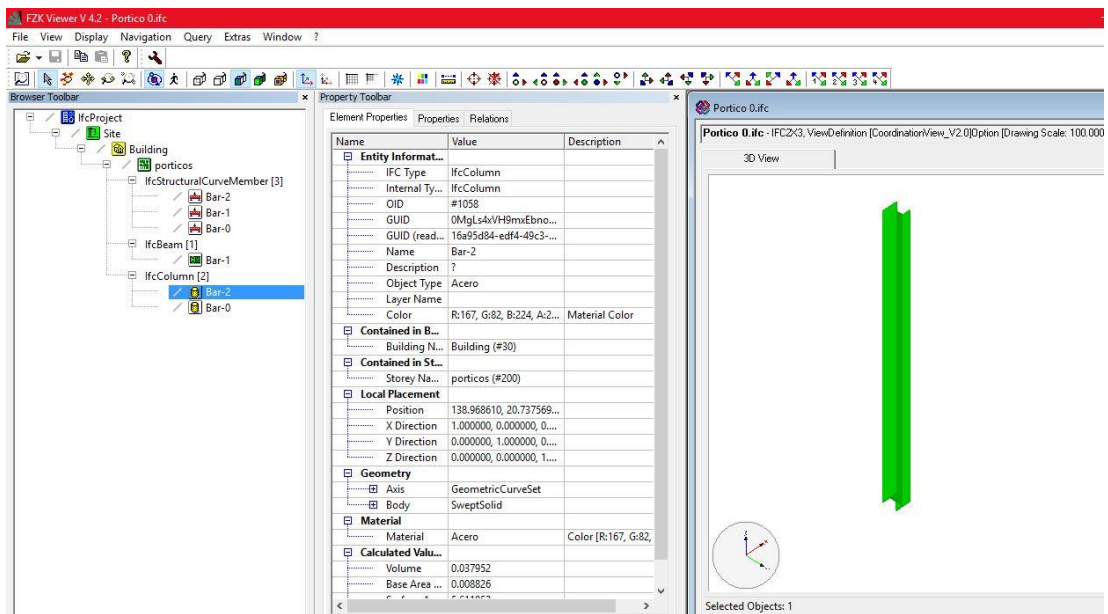
*Pórtico en FZKViewer: volumetría y funciones IFC utilizadas. (elaboración propia)*

En el caso del visualizador FZK, comprobamos que se visualiza toda la información que hemos añadido, de hecho, nos muestra un diagrama de las funciones IFC utilizadas en el proyecto completo; el número de barras utilizadas y la clasificación de éstas. En el diagrama podemos ver, que hay tres elementos que hacen referencia al `IfcStructuralCurveMember` refiriéndose a que son tres elementos estructurales, por otro lado, la clasificación correcta de la viga dentro de la función `IfcBeam` y de las dos columnas dentro de `IfcColumn`.

En las dos siguientes imágenes, se filtra la visualización según la barra seleccionada y se informa de las características acertadas de cada una de ellas también en esta ocasión.



**Figura 4.18.**  
Pórtico en FZKViewer: Propiedades viga. (elaboración propia)

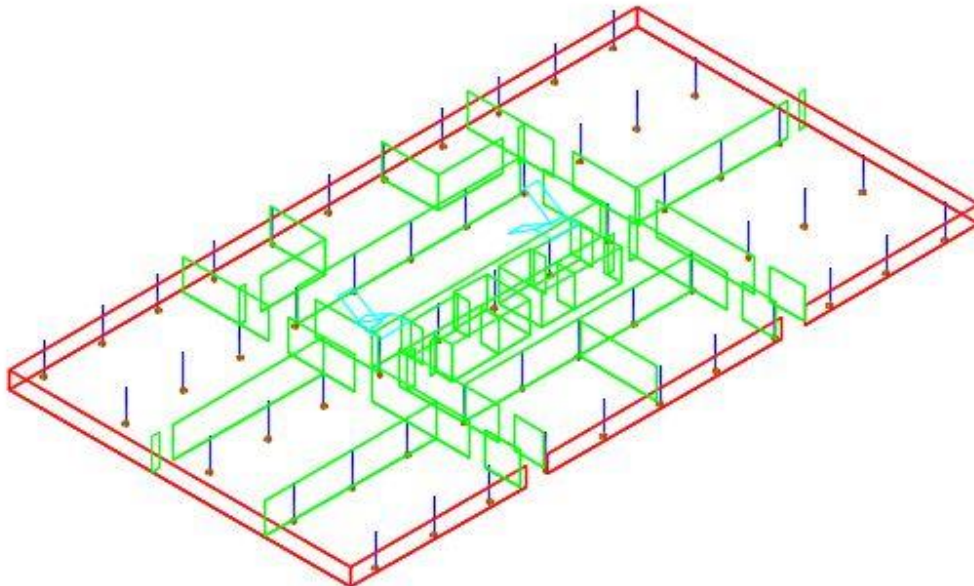


**Figura 4.19.**  
Pórtico en FZKViewer: Propiedades pilar. (elaboración propia)

### 4.3.2. CREACIÓN DEL CROWN HALL EN ARCHITRAVE

Seguidamente al pórtico, el cual era una manera sencilla de introducir el procedimiento y poder mostrar el archivo IFC completo, hemos querido comprobar el funcionamiento del protocolo creado con la representación de una estructura más compleja y popular como es la del “Crown Hall” del famoso arquitecto Mies Van der Rohe.

#### a. Planta semisótano.



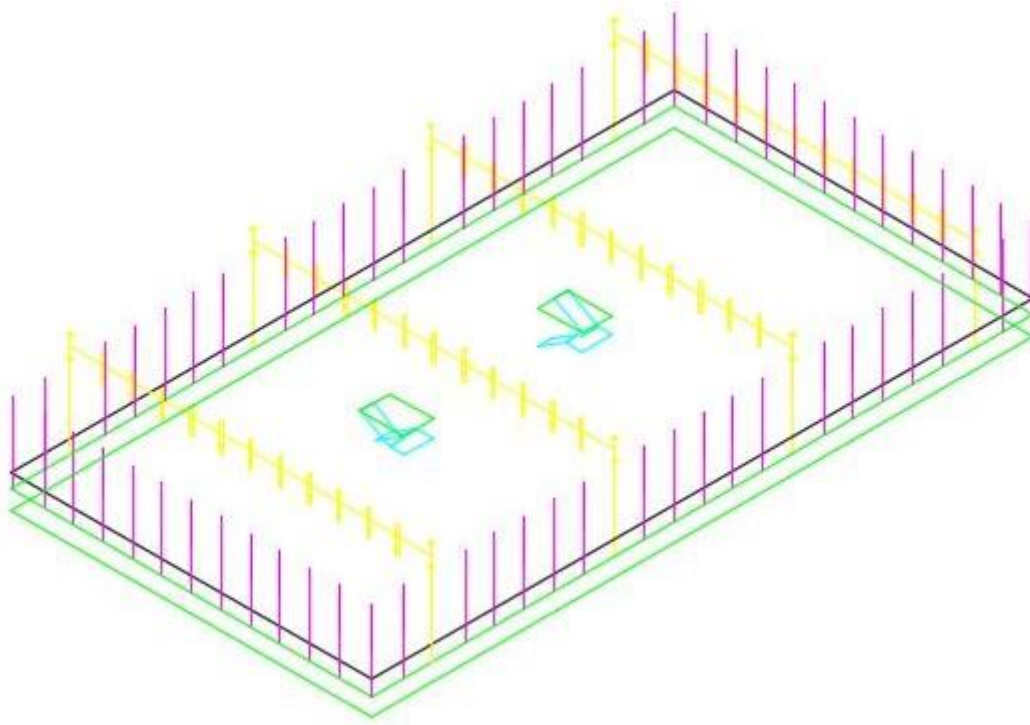
**Figura 4.20.**

*Crown Hall en Architrave: planta semisótano. (elaboración propia)*

Los datos de aquello que se ha dibujado en la planta semisótano coinciden con:






<span style="display: inline-block; width: 15px; height: 10px; background-color: red; margin-right: 5px;"></span>	Muro contenedor	HA -25	e=600mm
<span style="display: inline-block; width: 15px; height: 10px; background-color: green; margin-right: 5px;"></span>	Muros divisorios	HA -25	e=300mm
<span style="display: inline-block; width: 15px; height: 10px; background-color: blue; margin-right: 5px;"></span>	Pilares	HA -25	400x400mm
<span style="display: inline-block; width: 15px; height: 10px; background-color: cyan; margin-right: 5px;"></span>	Losa escalera	HA -25	e=150mm
	Losa base	HA -25	e=300mm

**b. Planta baja.**

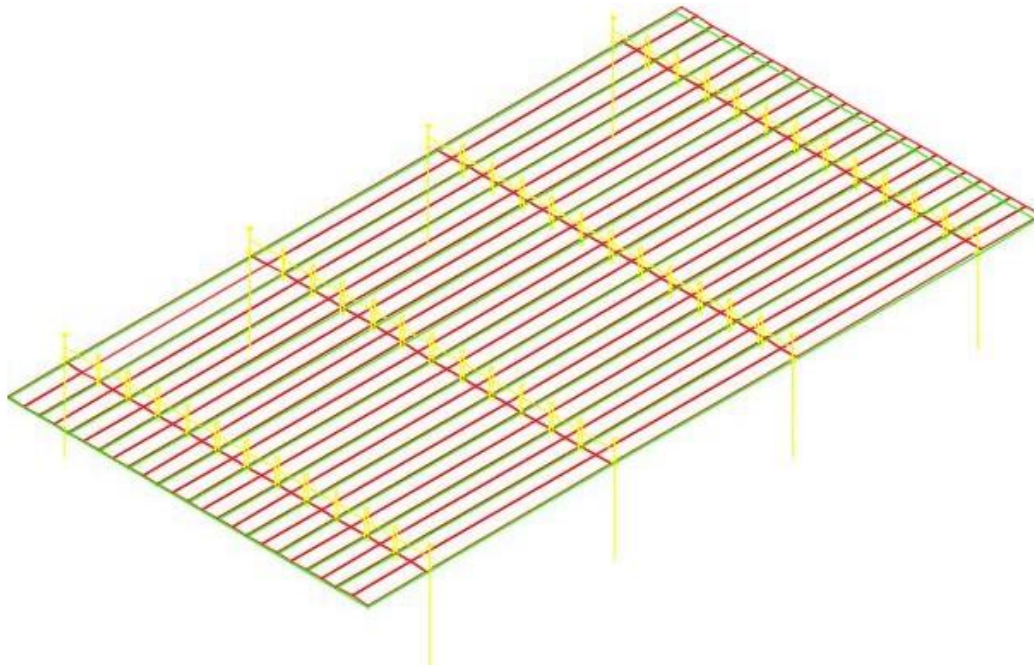


**Figura 4.21.**  
Crown Hall en Architrave: planta baja. (elaboración propia)

Los datos de aquello que se ha dibujado en la planta baja coinciden con:




	Perfiles	compuestos	
	Losa	HA -25	e=150mm
	Forjado	HA -25	e=300mm
	Losa escalera	HA -25	e=150mm
	Pórticos	Formado cada uno por:	
		Viga acero S275 2000x36420mm	e= 300mm
		Dos pilares S275	HEB 360
		Once refuerzos viga S275 300x2000mm	e= 25mm

c. Cubiertas.



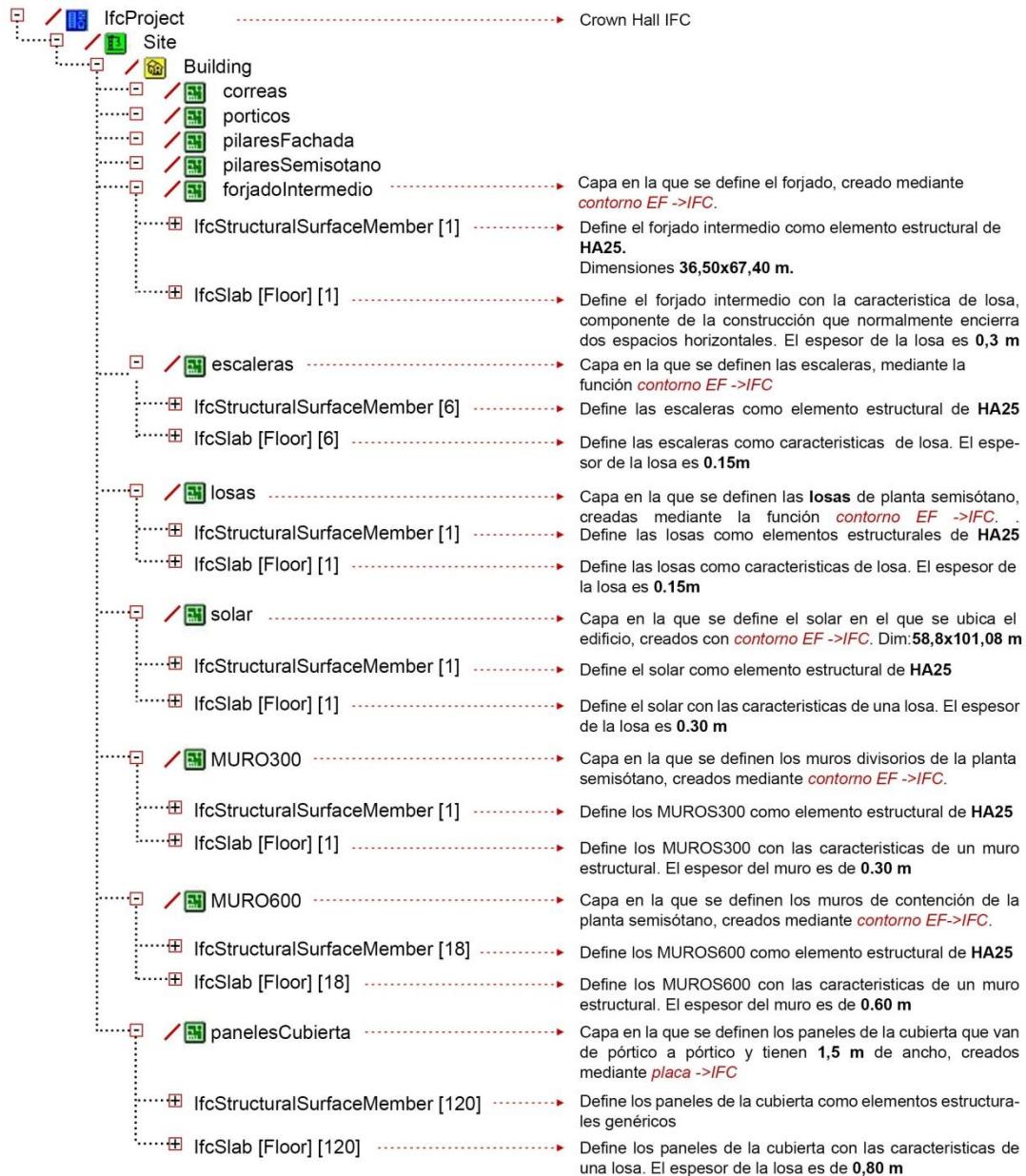
**Figura 4.22.**  
Crown Hall en Architrave: cubierta. (elaboración propia)

Los datos de aquello que se ha dibujado en la cubierta coinciden con:

	Paneles cubierta	losa	e= 80mm
	Correas	acero S275	IPE 240
	Pórticos	Formado cada uno por:	
		Viga acero S275 2000x36420mm	e= 300mm
		2 pilares S275	HEB 360
		11 refuerzos viga S275 300x2000mm	e= 25mm

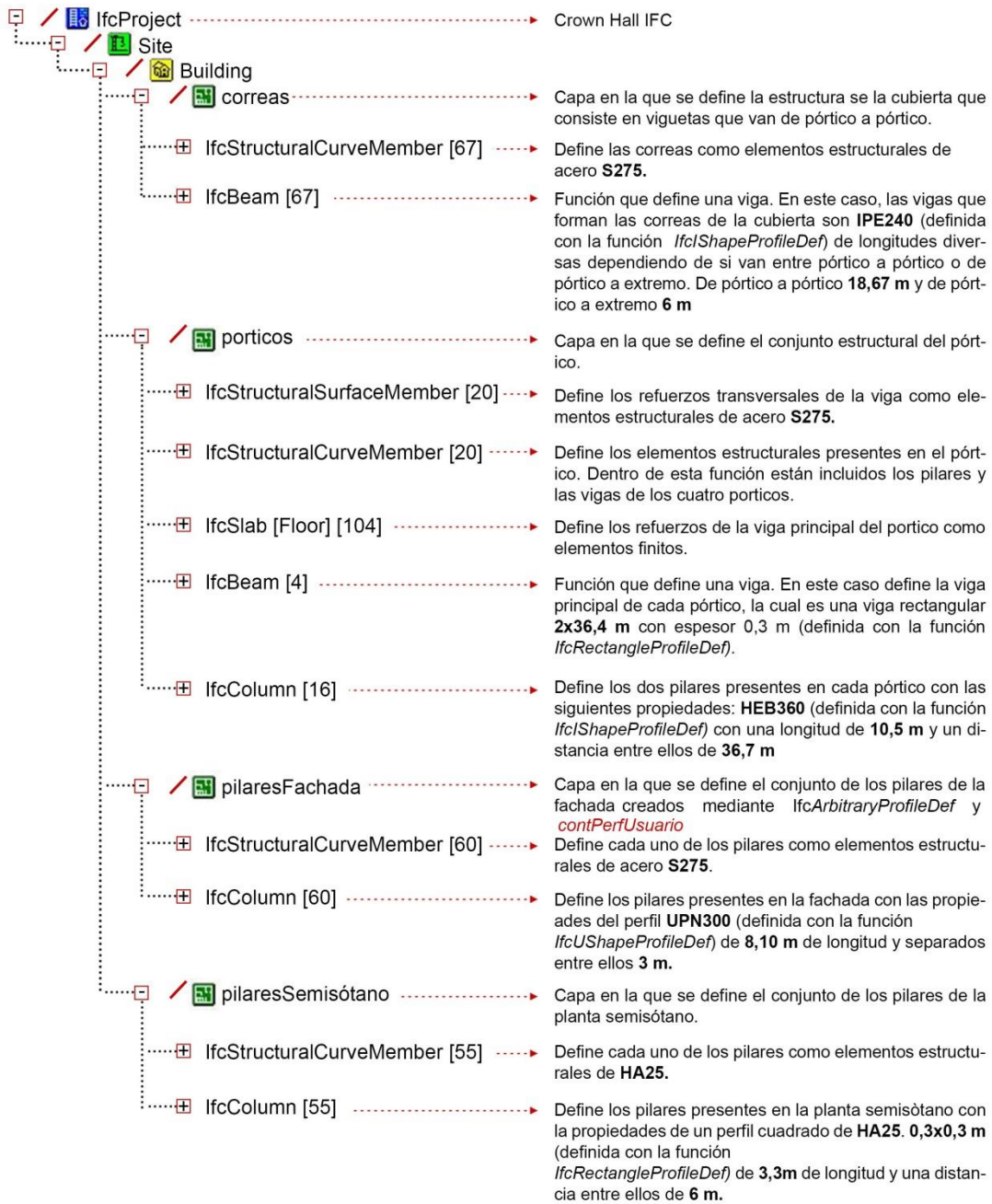


### 4.3.2.1. ANÁLISIS DEL DOCUMENTO IFC OBTENIDO



**Figura 4.23.**

Funciones IFC utilizadas en Crown Hall para elementos finitos, losas y forjados. (elaboración propia)



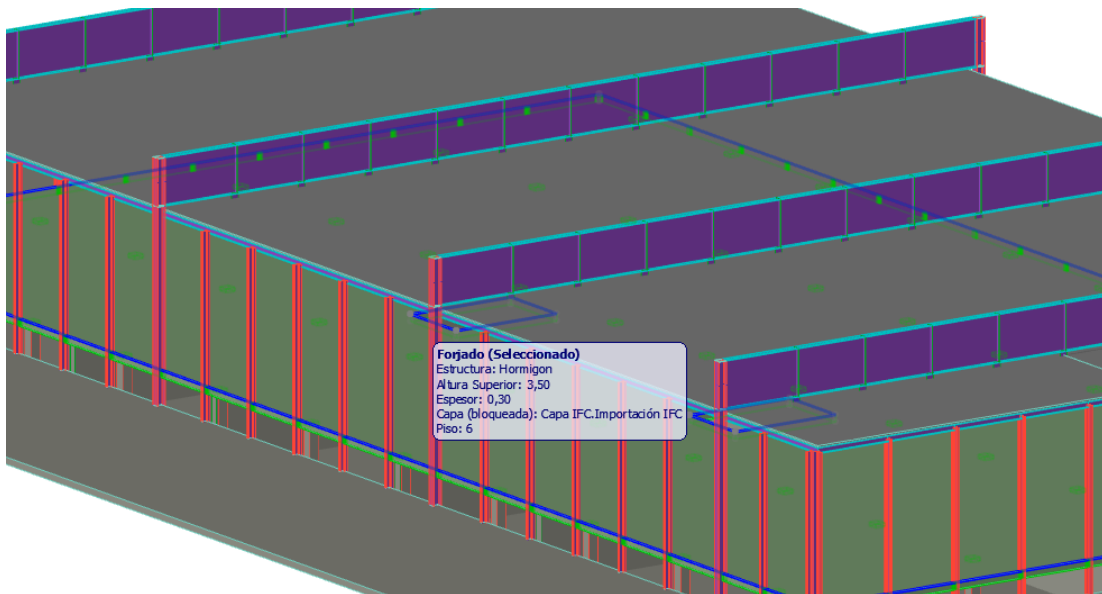
**Figura 4.24.**

Funciones IFC utilizadas en Crown Hall para barras. (elaboración propia)

### 4.3.2.2. VISUALIZACIONES EN PROGRAMAS BIM

En el presente apartado, se va a seguir el mismo procedimiento que en el del pórtico, mostrando una vez más la modalidad de importación dependiendo de la tipología de aplicación BIM: en este caso 2 Nativas BIM y el visor gratuito de archivos IFC.

#### a. Programa Nativo BIM: ArchiCAD.



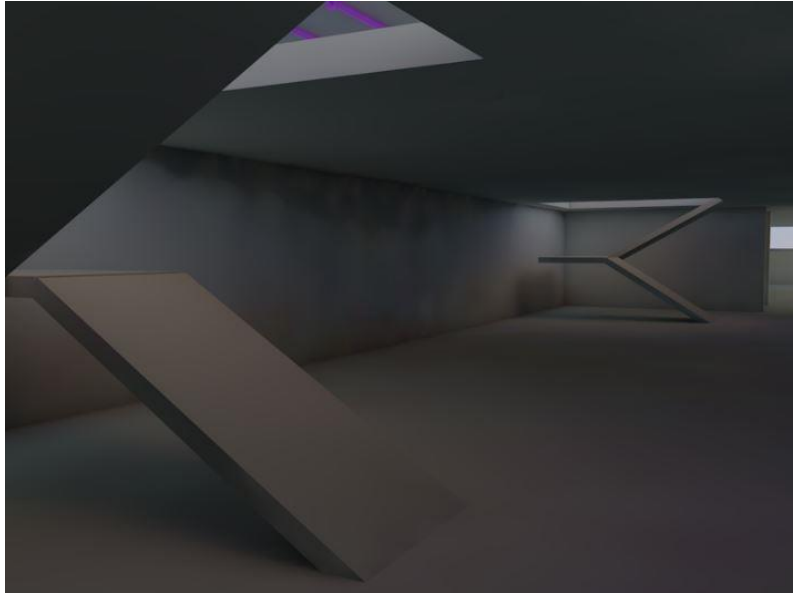
**Figura 4.25.**

*Crown Hall en ArchiCAD: forjado. (elaboración propia)*

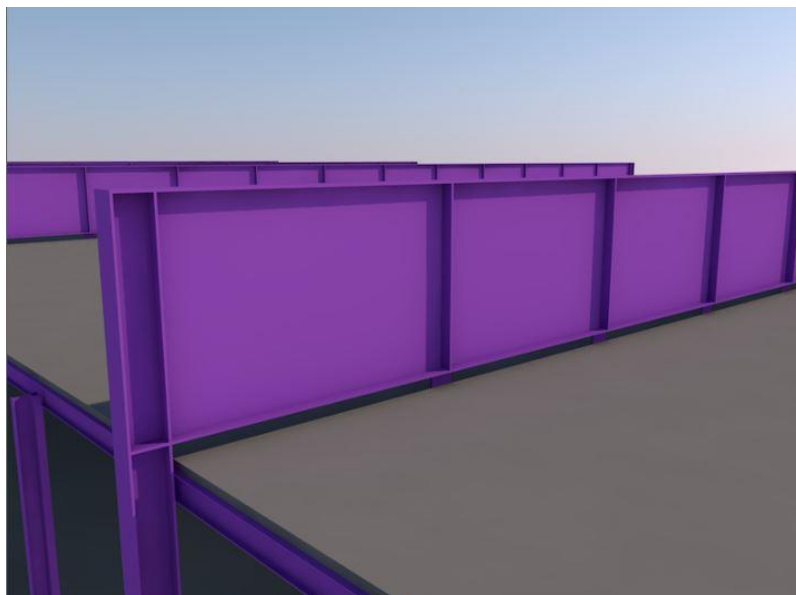


**Figura 4.26.**

*Crown Hall en ArchiCAD; vista exterior. (elaboración propia)*

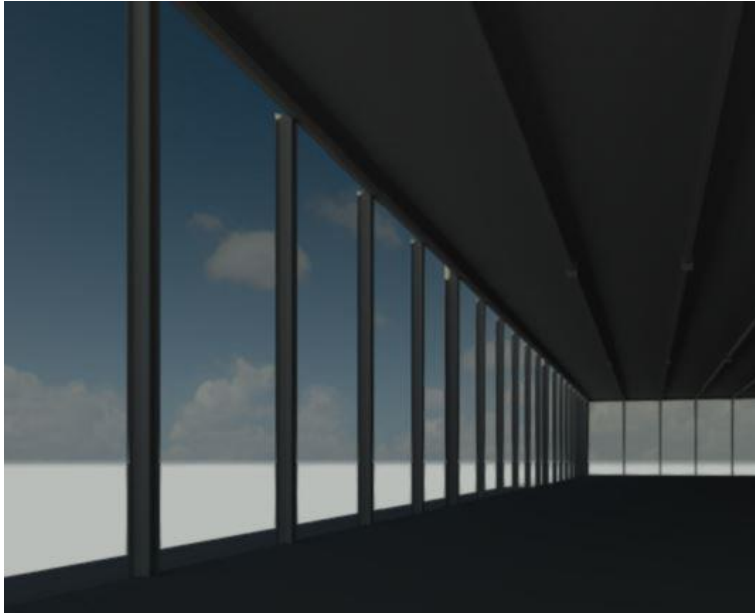


**Figura 4.27.**  
*Crown Hall en ArchiCAD; vista interior. (elaboración propia)*



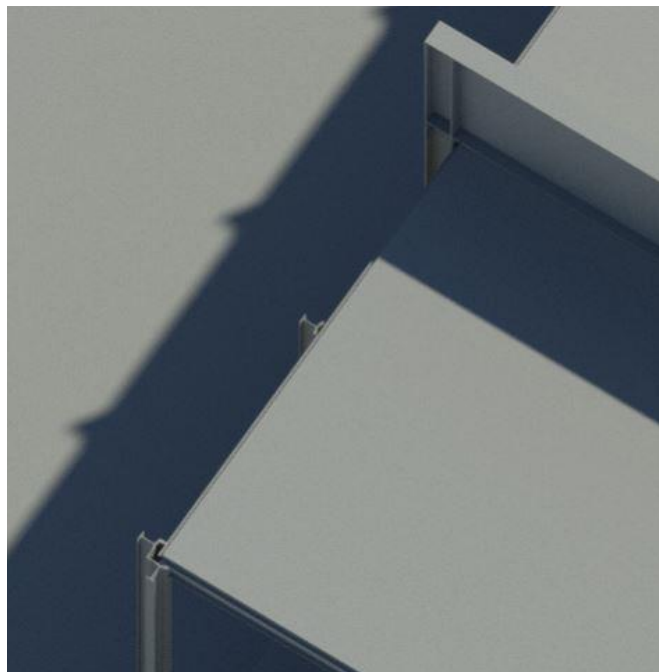
**Figura 4.28.**  
*Crown Hall en ArchiCAD; vista detalle refuerzos viga. (elaboración propia)*

**b. Programa Nativo BIM: Revit.**

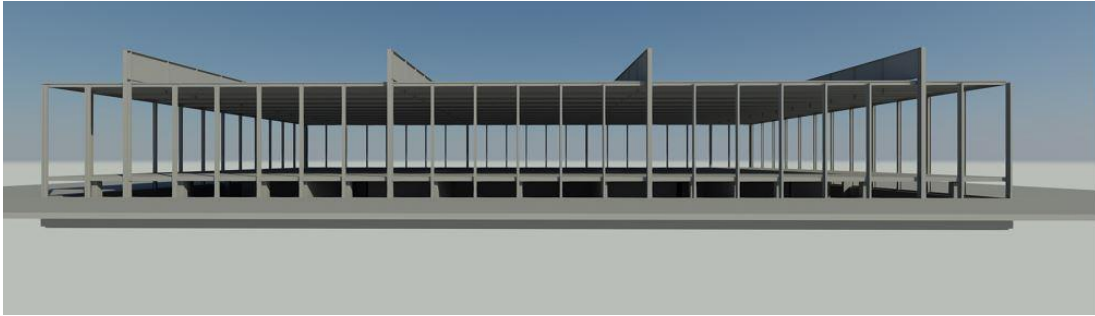


*La importación a Revit se ha realizado también de forma correcta. A continuación, algunos renders que lo demuestran.*

**Figura 4.29.**  
*Crown Hall en Revit; vista interior1. (elaboración propia)*



**Figura 4.30.**  
*Crown Hall en Revit; detalle perfil esquina. (elaboración propia)*



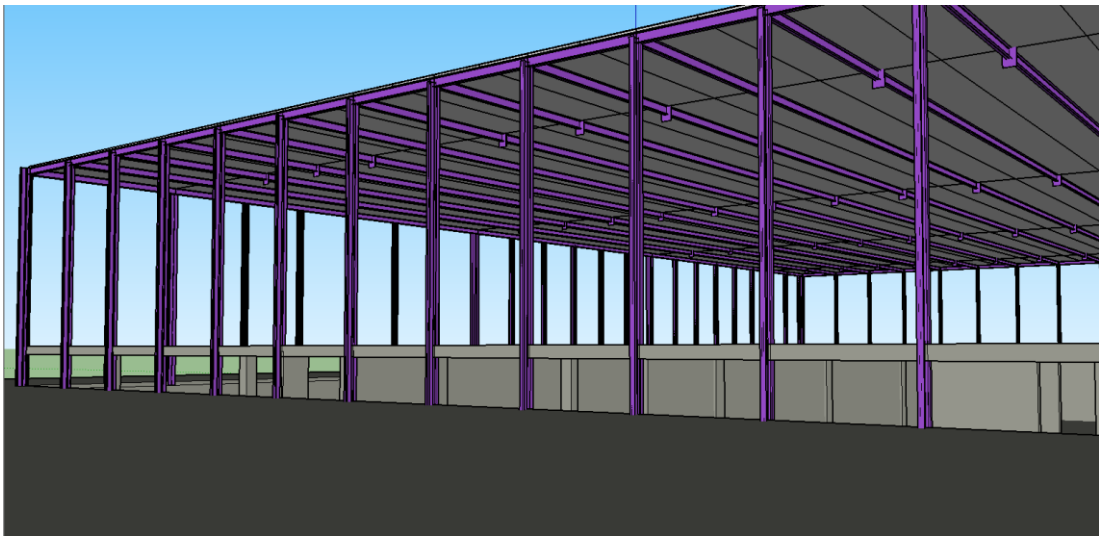
**Figura 4.31.**  
*Crown Hall en Revit; vista exterior. (elaboración propia)*



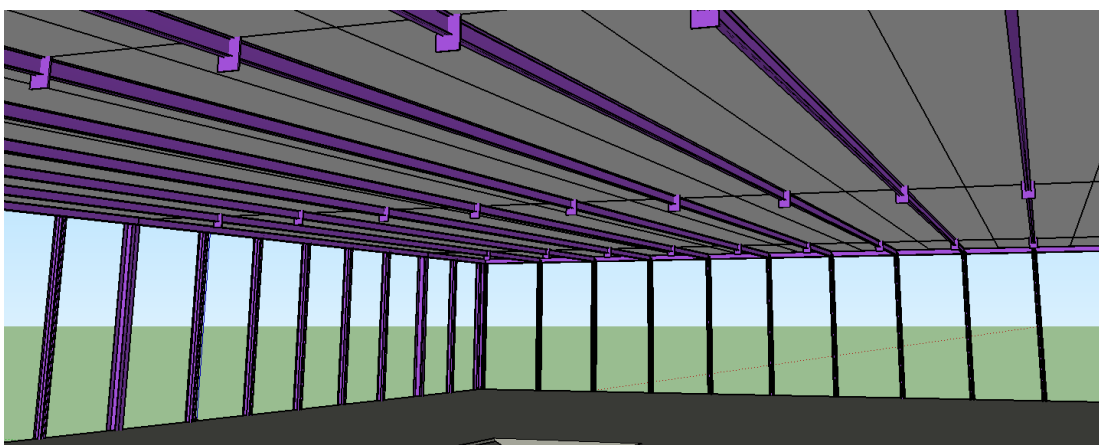
**Figura 4.32.**  
*Crown Hall en Revit; vista interior2. (elaboración propia)*

### c. Programa No Nativo BIM: SketchUp Pro.

SketchUp permite la importación y la exportación de forma acertada, aunque es importante que sea la versión Pro para que pueda abrir archivos IFC. Hemos realizado una vista exterior y otra interior para que se pueda apreciar que en general todas las características definidas se cumplen adecuadamente.



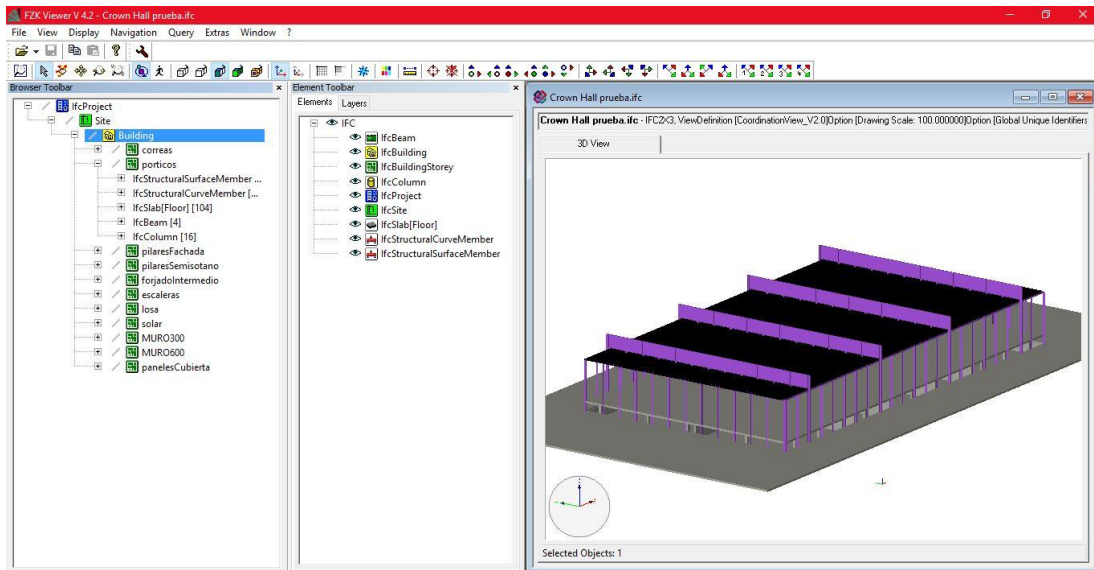
**Figura 4.33.**  
Crown Hall en SketchUp Pro: vista exterior. (elaboración propia)



**Figura 4.34.**  
Crown Hall en SketchUp Pro: vista interior. (elaboración propia)

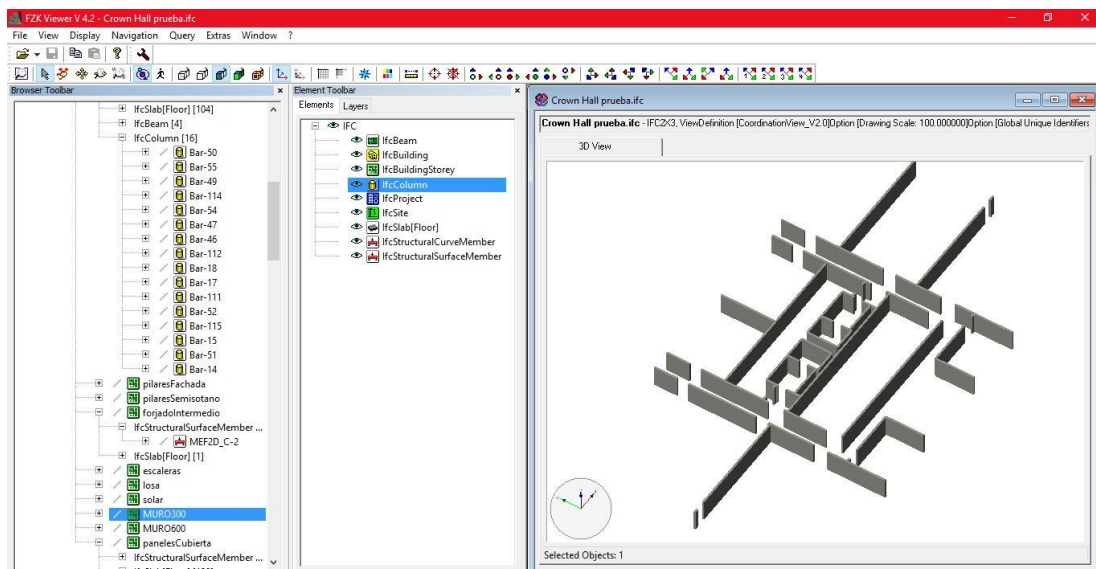
### d. Programa visualizador de IFCs gratuito en la web: FZK Viewer.

El visor FZK, también en el caso del proyecto “Crown Hall” nos proporciona una lista de todas las funciones IFC que definen los elementos estructurales de dicho proyecto, la cantidad y las propiedades de cada una de las barras y elementos finitos.



**Figura 4.35.**

*Crown Hall en FZK Viewer: axonometría. (elaboración propia)*



**Figura 4.36.**

*Crown Hall en FZK Viewer: muros planta semisótano. (elaboración propia)*





Durante el desarrollo del TFG, considero que se han ido cumpliendo los objetivos propuestos al inicio de éste. Por consiguiente, nos encontramos en condiciones de poder extraer algunas conclusiones del análisis de la metodología BIM y el formato IFC estudiados.

En primer lugar, al **estudiar la teoría de la metodología BIM**, podemos afirmar que es conveniente adoptarla en comparación con cualquier otra metodología habitual debido a que:

- La existencia de **un único contenedor de información del proyecto**, evita las posibles incoherencias que se puedan generar cuando hay múltiples documentos diferentes de un mismo modelo.
- La **previa visualización**, diseño y cálculo del proyecto permite conocer con anterioridad “problemas” que podrían darse a pie de obra, permitiendo así anticiparse a las tomas de decisiones. Esto por **consiguiente evita pérdidas de tiempo y económicas**.
- El modelo creado con un software BIM permite seguir su **trayectoria durante todo su ciclo de vida**: conservando así la información para cuando tengan que hacerse reformas o el edificio tenga que demolerse.
- Permite la **interdisciplinaridad**, mediante la nube en la que se encuentra el proyecto, debido a esto el archivo siempre estará actualizado gracias a que las modificaciones realizadas por sus agentes intervinientes se actualizan automáticamente.
- Es una metodología **multivista**, ya que permite todas las visualizaciones que se necesiten al mismo tiempo, lo que **ayuda al análisis y mejora del proyecto**.
- Como conclusión, el uso de esta metodología conviene adoptarla, ya que **incrementa la productividad** de cualquier trabajador o empresa, debido a todo aquello comentado con anterioridad.

Por otra parte, mediante la creación del protocolo de generación de elementos BIM hemos podido conocer y analizar el formato IFC que se utiliza para la creación de dichos elementos. Por lo tanto, puedo afirmar que:

- El formato IFC permite el **intercambio de información entre múltiples software nativos y no nativos BIM**. Característica ventajosa, debido a que permite evitar las interferencias que podría ocasionar el tener esta información en diversos documentos.
- Al analizar el archivo IFC, nos damos cuenta que toda la **información está organizada jerárquicamente** e incluso a través de él podemos conocer información importante que desde el visualizador del 3D no nos permite conocer.
- El **archivo IFC pesa un 20% menos** que un archivo generado por cualquier Software BIM nativo.
- **Nunca se desfasa** debido a que es un **formato de texto**, por lo tanto aunque la versión del programa en el cual se ha creado se actualice, no correremos el riesgo de no poder acceder a dicha información.
- Permite conocer **la información basada en** los cuatro siguientes ámbitos:
  - o La **geometría** del modelo.
  - o La **relación** que guarda el modelo con otros modelos del edificio.
  - o Las **propiedades** inherentes del modelo.
  - o La **posibilidad de adicionar metadatos** a lo largo del ciclo de vida del edificio.
- Hemos podido comprobar mediante la realización del presente trabajo que **cualquier fabricante podría hacer compatible su software con el formato IFC** y beneficiarse de sus múltiples ventajas.

En cuanto a las *líneas de desarrollo futuras* de este trabajo;

Al finalizar este TFG, hemos conseguido explicar el procedimiento de creación de un protocolo que permita exportar documentos IFC. En un futuro, mediante la correspondiente investigación, se podría conseguir el proceso inverso: construir el modelo de la estructura en Revit, ArchiCAD, SketchUp... o cualquier programa BIM, para generar un IFC que contenga los ejes de las barras, los contornos de los muros y las losas, la información de los tamaños de las secciones, los espesores que se precisen y los materiales. El importador de Architrave interpretaría toda la información y se podría entonces calcular la estructura del modelo sin interferencias.

Por último, añadir que el objetivo personal que suelo ponerme en cualquiera de mis proyectos, también he conseguido cumplirlo; el de *aprender*.

La realización del presente trabajo me ha ayudado a entender, desde las bases, el procedimiento de funcionamiento de la metodología BIM, con la que he podido experimentar todas sus características y los numerosos beneficios que puede aportarnos.



<b>Figura 1.1.</b>	Metodología operativa (elaboración propia).	19
<b>Figura 2.1.</b>	Hoja de Ruta en España ( <a href="http://www.esbim.es">www.esbim.es</a> ).	21
<b>Figura 2.2.</b>	Logo ESBIM ( <a href="http://www.esbim.es">www.esbim.es</a> ).	22
<b>Figura 2.3.</b>	Implantación del BIM a nivel mundial ( <a href="http://blog.aulatematica.com/">http://blog.aulatematica.com/</a> ).	24
<b>Figura 3.1.</b>	Interoperabilidad BIM ( <a href="http://biblus.acca.it/ifc-cose-e-a-che-serve/">http://biblus.acca.it/ifc-cose-e-a-che-serve/</a> ).	27
<b>Figura 3.2.</b>	Cronología BIM (elaboración propia).	28
<b>Figura 3.3.</b>	Palabras que engloban la definición de BIM. ( <a href="http://maproinco.blogspot.com.es/">http://maproinco.blogspot.com.es/</a> )	31
<b>Figura 3.4.</b>	Representación Level of Detail (practical BIM.net).	34
<b>Figura 3.5.</b>	Representación Level of Development (practical BIM.net).	35
<b>Figura 3.6.</b>	Proyecto propio que muestra los diversos niveles de detalle LOD (elaboración propia).	37
<b>Figura 3.7.</b>	Dimensiones BIM ( <a href="http://www.bimbarcelona.com/bim-es-el-presente-no-el-futuro/">http://www.bimbarcelona.com/bim-es-el-presente-no-el-futuro/</a> ).	39
<b>Figura 3.8.</b>	Beneficios del BIM (elaboración propia).	39
<b>Figura 3.9.</b>	BIM Multivista (elaboración propia de un proyecto en Revit)	40
<b>Figura 3.10.</b>	Productividad del diseño durante la implementación del BIM (elaboración propia).	42
<b>Figura 3.11.</b>	Familias de objetos creadas que ofrece Revit. (elaboración propia).	43
<b>Figura 3.12.</b>	Familias de pilares estructurales de acero ya creadas que ofrece Revit. (elaboración propia).	43
<b>Figura 3.13.</b>	Todos los flujos de información llegan al mismo contenedor. (elaboración propia).	44
<b>Figura 3.14.</b>	Dificultades BIM. (elaboración propia).	45
<b>Figura 3.15.</b>	Estándares básicos. ( <a href="http://buildingsmart.org/standards/">http://buildingsmart.org/standards/</a> ).	46
<b>Figura 3.16.</b>	Estructura jerárquica de archivos IFC: funciones de elementos constructivos. (elaboración propia).	47

<b>Figura 3.17.</b>	Logo IFC: reflejo de la interoperabilidad. ( <a href="http://buildingsmart.es">http://buildingsmart.es</a> ).	49
<b>Figura 3.18.</b>	Cronología del desarrollo del IFC ( <a href="http://buildingsmart.es">http://buildingsmart.es</a> ).	50
<b>Figura 3.19.</b>	Programas Nativos BIM y no Nativos BIM. (elaboración propia).	51
<b>Figura 3.20.</b>	Logos aplicaciones Nativas BIM. (páginas oficiales de cada una de ellas)	52
<b>Figura 3.21.</b>	Logos aplicaciones No Nativas BIM. (páginas oficiales de cada una de ellas).	53
<b>Figura 4.1.</b>	Ventana con la consola de Visual LISP (elaboración propia)	59
<b>Figura 4.2.</b>	Clasificación de las funciones IFC en la página oficial. ( <a href="http://www.buildingsmart-tech.org/ifc/IFC2x3/TC1/html/">http://www.buildingsmart-tech.org/ifc/IFC2x3/TC1/html/</a> ).	62
<b>Figura 4.3.</b>	Esquema explicativo de funcionamiento de la función IfcIShapeProfileDef. (elaboración propia).	79
<b>Figura 4.4.</b>	Esquema explicativo de funcionamiento de la función IfcUShapeProfileDef. (elaboración propia)	82
<b>Figura 4.5.</b>	Esquema explicativo de funcionamiento de la función IfcRectangleHollowProfileDef. (elaboración propia).	83
<b>Figura 4.6.</b>	Esquema explicativo de funcionamiento de la función IfcCircleHollowProfileDef. (elaboración propia).	91
<b>Figura 4.7.</b>	Esquema explicativo de funcionamiento de la función IfcRectangleProfileDef. (elaboración propia).	92
<b>Figura 4.8.</b>	Esquema explicativo de funcionamiento de la función IfcCircleProfileDef. (elaboración propia).	93
<b>Figura 4.9.</b>	Ejes del pórtico creado. (elaboración propia).	94
<b>Figura 4.10.</b>	Volumetría y detalle esquina pórtico. (elaboración propia).	95
<b>Figura 4.11.</b>	Funciones IFC utilizadas para la creación del pórtico. (elaboración propia).	96

<b>Figura 4.12.</b>	Pórtico en ArchiCAD: volumetría. (elaboración propia)	104
<b>Figura 4.13.</b>	Pórtico en ArchiCAD: Propiedades viga. (elaboración propia)	105
<b>Figura 4.14.</b>	Pórtico en ArchiCAD: Propiedades pilar. (elaboración propia)	105
<b>Figura 4.15.</b>	Pórtico en Revit: volumetría renderizada. (elaboración propia)	106
<b>Figura 4.16.</b>	Pórtico en SketchUp: volumetría. (elaboración propia)	107
<b>Figura 4.17.</b>	Pórtico en FZKViewer: volumetría y funciones IFC utilizadas. (elaboración propia)	108
<b>Figura 4.18.</b>	Pórtico en FZKViewer: Propiedades viga. (elaboración propia)	109
<b>Figura 4.19.</b>	Pórtico en FZKViewer: Propiedades pilar. (elaboración propia)	109
<b>Figura 4.20.</b>	Crown Hall en Architrave: planta semisótano. (elaboración propia)	110
<b>Figura 4.21.</b>	Crown Hall en Architrave: planta baja. (elaboración propia)	111
<b>Figura 4.22.</b>	Crown Hall en Architrave: cubierta. (elaboración propia)	112
<b>Figura 4.23.</b>	Funciones IFC utilizadas en Crown Hall para elementos finitos. (elaboración propia)	113
<b>Figura 4.24.</b>	Funciones IFC utilizadas en Crown Hall para barras. (elaboración propia)	114
<b>Figura 4.25.</b>	Crown Hall en ArchiCAD: forjado. (elaboración propia)	115
<b>Figura 4.26.</b>	Crown Hall en ArchiCAD: vista exterior. (elaboración propia)	115
<b>Figura 4.27.</b>	Crown Hall en ArchiCAD; vista interior. (elaboración propia)	116



<b>Figura 4.28.</b>	Crown Hall en ArchiCAD: vista detalle refuerzos viga. (elaboración propia)	116
<b>Figura 4.29.</b>	Crown Hall en Revit: vista interior. (elaboración propia)	117
<b>Figura 4.30.</b>	Crown Hall en Revit; detalle perfil esquina. (elaboración propia)	117
<b>Figura 4.31.</b>	Crown Hall en Revit: vista exterior1. (elaboración propia)	118
<b>Figura 4.32.</b>	Crown Hall en Revit; vista interior2. (elaboración propia)	118
<b>Figura 4.33.</b>	Crown Hall en SketchUp Pro: vista exterior. (elaboración propia)	119
<b>Figura 4.34.</b>	Crown Hall en SketchUp Pro: vista interior. (elaboración propia)	119
<b>Figura 4.35.</b>	Crown Hall en FZK Viewer: axonometría. (elaboración propia)	120
<b>Figura 4.36.</b>	Crown Hall en FZK Viewer: muros planta semisótano. (elaboración propia)	120



- ALONSO MADRID, J. 2013.** Revista: Nivel de desarrollo LOD. Definiciones, innovaciones y adaptación a España. Disponible en: [http://escuelaedificacion.org/images/zoo/uploads/PDF/PRESENCIALES/MBIM/Revista\\_Building\\_Smart\\_JAM\\_Art.pdf](http://escuelaedificacion.org/images/zoo/uploads/PDF/PRESENCIALES/MBIM/Revista_Building_Smart_JAM_Art.pdf). (Julio 2016)
- ARCHITRAVE.** Manual de Diseño. Disponible en: <http://www.architrave.es/> (Junio 2016)
- AUTODESK AUTOCAD 2016 /AYUDA.** Búsqueda de definiciones de funciones LISP ([www.help.autodesk.com](http://www.help.autodesk.com)). (Mayo 2016)
- BIMFORUM.** The US chapter of BuildingSMART International. Level of Development Specification. 2015. Disponible en: <http://bimforum.org/lod> (Junio 2016)
- BLOG DE AULATEMÁTICA, 2016.** La situación del BIM en España. Presente y futuro de la Arquitectura, edificación y construcción. Disponible en: <http://blog.aulatematica.com/situacion-bim-en-espana/> (Abril 2016)
- BUILDINGSMART SPANISH CHAPTER.** uBIM. Guía de usuarios. / Análisis energético. Disponible en: <http://www.buildingsmart.es> (6 de mayo 2016)
- BUILDINGSMART.** Modeling Support Group IFC 2x Edition 2. Disponible en: <http://www.buildingsmart-tech.org/downloads/accompanying-documents/guidelines/IFC2x%20Model%20Implementation%20Guide%20V2-0b.pdf>. (Junio 2016)
- COLOMA, E., 2008.** Introducción a la tecnología BIM. 1ª edn. Barcelona: Universitat Politècnica de Catalunya – Departament d'Expressió Gràfica Arquitectónica I. Disponible en: <http://upcommons.upc.edu/bitstream/handle/2117/12226/Introducci%F3n+a+la+Tecnolog%EDa+BIM.pdf;jsessionid=D84FBC0B1A840689F305AB3626C52B9E?sequence=1>. (Febrero 2016)
- ESBIM.** Implantación del BIM en España. Decálogo: Plan de acción. Disponible en: <http://www.esbim.es/> (Abril 2016)
- PÉREZ GARCÍA, A. y GUARDIOLA VILLORA, A.** Prontuario y herramientas informáticas para cálculo de estructuras. (Julio 2016)
- FUENTES GINER, B. 2014.** Impacto de BIM en el proceso constructivo español. Cuadernos EUBIM.

**INFOGRAPH.** InfoCAD – IFC Data Transfer. Disponible en: [http://download.infograph.de/en/infocad\\_ifc\\_manual.pdf](http://download.infograph.de/en/infocad_ifc_manual.pdf). (Mayo 2016)

**ITC, INSTITUTO TECNOLÓGICO DEL CANTÁBRICO.** Expertos en consultoría de formación y gestión de proyectos. Informe: Qué es el BIM, obligatorio en Europa, implantación en España, y competitividad. Disponible en: <http://itcformacionyconsultoria.com/bim-espana-europa/>. (Abril 2016)

**KAREN M. KENSEK I DOUGLAS E. NOBLE.** Building Information Modeling in current and future practice.

**KYMMEL, W., 2008.** Building Information Modeling: Planning and managing construction projects with 4D CAD and simulations. EEUU: McGraw-Hill Construction.

**MILLER, H. Clarín, Arquitectura.** Sexta entrega BIM: ventajas de su aplicación. 18 de noviembre 2015. Disponible en: [http://arq.clarin.com/arquitectura/Sexta-ventajas-aplicacion-concepto-BIM\\_0\\_1469853468.html](http://arq.clarin.com/arquitectura/Sexta-ventajas-aplicacion-concepto-BIM_0_1469853468.html) (Mayo 2016)

**MINISTERIO DE FOMENTO, 2015.** Artículo: El Ministerio de Fomento constituye la Comisión para la implantación de la metodología BIM. Disponible en: <http://www.fomento.gob.es/MFOMB Prensa/Noticias/El-Ministerio-de-Fomento-constituye-la-Comisi%C3%B3n-la/1b9fde98-7d87-4aed-9a46-3ab230a2da4e>. (Julio 2016)

**NISSIM, L. IBIMI.** IFC: cos'è? e come è fatto? 30 diciembre 2015. Disponible en: <http://www.ibimi.it/ifc-cose-e-come-e-fatto/>. (Junio 2016)

**SARA IBÁÑEZ LOZANO.** Artículo: BIM es el presente, no el futuro. Disponible en: <http://www.bimbarcelona.com/bim-es-el-presente-no-el-futuro/> (Junio 2016)

**STUDIOSEED.** LOD en el centro de cómo hacer un presupuesto para un proyecto BIM. Disponible en: <http://www.studioseed.net/blog/lod-nivel-de-desarrollo-en-el-centro-de-como-hacer-un-presupuesto-para-un-proyecto-bim/>. (Julio 2016)

**TOGORES FERNÁNDEZ, R. Y OTERO GONZÁLEZ, C.** Programación en AutoCAD con Visual Lisp.