



MODELADO PARAMÉTRICO 3D, RENDER Y ANIMACIÓN CON SOFTWARE LIBRE: INTERACCIÓN FREECAD + BLENDER

RUBÉN DARÍO MORELLI

Universidad Nacional de Rosario, República Argentina (rdm@fceia.unr.edu.ar)

HERNÁN ALFREDO PANGIA CTENAS

Universidad Nacional de Rosario, República Argentina (hernanpangia@gmail.com)

LUIS SEBASTIÁN NIEVA

Universidad Nacional de Rosario, República Argentina (5e6anieva@gmail.com)

ABSTRACT

The purpose of this work is to spread the last experiences and results of a research on the use of *Free Software* and *Freeware* for teaching Graphic Representation that began three years ago. We describe the state of the Art from a conceptual framework and we present a 3D parametric modeling exercise of a mechanical assembly. We assigned materials, rendered and made an animation using *FreeCAD* and *Blender* interactively.

Both are *Free Software* and, also, free of charge. By doing this exercise, the design options that these CAD software provide can be appreciated. *Free Software* philosophy, as a new paradigm in this era, generates resistance to change in most people. We, as teachers and final users, want to debunk myths about *Free Software* and put ourselves at other colleagues' disposal to encourage their use.

KEYWORDS: FreeCAD, Blender, Parametric Modeling, Rendering, Animation.

INTRODUCCIÓN

En el año 2012 comenzamos una Investigación sobre el uso de software libre y gratuito en la enseñanza de la representación gráfica con el propósito de encontrar alternativas a los programas CAD tradicionales debido al elevado costo de sus licencias de uso, lejos del alcance para estudiantes, docentes e instituciones públicas como las Escuelas y Universidades que en nuestro país, Argentina, son gratuitas y de ingreso libre e irrestricto.

La actividad del diseño, ya sea en la enseñanza o en la profesión, se basa en el uso intenso de software CAD. Si bien nuestra investigación es en el área Educación,



pretendemos también sumar nuestro aporte al campo profesional en lo referido al diseño ingenieril y arquitectónico en general. En lo referido a nuestra disciplina, *“debido al alto costo de las licencias de los programas CAD más difundidos (en nuestra región AutoCAD, CATIA, MICROSTATION, SOLID WORKS, SOLID EDGE, etc.), resulta inaccesible para la mayoría disponer de licencias originales de alguno de estos programas, dependiendo de las instituciones lograr algún tipo de convenio (que no siempre es factible) con cualquiera de las empresas propietarias del software para no caer en la ilegalidad”* [1].

A no ser que dichas empresas dispongan de versiones de software CAD con licencias de costo cero para uso educativo exclusivamente, como afortunadamente está sucediendo en los últimos tiempos con las grandes empresas de software CAD privativo, una concesión tardía que probablemente se deba a la necesidad de no perder mercado. El problema se traslada a los jóvenes profesionales que egresan de nuestra universidad (ingenieros, arquitectos, etc.) ya que instalar un estudio profesional o una pequeña empresa les insume un alto costo al que además se debe agregar el precio a pagar por licencias de software de diseño, de ofimática, etc. De acuerdo con Hernández: “A medida que pasan los años y la sociedad de la información avanza, el gasto en tecnologías de la información será mayor y mayores serán aún los ahorros en coste de licencia que el software libre representa” [2]. Esta situación por sí sola ya es una buena motivación para investigar alternativas de *Software Libre* y/o preferentemente gratuito y la correspondiente migración a estas herramientas.

Este trabajo tiene por objetivo la difusión de nuestras últimas experiencias y resultados de Investigación sobre uso de *Software Libre* y *Freeware* para la Representación Gráfica Técnica que comenzamos hace tres años. Se describe el estado actual del arte desde un marco conceptual y luego se presenta el desarrollo de un ejercicio de modelado paramétrico 3D de un conjunto mecánico, con asignación de materiales, render y animación, donde se podrán apreciar las posibilidades de diseño utilizando en forma combinada dos programas CAD que son *Software Libre* y gratuitos: *FreeCAD* y *Blender*.

MARCO CONCEPTUAL

Es importante saber diferenciar entre los conceptos de *Software Libre* y *Freeware*. ¿Qué es *Software Libre*? Es todo software que respeta la libertad de los usuarios sobre el producto que ha adquirido y por lo tanto puede ser usado, copiado, estudiado su código fuente, modificado y redistribuido libremente. Por ejemplo: los programas



FreeCAD, Blender, Gimp e Inkscape. Aunque generalmente es gratuito, el *Software Libre* puede tener un costo que en general es bajo, sin perder sus características definitorias. Fue Richard M. Stallman, programador estadounidense y fundador del movimiento por el *Software Libre* en el mundo, quien definió las cuatro clases de libertades¹ que debe cumplir un *software* para ser Libre, desarrollando toda su filosofía en [3]. ¿Entonces cuál es el *Software* que no es libre? Es el denominado *Software Privativo*, que expresa en sus licencias de uso lo contrario del concepto de *Software Libre*: al adquirir cualquier *software* privativo, en realidad lo que se adquiere es una licencia de uso (EULA, que significa *End User License Agreement*), o sea, se adquiere un permiso para usar dicho programa bajo determinadas condiciones que fija el propietario del *software* y coincidimos con Adell et al. cuando dicen que “en realidad no hemos comprado el software, sino que simplemente hemos adquirido el derecho a usarlo con numerosas restricciones, es decir, que el programa no es nuestro” [4]. Son ejemplos de *Software Privativo*: *AutoCAD, SolidWorks, SketchUp y Photoshop*. Respecto a las licencias de uso, “dentro del mundo del *Software Privativo*, cada empresa define su propia licencia, sin embargo pueden mencionarse algunas características, en general comunes: limitación/prohibición de ciertos tipos de usos, limitación en el número de usuarios finales, expiración luego de cierto tiempo, prohibición de redistribuir, copiar o acceder al código fuente, entre otras” [1]

¿Qué es el *Freeware*? El *Freeware* es *Software Privativo* pero de licencia gratuita para su uso por un tiempo determinado o no, y en general de libre disponibilidad tanto para las instituciones como para docentes y alumnos, como así también para los particulares. Por ser *Software Privativo*, el *Freeware* es de código cerrado. En la mayoría de los casos, la licencia gratuita es restringida al uso educativo o personal como sucede con *SketchUp Make y DraftSight*, o sólo uso educativo como *AutoCAD* para estudiantes, educadores y escuelas. Desde el punto de vista de nuestra investigación, y como usuarios finales no desarrolladores, nos interesan tanto el *Software Libre* gratuito como el *Freeware*, ya que con ambos podemos

¹ Define Stallman:

“Con *software libre* nos referimos a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software. Nos referimos especialmente a cuatro clases de libertad: Libertad 0: la libertad para ejecutar el programa sea cual sea nuestro propósito. Libertad 1: la libertad para estudiar el funcionamiento del programa y adaptarlo a tus necesidades - el acceso al código fuente es condición indispensable para esto. Libertad 2: la libertad para redistribuir copias y ayudar así a tu vecino. Libertad 3: la libertad para mejorar el programa y luego publicarlo para el bien de toda la comunidad - el acceso al código fuente es condición indispensable para esto.” [3].



avanzar en el aprendizaje de nuestra disciplina Representación Gráfica en un marco de libertad y legalidad. Pero el *Software Libre* va mucho más allá de esta situación de resolver el problema económico-legal con respecto a las licencias, ya que su filosofía se inscribe en un nuevo paradigma ético y social dentro de la sociedad del conocimiento, estableciendo un marco propio de referencia moral, política y legal, bajo premisas indispensables en sociedades democráticas como la solidaridad, la cooperación y la libertad. En este sentido, Buxarrais Estrada y Ovide afirman:

“El software libre tiende a organizarse de forma comunitaria, en el cual voluntarios contribuyen con distintas partes del programa, traducciones, documentación, pruebas del software, etc. Así, es un movimiento que busca la inclusión de todos, aportando cada uno lo que puede en función de sus habilidades/conocimientos, el trabajo colaborativo y la generosidad para participar en un proyecto cuyo resultado final tendrá como única restricción el hecho de que todo el que lo recibe está obligado a garantizar las mismas libertades con que lo recibió a aquellos a quienes se lo venda o regale” [5].

Para nosotros es muy importante poder aplicar esta filosofía en la enseñanza de la Representación Gráfica.

En general, el usuario final piensa que el *Software Libre* es para expertos y lo asocia exclusivamente al sistema operativo *Linux*. Hoy este concepto está errado por la evolución del *Software Libre* en los últimos años. Tal vez se piense así porque la mayoría aprendimos a utilizar *software* dentro del paradigma comercial del *Software Privativo*. De acuerdo con Romero:

“El software libre aplicado a contextos educativos, ofrece posibilidades que pueden favorecer el proceso de enseñanza - aprendizaje en función de los destinatarios, de sus necesidades, su nivel de formación ya que puede ser modificado y adaptado en función de nuestros intereses y de los objetivos que persigamos.” [6].

Durante estos tres años de investigación, pudimos avanzar con distintas alternativas de *Software Libre* gratuito y *Freeware CAD*, que se resumen en la Tabla 1. Los primeros resultados importantes de nuestra investigación fueron experiencias de migración de *AutoCAD* a programas *Freeware* como *SketchUp* y *DraftSight*.

Para ejercicios de modelado 3D y proyecciones automáticas trabajamos con las versiones gratuitas de *SketchUp* (SU 8, Make 2013 y 2014). Una limitación importante de estas versiones es la no generación de planos y *layouts*. Pero pudimos resolver esta dificultad combinando el trabajo de *SketchUp* con *DraftSight*, un *CAD 2D* similar a *AutoCAD LT*, con el que sí se pueden hacer *layouts* en Espacio Papel.

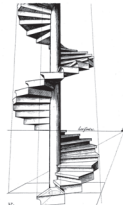


Tabla 1. *Software Libre y Freeware* que estamos estudiando y utilizando.

Tipo	Software Libre y gratuito	Freeware	Website oficial - Download
CAD 2D	<i>LibreCAD</i>		http://librecad.org/
		<i>DraftSight</i>	www.3ds.com/es/productos-y-servicios/draftsight/
Modelado paramétrico 3D	<i>FreeCAD</i>		http://freecadweb.org/
Modelado 3D caras		<i>SketchUp</i>	http://www.sketchup.com/es
Modelado 3D, iluminación, render y animación	<i>Blender</i>		http://www.blender.org/
Edición de imágenes y Diseño Gráfico	<i>Gimp</i>		http://www.gimp.org/
	<i>Inkscape</i>		http://www.inkscape.org/
Paquete de Oficina	<i>LibreOffice</i>		https://es.libreoffice.org/

Para utilizar estos programas gratuitos en forma combinada se puede seguir la siguiente metodología:

“Con SketchUp hacer el modelado 3D y operaciones geométricas como la generación de vistas planas y desarrollos. Luego, mediante la exportación de proyecciones planas al formato dxf, pasar a trabajar con el programa DraftSight para completar la representación de las vistas, acotado, y finalmente hacer las presentaciones para salida en pdf o papel” [7].

Combinándolos así se pueden resolver muchos ejercicios de representación gráfica que originalmente se resuelven con *AutoCAD*. No obstante, otras limitaciones como la imposibilidad del modelado de sólidos en *SketchUp* gratuito nos llevan a continuar en la búsqueda de otras alternativas libres. Sin embargo, sigue siendo válido el criterio del uso combinado de los programas libres, y este criterio es el que aplicamos en la resolución del ejemplo que estamos presentando. La Fig. 1 es un gráfico de interacción de las aplicaciones gratuitas que hemos investigado. El área poligonal en magenta, sombreada con un rayado, muestra cómo se relacionan e interactúan los programas *FreeCAD* y *Blender* utilizados en este trabajo.

DESARROLLO DEL TRABAJO

El proyecto consiste en diseñar una rueda de carro de compras. Para los datos técnicos del tipo de rueda tomamos como referencia y adaptamos un ejemplo obtenido de

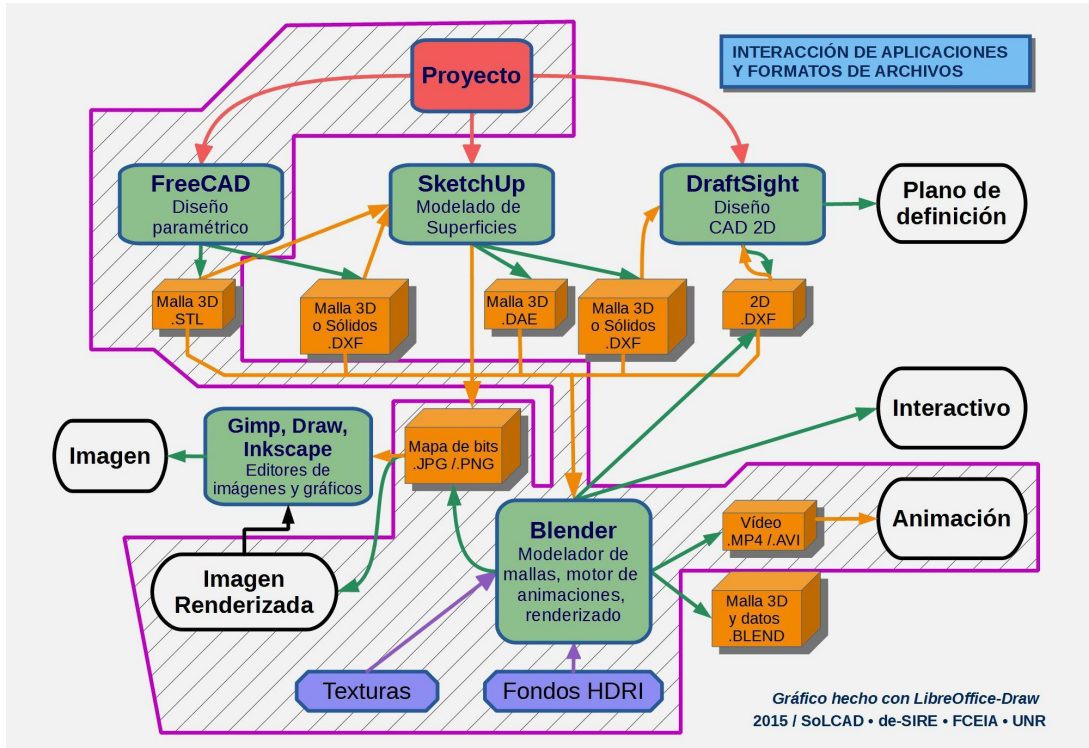


Fig. 1 - Interacción de aplicaciones Software Libre y Freeware para representación gráfica.

bibliografía clásica sobre Dibujo Mecánico [8] según la Fig. 2. Utilizamos los programas *FreeCAD* y *Blender*. Ambos son programas multiplataforma, y se comportan de la misma manera tanto en *Windows*, como en *GNU/Linux* o *Mac OS X*. El proceso de diseño se organizó en dos etapas:

1ª) Con *FreeCAD* hicimos el modelado paramétrico 3D de cada componente del rodamiento, mostrando el resultado con una perspectiva axonométrica explotada y animada donde se aprecia el ensamblaje del conjunto.

2ª) Con *Blender* se trabajó para lograr la imagen fotorrealista del producto. Los componentes se obtuvieron importando desde *Blender* todos los archivos exportados desde *FreeCAD* en formato *STL*.

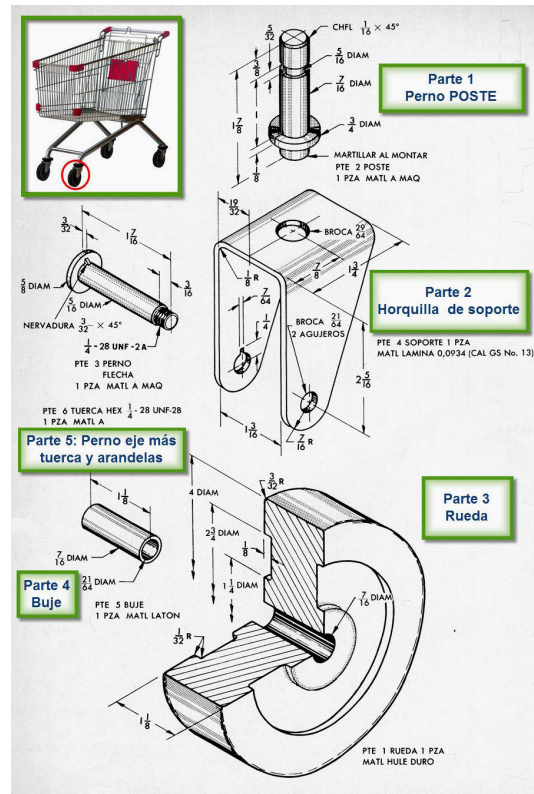


Fig. 2 - Datos rueda de carro de compras.



Se hizo la asignación de materiales y atributos de diseño de producto, terminando la presentación gráfica con *render* en imágenes ráster fotorrealistas y un vídeo de presentación.

1ª ETAPA

La interfaz de *FreeCAD*² está organizada bajo el concepto de “*workbenchs*” o escenarios de trabajo (*Sketcher*, *Drawing*, *Part*, *Part Design*, *Mesh Design*, *Arch*, *RayTracing*, etc.) que permiten seleccionar las barras de herramientas necesarias en base a la tarea a desarrollar. En este modelado sólo utilizamos los escenarios “*Sketcher*”, “*Part Design*” y “*Part*”. El escenario “*Sketcher*” se utiliza para crear *croquis* de geometrías 2D, considerándose este proceso el punto de partida para la mayoría de los modelos CAD, pues por ejemplo un boceto 2D simple puede ser extrudido en una forma 3D. También desde *croquis* 2D hechos sobre las caras de un objeto 3D se pueden crear extrusiones en la superficie de dichos objetos. Junto con las operaciones booleanas, el *Sketcher* forma el núcleo del diseño de la forma sólida generativa. *Sketcher* trabaja con *croquis* libres 2D al que luego se le van agregando restricciones geométricas para darle precisión. Posee un solucionador de restricciones que calcula los grados de libertad de la geometría 2D y permite la exploración interactiva de los grados de libertad del *croquis*. “*Part Design*” contiene herramientas de diseño por modelado sólido (extrusión, revolución, transformaciones, etc.) que incluyen herramientas de *croquis* del escenario *Sketcher*, pues *Part Design* es dependiente de los *croquis* con restricciones. “*Part*” tiene comandos que permiten operaciones más avanzadas como la creación de sólidos primitivos, formas, chaflanes, giros y operaciones booleanas, comportamiento paramétrico, etc. Comenzamos modelando la Parte 1: “Perno Poste”. La pieza es un cilindro desarrollado a lo largo del eje z, con secciones y forma variable. El objeto se pudo modelar fácilmente mediante una revolución. El escenario de *FreeCAD* correspondiente es *Sketcher*, que permite generar un sólido por revolución a partir del perfil de la pieza

² “*FreeCAD* es una aplicación de modelado paramétrico CAD/CAE de arquitectura modular. Está orientado a la ingeniería mecánica y al diseño de producto, pero también se ajusta a un amplio rango de usos de otras especialidades de ingeniería, y también a arquitectura. Sus herramientas son similares a las de *Catia*, *SolidWorks* o *Solid Edge*, por lo tanto también pertenece a la categoría de MCAD, PLM, CAx y CAE. Cuenta con una completa interfaz gráfica de usuario basada en el famoso entorno Qt, con un visualizador 3D basado en *Open Inventor*, que permite un rápido renderizado de escenas 3D y una representación gráfica de escenas muy accesible. Sus capacidades se basan en el kernel *OpenCascade*, un núcleo CAD de nivel profesional que cuenta con la manipulación de geometría 3D avanzada y objetos. Algunos módulos aún está en una etapa de desarrollo (como el módulo de mecanizado CAM) por lo que todavía para el usuario final no es recomendado su uso en entornos de producción” [9].

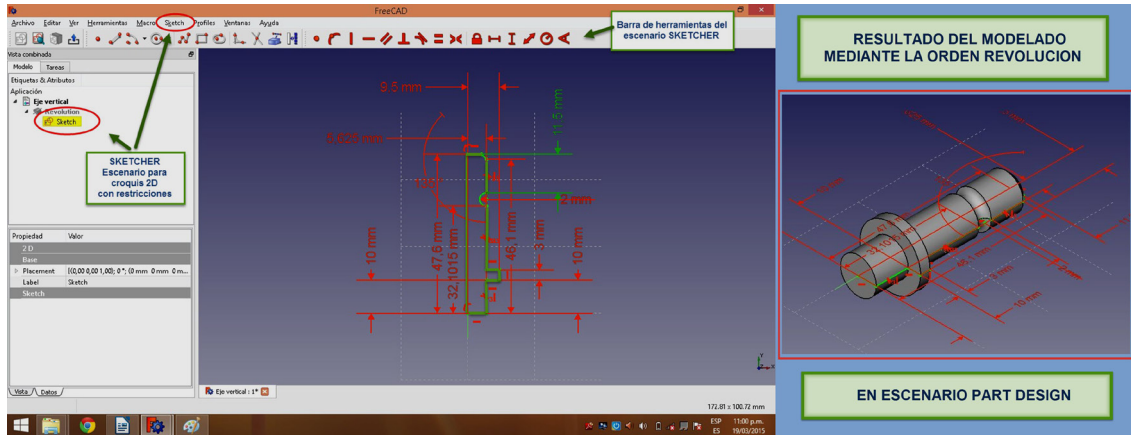


Fig. 3 - Generación del Perno poste. Parte 1.

hecho con un *croquis* 2D restringido como muestra la Fig. 3 a la izquierda. Las restricciones son imprescindibles para generar el sólido, y se dan desde la barra de herramientas o teclado. El gráfico es color blanco al inicio y cuando está totalmente restringido pasa a color verde. Terminado el *croquis*, fuimos al escenario *Part Design* que tiene su propia barra de herramientas. Con la orden “Revolucionar”, una vez seleccionado el *croquis*, se logró el resultado buscado que se ve en la Fig. 3 a la derecha.

El modelado de la Parte 2 - Horquilla, se muestra en secuencia en el collage de la Fig. 4. El escenario inicial es *Sketcher*. Allí se hizo un *croquis* con restricciones del plano lateral de la horquilla y luego en el escenario *Part* se hizo la extrusión de la forma.

A continuación se hizo, mediante *croquis* de cara y extrusión, el modelado del prisma que luego se restó a la pieza principal mediante la operación booleana “Diferencia” para conformar la horquilla, que es una chapa de poco espesor. El redondeo en las aristas de las dos alas donde se curva la horquilla se hizo con la orden “Redondear”.

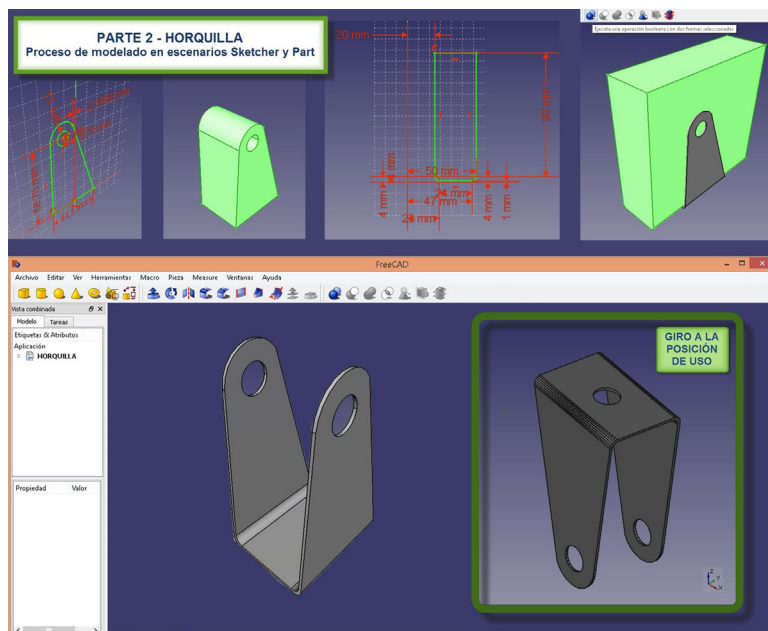


Fig. 4 - Generación de la Horquilla. Parte 2.



El modelado de la Parte 3 - Rueda se ve en secuencia en la Fig. 5. Se utilizaron los escenarios *Sketcher* (*croquis con restricciones*) y *Part Design* (orden “Revolución”). En la Fig. 6 se ve el proceso de modelado de la Parte 5 - Perno eje. También inicia con el escenario *Sketcher* para *croquis 2D con restricciones* y luego se modela en el escenario *Part* cada parte que luego se unen por operación booleana. La rosca del extremo del perno se genera por barrido de una sección triangular en trayectoria helicoidal (orden “Primitivas geométricas”). Cada parte está en un archivo individual. Para montar el conjunto se abrió un archivo nuevo de *FreeCAD* y desde menú Archivo/Fusionar proyecto/ se abrió la ventana “*Merge project*” donde se seleccionaron los archivos de cada parte. Cada pieza se orientó y ubicó en su

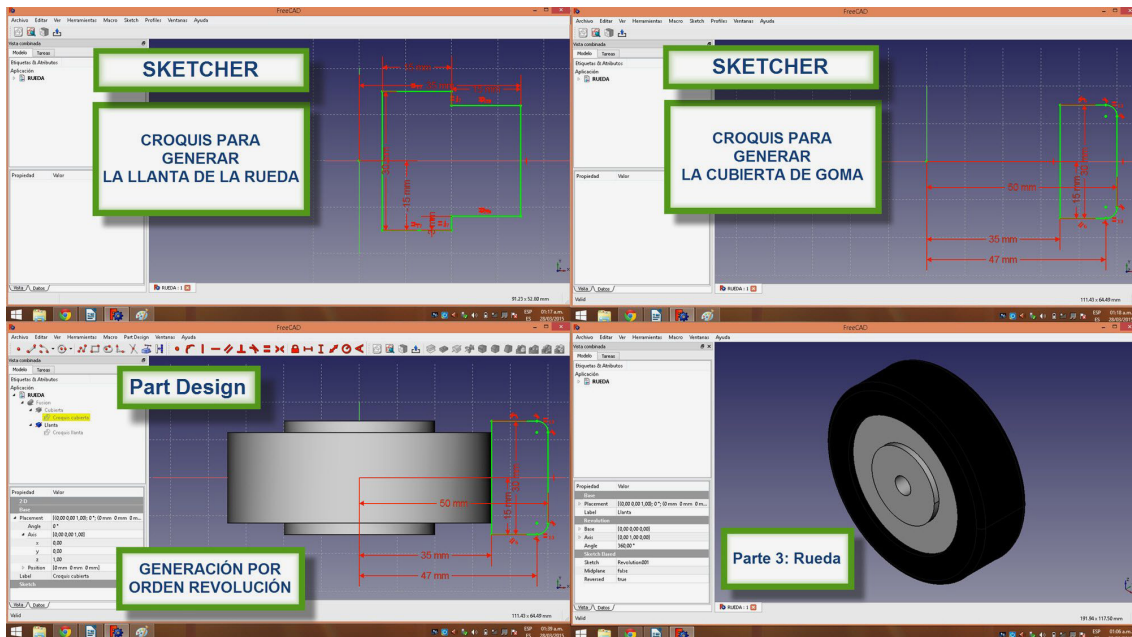


Fig. 5 - Generación del sólido de la Rueda. Parte 3.

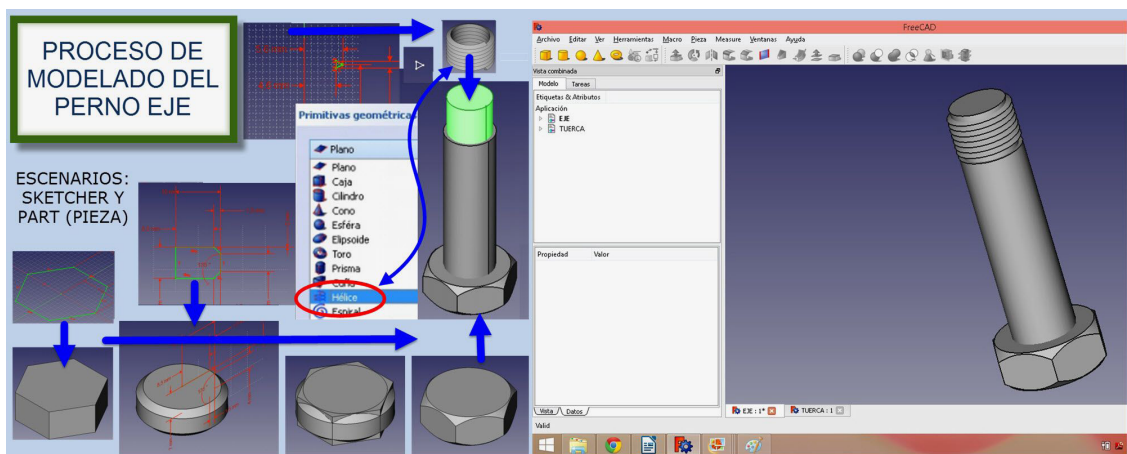


Fig. 6 - Generación del Perno eje. Parte 5.



posición de uso para el ensamblado. La Fig. 7 muestra la perspectiva explosionada del conjunto. Se puede ver un vídeo del montaje en [10]. El vídeo fue obtenido por captura de una animación hecha con *FreeCAD*.

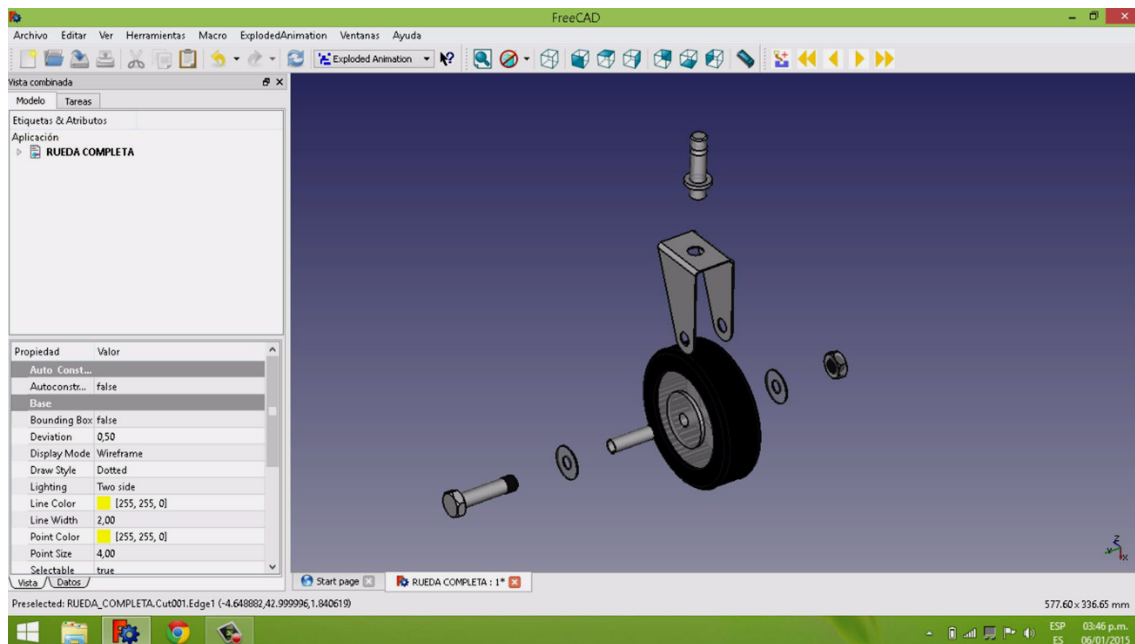


Fig. 7 - Perspectiva explosionada del conjunto.

2ª ETAPA

*Blender*³ es un potente programa para modelado 3D, iluminación, render, animación y creación de gráficos tridimensionales. La interfaz⁴ predeterminada se compone de Editores, Cabeceras, Botones contextuales, Paneles y Controles. Se llaman Ventanas a la parte del programa que responde a una función específica (ventana de Vista 3D, de Propiedades, de Edición de secuencias de video, de Editor de nodos, etc.). Cada una tiene su propia Cabecera en la parte superior o inferior. Como ya se explicó, se importó cada parte de la rueda en formato *STL*, que define la geometría de

³ “*Blender también es para composición digital utilizando la técnica procesal de nodos, edición de vídeo, escultura y pintura digital. Se pueden desarrollar videojuegos ya que posee un motor de juegos interno. Tiene la capacidad de representar simulaciones físicas como deformación de sólidos, acciones con fluidos, humo, etc.*” [11]

⁴ Rigen 3 principios invariables para la interfaz:

- 1) Nada se superpone: no hay ventanas ni herramientas flotantes, todo está en un mismo plano;
- 2) No hay bloqueos: ninguna herramienta espera que el usuario haga algo, no hay área de comandos;
- 3) No modal: deberá ser siempre tan consistente y predecible como sea posible sin cambiar métodos comunes (mouse, teclado).

La interfaz del usuario es un entorno multitareas, simultáneo, automatizable y 100% personalizable. Al igual que con la ventana principal, cada ventana nueva se puede subdividir. Se pueden tener muchas configuraciones de pantallas simultáneas y moverse entre ellas con velocidad. Es posible ubicar entornos en múltiples monitores.



objetos 3D como “mallas” (compuestas por vértices, aristas y caras triangulares) sin información de color, texturas o propiedades físicas que sí dan otros formatos CAD. (Fig. 8).

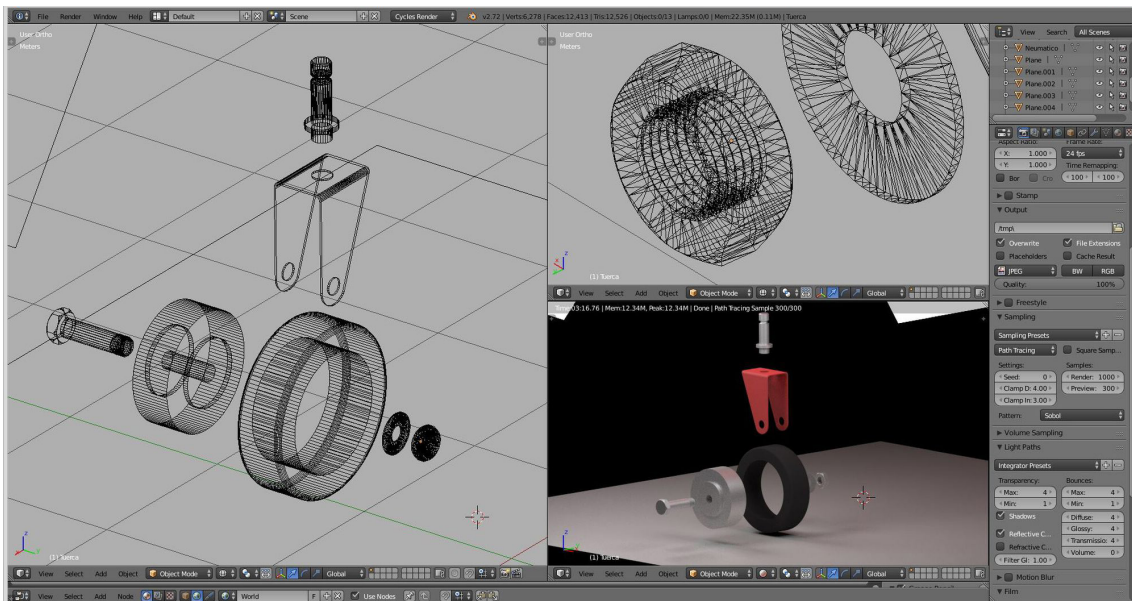


Fig. 8 - Importar malla STL en Blender.

Preparación de la escena: se creó una escena simple, constituida por un plano de apoyo, un maletín modelado - *in situ* -, luces y una cámara principal. Tal como en una sesión de fotos se preparó el lugar para la toma, se colocaron los objetos en suposición, se los iluminó⁵ y finalmente se ajustaron los parámetros de la cámara. Se puede ver el resultado de esta escena en la Fig. 9.

En referencia a los materiales, se utilizó chapa pintada para la horquilla, hierro cincado para perno, tornillo, tuerca y arandela, aluminio y goma para la rueda. El estuche es de plástico rugoso semi-mate y el interior es de una suerte de tela tejida con una trama suave. Con respecto al render, *Blender* tiene dos motores para renderización: uno interno o nativo, llamado “*Blender Render*”, que fue creado para que sea veloz, y el otro motor es “*Cycles*” que busca la calidad y demanda mucho más recursos de *hardware*. El motor nativo solo simula iluminación directa y sombras. *Cycles* considera el camino que la luz recorre en la realidad, solo que en sentido contrario. Es decir, de la cámara a la fuente de luz, calculando los rebotes, refracciones y alteraciones de color. En este trabajo se utilizó el motor *Cycles*, y el resultado se muestra en la Fig. 10.

⁵ Las fuentes de Luz provienen de paneles emisivos dispuestos alrededor y por encima de los objetos con una luz cálida y una fuente extra de luz roja para resaltar el brillo de los materiales.



Fig. 9 - Escena del maletín con los componentes del proyecto.

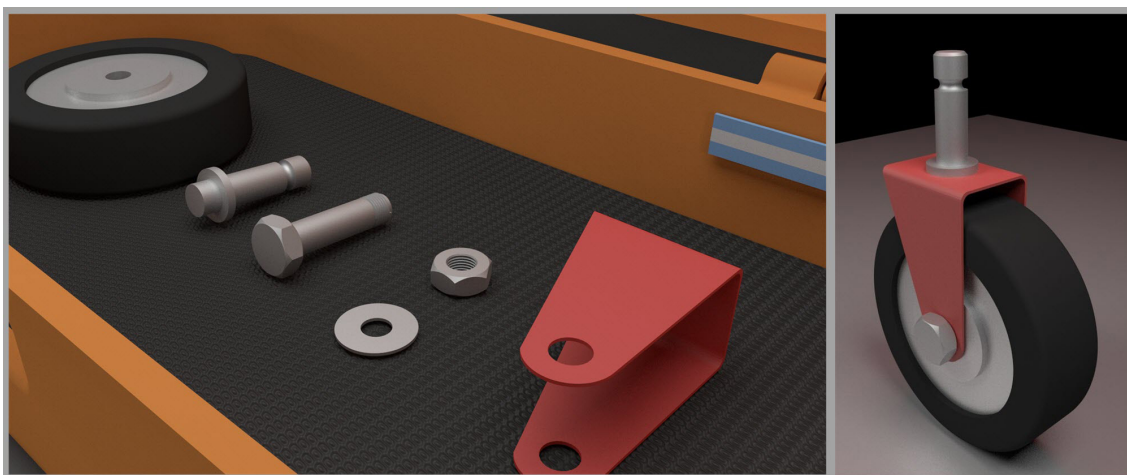


Fig. 10 - Render con atributos de diseño de producto en Blender.

Por último creamos una salida en vídeo, que se puede ver en [12], donde se muestra un ejemplo de presentación del producto. Sintéticamente podemos decir que la tarea de animar dentro de *Blender* es similar a la que usan otras aplicaciones. Se basa en una línea de tiempo y claves (marcas que guardan propiedades o transformaciones de cada entidad en ese momento). Antes de comenzar un proyecto de animación medianamente complejo, se debe crear un guion. Es igual que con una imagen fija sólo que secuencial, a razón de 24 fps (frames por segundo) en el formato de vídeo que se desee, o como imágenes sueltas.



CONCLUSIONES

Através de un ejercicio relativamente sencillo de modelado paramétrico 3D de un conjunto mecánico apreciamos las posibilidades actuales de diseño CAD con *Software Libre* y consolidamos nuestro camino hacia el objetivo central de nuestra investigación, el cual es trabajar con *software* legal y gratuito. Experimentamos con la interacción de los programas *FreeCAD* y *Blender*, verificando sus excelentes posibilidades de diseño. Aprendimos haciendo. Tal vez estos programas tengan una curva de aprendizaje más alta que los clásicos privativos, pero están evolucionados y hay mucho material disponible para un correcto aprendizaje. La filosofía del *Software Libre*, como nuevo paradigma de esta era, genera en muchos una resistencia al cambio. Como docentes y usuarios finales pudimos derribar mitos sobre el *Software Libre* y nos ponemos a disposición para animar a otros colegas a que usen estos programas.

REFERENCIAS

- [1] Morelli, R. D., & Morelli L. R. (2012). El Software Libre en la enseñanza de la Representación Gráfica. *IV Congreso Internacional de Expresión Gráfica en Ingeniería, Arquitectura y Carreras afines - Egrafía 2012. Gráfica del diseño. Tradición e Innovaciones* (pp. 219-223). La Plata, Argentina. Recuperado de <http://goo.gl/Bzr246>
- [2] Hernández, J. M. (2005) *Software libre: técnicamente viable, económicamente sostenible y socialmente justo*, p. 154. Barcelona: Infonimia, 2005. Recuperado de <https://goo.gl/AoV9eX>
- [3] Stallman, R. M. (2004). *Software libre para una sociedad libre*, pp. 59-60. Recuperado de <https://goo.gl/KYzyQ0>
- [4] Adell, J., & Bernabé, I. (2006). *Software libre en educación*, p. 7. Recuperado de <http://goo.gl/tJ1Z92>
- [5] Buxarrais Estrada, M. R., & Ovide, E. (2011). El impacto de las nuevas tecnologías en la educación en valores del siglo XXI. *Sinéctica*, (37), pp. 1-14. Recuperado en 05/15 de <http://goo.gl/Kohg2z>
- [6] Romero, T. A. (2006). *Moodle, Unimos Mentes, Creamos Conocimiento Libre*. Ponencia presentada al VI Congreso Internacional Virtual de Educación CIVE 2006, Islas Baleares.
- [7] Morelli, R. D., & Pangia Ctenas, H. A. (2013, Noviembre). Enseñar utilizando programas CAD gratuitos. *X International Conference on Graphics for Arts and Design - Graphica 2013 - Expressão Gráfica - Tecnologia e Arte para Inovação*. Florianópolis, SC, Brasil. Recuperado de <http://goo.gl/WqmwXe>



- [8] Jensen, C. H., & Mason, F. H. S. (1971). *Fundamentos de Dibujo Mecánico*, p. 97. México, Panamá, São Paulo, Nueva York: Libros Mc Graw-Hill.
- [9] FreeCAD Wiki. (2015). *Documentación Oficial y manuales en línea*. Recuperado en 02/15 de <http://goo.gl/3lp9e2>
- [10] Nieva, L. S. (2015, Abril 04). *Animación ensamblado rueda para carrito de compras* [animacion_rueda_carrito.avi]. Recuperado de <http://goo.gl/n8sMxd>
- [11] Blender Wiki. (2015). *Blender*. Recuperado en Febrero 2015 de <http://goo.gl/iWJ5WQ>
- [12] Pangia Ctenas, H. A. (2015, Abril 17). *Vídeo del producto* [muestra_producto.avi]. Recuperado de www.fceia.unr.edu.ar/solcad/muestra_producto.avi