



Universidad de las Ciencias Informáticas

Facultad 5

**Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas**

Título: Módulo de gestión y visualización de roscas en
superficies cilíndricas

Autora: Yinelys Martin Martinez

Tutor: Ing. Ernesto Carrasco De la Torre

La Habana

Junio de 2013

“Los estudiantes son en su mayoría revolucionarios. Revolucionarios por naturaleza, porque pertenecen a ese estrato de jóvenes que se abren a la vida y que adquieren todos los días conocimientos nuevos”.

Ernesto “Che” Guevara.

DATOS DE CONTACTO

Ing. Ernesto Carrasco De la Torre

Institución: Universidad de las Ciencias Informáticas (UCI).

Título: Ingeniero en Ciencias Informáticas.

e-mail: ecarrasco@uci.cu.

Graduado en la UCI en el año 2009, especialista con 4 años de experiencia en el desarrollo de *software* y 3 años de experiencia en el desarrollo de sistemas de diseño asistido por computadora (sistemas CAD).

DECLARACIÓN DE AUTORÍA

Declaramos ser autores del presente trabajo de diploma y autorizamos a la Universidad de las Ciencias Informáticas los derechos patrimoniales del mismo, con carácter exclusivo.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Yinelys Martin Martinez

Ing. Ernesto Carrasco De la Torre

Firma del autor

Firma del tutor

AGRADECIMIENTOS

A mi padre por brindarme su apoyo incondicional desde el primer día, por hacer de mí la mujer que soy hoy y por guiarme siempre para escoger el camino correcto en la vida.

A mi madrastra por desempeñar en mi vida un papel importante y ayudarme en momentos difíciles, y darme siempre el mejor consejo.

A mis tías María del Carmen, Lourdes y Ángela por ser como segundas madres y brindarme apoyo y consejo siempre que lo he necesitado, a Maye por estar pendiente siempre a mis problemas.

A mi hermana Yanelys por alegrar mis días y compartir conmigo momentos especiales.

A mi sobrinita Dagneilys y mi hermanita Yagna por ser ambas desde su primer día de vida la luz de mis ojos.

Al resto de mi familia que es igual de importante y que junto a mi han esperado cinco años para ver este sueño convertido en realidad.

A mi mejor amiga Giselle que en todos estos años juntas se convirtió en una hermana más.

A los amigos que siempre están ahí para cualquier cosa que se necesite, a Oscar, Abel, Arlan, Jorge Javier y a los que ya no están aquí pero siguen siendo amigos.

A todos los profesores que aportaron su granito de arena a mi formación profesional durante toda mi vida.

A la Revolución por brindarme la oportunidad de estudiar en esta universidad, y gracias a esta universidad por darme la oportunidad de formarme como una profesional capacitada.

A la persona más importante durante todo este último curso, gracias a mi tutor, amigo, compañero incondicional en las buenas y en las malas, a mi pareja, a Ernesto, por ser la persona que me brindó su mano desde el comienzo y se mantuvo fiel a pesar de los obstáculos del largo camino, gracias por ir conmigo y ayudarme a llegar a la meta, gracias por hacerme crecer como persona y como profesional, gracias por ayudarme a cumplir mi objetivo y por estar ahí para vivir conmigo ese momento, gracias por todo, mi amor.

A mi madre querida.

Mi ángel guardián, mi luz, mi guía.

Siéntete orgullosa hoy y siempre de tu hija.

RESUMEN

La modelación de roscas en los sistemas *CAD* permite la creación de piezas que incluyen roscas además de asociar roscas estandarizadas a superficies cilíndricas creadas mediante operaciones 3D.

A partir de esta premisa se realizó todo un estudio del arte relacionado con términos y conceptos de roscas, haciendo mayor énfasis en el análisis de los sistemas *CAD* líderes del mercado, con el objetivo de obtener una guía para el desarrollo del módulo propuesto como solución de esta investigación. Como resultado final del estudio realizado se elaboró una propuesta de solución basada en la visualización de roscas mediante texturas, logrando la creación edición y eliminación de roscas en superficies cilíndricas a partir de los datos obtenidos de las normas y estándares utilizados en la industria para la especificación de roscas.

Se obtuvo como resultado un módulo de gestión y visualización de roscas en superficies cilíndricas que permite la creación, edición y eliminación de roscas. Este módulo se integró como parte de las funcionalidades del sistema GALBA-CAD. En el desarrollo del módulo, se utilizó la arquitectura modelo / vista que propone el *framework* Qt y la implementación de componentes visuales para acceder a las especificaciones de la rosca, permitieron la extensibilidad y reusabilidad del módulo hacia otras funcionalidades del sistema GALBA-CAD.

ÍNDICE

Índice 7

Índice de figuras.....10

Índice de tablas.....11

Introducción 1

Capítulo 1: Fundamentación Teórica 4

 1.1. Conceptos generales..... 4

 1.1.1. Definición de rosca..... 4

 1.1.2. Perfil de una rosca 5

 1.1.3. Clasificación de la rosca..... 7

 1.1.4. Sistemas de roscas 9

 1.2. Sistemas propietarios y libres con módulos para la creación de roscas10

 1.2.1. Autodesk Inventor10

 1.2.2. SolidWork.....12

 1.2.3. CATIA13

 1.2.4. VariCAD14

 1.2.5. Salome Geometric.....14

 1.2.6. FreeCAD15

 1.2.7. BRL-CAD16

 1.3. Elementos de *Opencascade*.....17

 1.3.1. Manipulación de geometrías18

 1.3.2. Manipulación de topologías.....19

 1.3.3. Visualización de objetos21

 1.3.4. Marco de trabajo para el desarrollo de aplicaciones22

Conclusiones del capítulo24

Capítulo 2: Solución Propuesta.....26

2.1. Solución propuesta.....	26
2.1.1. Normas y estándares	26
2.1.1.1. Rosca Métrica ISO (<i>ISO Metric Profile</i>).....	27
2.1.1.2. Rosca Métrica Trapezoidal ISO (<i>ISO Metric Trapezoidal Threads</i>).....	28
2.1.1.3. Rosca de tubería ISO (<i>ISO Pipe Threads</i>).....	29
2.1.2. Organización y persistencia de los datos de la rosca	30
2.1.3. Visualización de la rosca.....	31
2.1.4. Conexión con la Base de Datos	33
2.2. Especificación de requisitos	33
2.2.1. Requisitos funcionales	34
2.2.2. Requisitos no funcionales.....	34
2.3. Modelo de Casos de Uso	35
2.3.1. Actor del sistema.....	35
2.3.2. Diagrama de Casos de Uso	35
2.3.3. Descripción de los Casos de Uso.....	36
2.4. Diseño del sistema.....	41
2.4.1. Modelo de dominio	41
2.4.2. Diagrama de paquetes	42
2.4.3. Diagrama de clases.....	43
Conclusiones del capítulo	43
Capítulo 3: Implementación y pruebas	45
3.1. Implementación de la solución propuesta.....	45
3.1.1. Diagrama de componentes	45
3.2. Resultados	46
3.2.1. Crear una rosca	46
3.2.1. Editar una rosca	48
3.2.1. Eliminar una rosca.....	49

3.3. Pruebas al módulo desarrollado	50
3.3.1. Casos de prueba.....	50
3.3.1.1. Caso de prueba # 1	50
3.3.1.2. Caso de prueba # 2	51
3.3.1.3. Caso de prueba # 3	52
3.3.1.4. Caso de prueba # 4	53
3.3.1.5. Caso de prueba # 5	54
3.3.1.6. Caso de prueba # 6	55
3.4. Resultados de las pruebas.....	55
Conclusiones del capítulo	56
Conclusiones	58
Recomendaciones	59
Referencias bibliográficas	60
Glosario de Términos.....	63
Anexos.....	65

ÍNDICE DE FIGURAS

Fig. 1 Perfil de una rosca.	6
Fig. 2 Perfil de la una rosca más detallado.....	6
Fig. 3 Izquierda: rosca de una sola entrada. Derecha: rosca de varias entradas.....	7
Fig. 4 Izquierda: rosca exterior. Derecha: rosca interior.	7
Fig. 5 Tipos de roscas según la forma de su filete.	8
Fig. 6 Izquierda: rosca a derecha. Derecha: rosca a izquierda.....	8
Fig. 7 Visualización de una rosca con <i>Autodesk Inventor</i>	11
Fig. 8 Visualización de una rosca con <i>SolidWork</i>	13
Fig. 9 Visualización de una rosca con <i>CATIA</i>	14
Fig. 10 Estructuras por paquetes de OpenCASCADE.....	17
Fig. 11 Conectividad entre formas topológicas.....	20
Fig. 12 Designación de los identificadores de las etiquetas.	23
Fig. 13: Rosca Métrica ISO.	27
Fig. 14: Designación de la rosca métrica.....	27
Fig. 15 Rosca trapezoidal.	28
Fig. 16: Designación de la rosca trapezoidal.....	28
Fig. 17: Rosca de tuberías.	29
Fig. 18: Designación de la rosca de tuberías.	29
Fig. 19 Ubicación de los datos en el nodo correspondiente a la rosca.	30
Fig. 20: Textura original que representa un paso.	31
Fig. 21: Textura original visualizada después de las repeticiones.	31
Fig. 22: Representación de parámetros U y V de la superficie lateral de un cilindro.	32
Fig. 23 Diagrama de Casos de Uso del sistema.....	36
Fig. 24 Modelo de dominio.....	42
Fig. 25 Diagrama de paquetes del diseño.....	42
Fig. 26 Diagrama de clases.....	43
Fig. 27 Diagrama de componentes.	45
Fig. 28: Pasos para la creación de una rosca en el sistema GALBA-CAD.	47
Fig. 29: Rosca creada en el sistema GALBA-CAD.....	47
Fig. 30: Pasos para editar una rosca creada en el sistema GALBA-CAD.....	48
Fig. 31: Pasos para eliminar una rosca en el sistema GALBA-CAD.....	49
Fig. 32: Rosca eliminada del sistema GALBA-CAD.	50

ÍNDICE DE TABLAS

Tabla 1: Descripción de actor del sistema.....35
Tabla 2: Descripción del Caso de Uso “Crear rosca”.....37
Tabla 3: Descripción del Caso de Uso “Editar rosca”.....38
Tabla 4: Descripción del Caso de Uso “Eliminar rosca”.....39
Tabla 5: Descripción del Caso de Uso “Especificar datos de rosca”.....41
Tabla 6: Caso de prueba # 1.....50
Tabla 7: Caso de prueba # 2.....52
Tabla 8: Caso de prueba # 3.....53
Tabla 9: Descripción de variables del caso de prueba # 3.53
Tabla 10: Caso de prueba # 4.....54
Tabla 11: Caso de prueba # 5.....54
Tabla 12: Caso de prueba # 6.....55
Tabla 13: Resultados de las pruebas.56

INTRODUCCIÓN

El diseño mecánico como disciplina encargada de proporcionar una o varias soluciones para definir un producto de forma que satisfaga determinadas restricciones y requisitos, ha desempeñado un papel fundamental en la fabricación y creación de objetos y sistemas de naturaleza mecánica como: máquinas, equipos, aparatos, dispositivos, entre otros. Dentro de este amplio conjunto de estructuras mecánicas, piezas como tornillos, tuercas, y otras que contienen elementos roscados entre sus componentes, tienen gran importancia en la industria dado que las roscas permiten la fijación y acoplamiento de piezas en sistemas mecánicos. La precisión, exactitud y confiabilidad del diseño de tales estructuras mecánicas ha manifestado un aumento notable desde el surgimiento y desarrollo de la modelación de prototipos virtuales y los sistemas de diseño asistido por computadora o sistemas CAD dadas sus siglas en inglés, los cuales mejoran la calidad y productividad del proceso de diseño.

La Universidad de las Ciencias Informáticas (UCI) como proyecto de la Revolución tiene la misión de formar profesionales capacitados en la rama de la informática, a partir de un modelo pedagógico que vincula de forma dinámica el estudio con la investigación y la producción, acorde con las necesidades sociales del país. En la UCI, dicho modelo pedagógico se realiza sobre un área temática determinada para contribuir así al desarrollo tecnológico del país. Para garantizar el trabajo en dichas áreas temáticas se crearon Centros de Desarrollo conformados por profesores, estudiantes y especialistas. En la facultad 5 se encuentra el Centro de Desarrollo de Informática Industrial (CEDIN), dentro del cual se crea el proyecto Centro de Diseño y Simulación de Estructuras Mecánicas (CDSEM). En dicho proyecto se desarrolló, bajo paradigmas de *software* libre y a partir de los convenios realizados entre Cuba, la Industria Chino-Venezolana de Taladros (ICVT) y PDVSA Industrial, la primera versión del modelador geométrico del sistema denominado GALBA-CAD, contribuyendo de esta manera a la integración de Cuba en el desarrollo de este tipo de sistemas.

La versión actual del modelador geométrico no permite la creación y visualización de elementos roscados lo que imposibilita insertar piezas estandarizadas que incluyen roscas entre sus componentes. Esto trae como consecuencia que no se puedan asociar roscas definidas mediante normas y estándares definidos internacionalmente, a las superficies que se crean en el modelador geométrico. Las deficiencias antes mencionadas disminuyen la funcionalidad, usabilidad y alcance del sistema GALBA-CAD.

A partir de la situación problémica mencionada se plantea como **problema tecnológico**: ¿Cómo insertar las funcionalidades gestión y visualización de roscas en superficies cilíndricas, en el sistema GALBA-CAD?

Se define como **objeto de estudio** de la investigación, la gestión y visualización de roscas en sistemas CAD.

Para dar respuesta al problema tecnológico planteado se define como **objetivo general** de la investigación desarrollar e integrar al sistema GALBA-CAD un módulo informático para gestionar y visualizar roscas en superficies cilíndricas.

Se establece como **campo de acción** la gestión y visualización de roscas en superficies cilíndricas para sistemas CAD basados en tecnología OpenCASCADE.

La investigación presenta la siguiente **idea a defender**: el desarrollo e integración de un módulo para gestionar y visualizar roscas en superficies cilíndricas permitirá al sistema GALBA-CAD crear elementos roscados con parámetros y propiedades definidas en normas y estándares establecidos, aumentando la usabilidad y alcance del sistema mencionado.

Para cumplir con el objetivo planteado se definen las siguientes **tareas de investigación**:

- Elaboración del marco teórico a partir del estudio del estado del arte existente actualmente sobre el tema a investigar.
- Identificación de los datos necesarios para gestionar roscas sobre superficies cilíndricas mediante las normas y estándares relacionados.
- Elaboración de diagramas de clases y paquetes para el diseño del sistema.
- Desarrollo de un módulo informático para la implementación de la solución propuesta.
- Validación del módulo desarrollado mediante la realización de pruebas internas.
- Integración del módulo desarrollado en la versión del Modelador Geométrico del sistema GALBA-CAD.

Para el desarrollo eficiente de dichas tareas fue necesario combinar diferentes **métodos y técnicas en la búsqueda y procesamiento de la información**, los fundamentales son:

A nivel teórico:

Análisis histórico – lógico: Utilizado durante la investigación de los antecedentes y las tendencias actuales relacionadas con el diseño mecánico y el desarrollo de los sistemas CAD y

en la profundización de los conceptos y términos relacionados con el objeto de estudio y el campo de acción.

Analítico – Sintético: Empleado durante el análisis y la selección de las tecnologías y lenguajes de programación a utilizar en el diseño e implementación de la aplicación.

A nivel empírico:

Consulta de fuentes de información: Estudio de documentos y otros tipos de bibliografías con el objetivo de obtener la información necesaria para llevar a cabo la investigación.

Pruebas: En la elaboración de pruebas, para ir comprobando las distintas funcionalidades de acuerdo a los requisitos.

El presente documento está compuesto por tres capítulos, estructurado de la siguiente forma:

En el **Capítulo 1** se profundizan y exponen en los principales conceptos relacionados al dominio de este trabajo y se realiza un estudio del estado del arte sobre los términos y conceptos asociados a las roscas. Además se describen algunos de los sistemas CAD actuales de los cuales se obtienen las características necesarias para el desarrollo de la aplicación. Se expone un estudio de la tecnología OpenCASCADE y sus elementos fundamentales.

En el **Capítulo 2** se describe la propuesta de solución, además se identifican los requisitos funcionales y no funcionales que deben tenerse en cuenta. También se describe, a partir de los requisitos funcionales, un modelo de casos de uso y los diagramas correspondientes a la etapa de diseño.

En el **Capítulo 3** se describe el resultado obtenido de la implementación de la solución propuesta. Además se diseñan y se muestran los casos de prueba para módulo desarrollado.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

El siguiente capítulo aborda los elementos teóricos que soportan la presente investigación. Se definen los principales conceptos relacionados con el dominio del problema. Se expone una descripción de sistemas y herramientas dedicadas al diseño de piezas mecánicas, haciendo énfasis en las principales funcionalidades del modelador geométrico de dichos sistemas y en el módulo para el diseño de roscas, si lo poseen. Además se describen los principales elementos de la tecnología OpenCASCADE para el desarrollo de aplicaciones CAD.

1.1. CONCEPTOS GENERALES

A continuación se presenta un estudio de los conceptos y términos importantes que son necesarios conocer para desarrollar esta investigación. Se describe lo relacionado a las roscas y los sistemas roscados, su utilidad e importancia en la industria actual.

1.1.1. Definición de rosca

Las roscas se han usado durante siglos como medio de unión o fijación de piezas mecánicas. Una rosca se define como una estructura helicoidal envuelta alrededor de un cilindro o cono, siendo la primera una rosca recta y la segunda una rosca cónica (1). Existen además otras definiciones del término rosca que plantean que una rosca está formada por el enrollamiento helicoidal de un prisma llamado filete¹, ejecutado en el exterior o interior de una superficie generalmente cilíndrica, que le sirve de núcleo (2).

También se puede definir una rosca como un reborde en forma de una hélice ya sea en la superficie externa o interna de un cilindro. Las roscas internas se asocian a los agujeros roscados o tuercas, mientras que las roscas utilizadas en tornillos son de tipo externa (3).

Teniendo en cuenta los criterios vistos, se define entonces que una rosca es: un prisma o filete, que se enrolla de manera helicoidal alrededor de un cilindro o cono, tanto en la parte interior como exterior del mismo.

¹ La unión de los flancos, el fondo y la cresta forman en su conjunto lo que es conocido como el prisma de la rosca o también llamado comúnmente filete.

1.1.2. Perfil de una rosca

En toda rosca existen elementos fundamentales que la caracterizan, y que conforman su perfil, estos elementos se mencionan a continuación y se pueden observar mejor en las figuras 1 y 2:

(1)

- **Núcleo:** Es el volumen ideal sobre el que se encuentra la rosca o cuerpo del elemento roscado.
- **Cresta:** Es la superficie exterior de unión de los flancos.
- **Fondo o Raíz:** Es la superficie interior de unión de los flancos.
- **Flancos:** Son las superficies teóricas de contacto que unen la cresta con el fondo.
- **Hilo:** Es la unión de cada uno de los vértices de las crestas a los largo de toda la rosca.
- **Paso:** Es la distancia medida paralelamente al eje entre dos hilos consecutivos. Puede darse en milímetros o en pulgadas, en función del número de filetes por pulgada, de acuerdo con las siguientes relaciones:
 - Paso en pulgadas = $1/\text{número de filetes por pulgada}$.
 - Paso en milímetros = $25,4/\text{número de filetes por pulgada}$.
- **Avance:** Es el desplazamiento medido para una vuelta completa de la rosca sobre el elemento de unión. En la rosca de una sola entrada, el avance es igual al paso; en roscas de dos entradas el avance es igual al doble del paso.
- **Diámetro mayor:** Se le conoce también como diámetro exterior y nominal de la rosca. Es el diámetro máximo del filete del tornillo o de la tuerca.
- **Diámetro menor:** También conocido como diámetro interior, del núcleo o de raíz. Es el diámetro mínimo del filete del tornillo o de la tuerca.
- **Diámetro primitivo:** Es el diámetro de un cilindro imaginario cuya superficie corta los filetes en puntos tales que resulten iguales al ancho de los mismos y al de los hoyos cortados por la superficie de dicho cilindro.

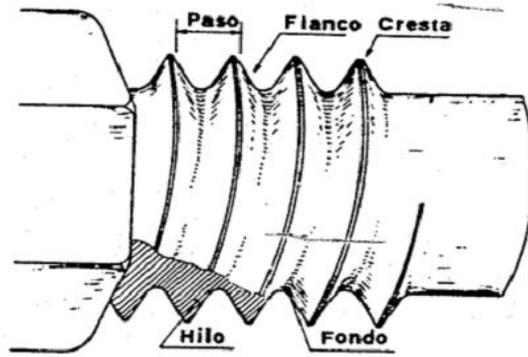


Fig. 1 Perfil de una rosca.

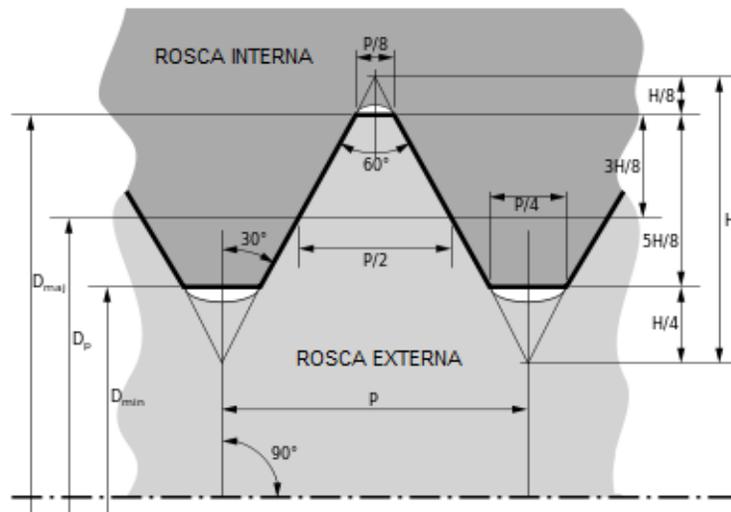


Fig. 2 Perfil de la una rosca más detallado.

Dónde:

Dmaj: Diámetro mayor.

Dmin: Diámetro menor.

Dp: Diámetro primitivo.

p: Paso.

H: Altura.

1.1.3. Clasificación de la rosca

Las roscas pueden tener diferentes tipos de clasificación (1).

Según el número de filetes:

- Roscas de una sola entrada, que tienen un filete.
- Roscas de varias entradas, con varios filetes.

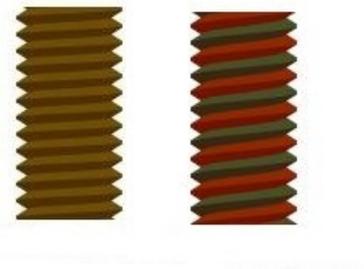


Fig. 3 Izquierda: rosca de una sola entrada. Derecha: rosca de varias entradas.

Según su posición las roscas se clasifican en:

- Roscas exteriores.
- Roscas interiores.



Fig. 4 Izquierda: rosca exterior. Derecha: rosca interior.

Según la forma del filete pueden ser: (2)

- **Roscas triangulares:** recibe este nombre cuando el prisma o filete que engendra la rosca tiene su sección parecida a un triángulo. Es la más utilizada en la industria, por destinarse a la sujeción de piezas.
- **Roscas trapeziales:** Es la engendrada por un filete cuya sección es un trapecio isósceles. Se emplea mucho en husillos de máquinas y herramientas, para conseguir movimientos de translación.

- **Roscas cuadradas:** Esta rosca posee filete de sección cuadrada. No está normalizada, por lo que en la actualidad tiende a desaparecer.
- **Roscas redondas:** Esta rosca es utilizada en husillos que tengan que soportar esfuerzos grandes y bruscos. Es una rosca de buenas condiciones mecánicas, pero de difícil elaboración.
- **Roscas de diente de sierra:** Tiene un filete cuya sección es aproximadamente un trapecio rectángulo. Rosca de difícil elaboración, pero muy resistente a los esfuerzos axiales en un solo sentido. Es muy utilizada en artillería y prensas.

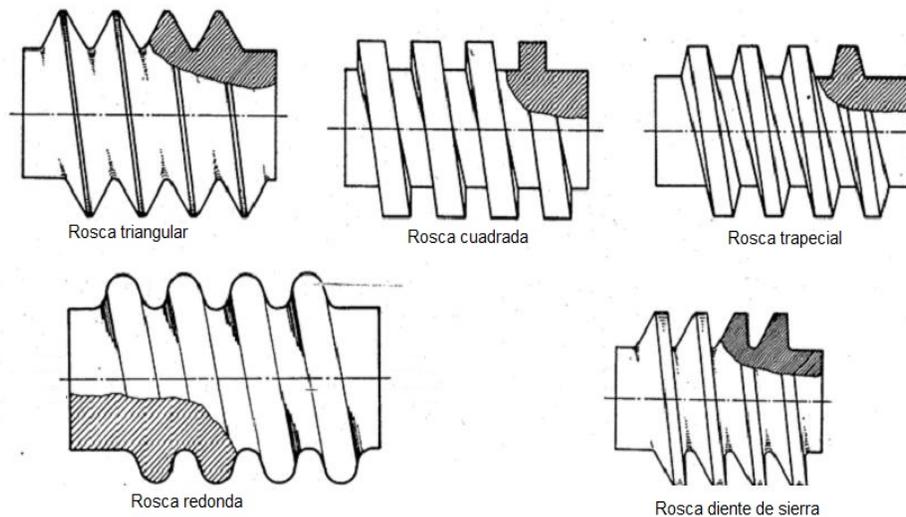


Fig. 5 Tipos de roscas según la forma de su filete.

Según su sentido se dividen en:

- Rosca a derecha cuando avanza o gira en sentido de las manecillas del reloj.
- Rosca a izquierda cuando avanza o gira en sentido contrario a las manecillas del reloj.

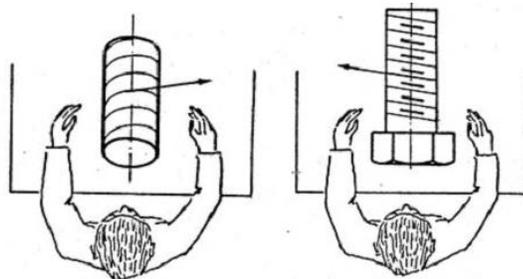


Fig. 6 Izquierda: rosca a derecha. Derecha: rosca a izquierda.

1.1.4. Sistemas de roscas

En la industria se utiliza una amplia variedad de tipos de roscas. Para lograr una estandarización en el uso de las mismas se ha procurado en diversos países normalizar las roscas, en otras palabras, darles dimensiones exactas y clasificarlas según su forma, utilidad y aplicaciones; dentro de cada uno de estas clasificaciones se establecen una serie de proporciones y medidas convenientemente escalonadas para que puedan cubrir las necesidades más comunes.

Se llama Sistema de Roscas a cada uno de los grupos en que se pueden clasificar las roscas normalizadas con especificaciones o reglas que deben cumplir (1). Estas reglas hacen referencia a los siguientes puntos:

- Forma y proporciones del prisma o filete.
- Escalonamiento de los diversos diámetros.
- Paso que corresponde a cada uno de los diámetros.
- Tolerancias² que se admiten en las medidas.

Algunos de los principales sistemas de roscas utilizados son:

- Roscas de Sujeción.
- Rosca *Whitworth*.
- Sistema Internacional (S.I).
- Sistema *Sellers* o *United States Standard* (U.S.S).
- Sistemas de roscas finas.
- Roscado *Acme*.
- Roscado Cuadrado.
- Roscado redondo normalizado.

Entre estos sistemas de roscas se destaca el Sistema Internacional puesto casi toda la tornillería actual que satisface a la mayoría de las uniones para tornillos y tuercas es utilizado por este sistema. Además es uno de los sistemas de roscado más extendido actualmente (1).

² Margen de error que admiten las medidas.

1.2. SISTEMAS PROPIETARIOS Y LIBRES CON MÓDULOS PARA LA CREACIÓN DE ROSCAS

El desarrollo rápido de las tecnologías hace que empresas de todo el mundo creen constantemente productos competitivos e innovadores, razón por la cual la ingeniería, el diseño y la arquitectura de estos productos son cada vez más exigentes. Los sistemas desarrollados por grandes industrias dedicadas al diseño de piezas mecánicas pueden ser tanto libres como propietarios. El análisis realizado en este epígrafe tiene en cuenta los software propietarios líderes del mercado y con altos niveles de aceptación y algunos software libres utilizados en la industria.

Los software propietarios seleccionados para su estudio son los siguientes:

- *Autodesk Inventor.*
- *SolidWork.*
- *CATIA.*
- *VariCAD.*

Los sistemas de diseño de piezas mecánicas basados en tecnologías libres que se seleccionaron son los siguientes:

- *Salome Geometric.*
- *FreeCAD.*
- *BRL-CAD.*

1.2.1. Autodesk Inventor

Autodesk Inventor es un paquete de modelado paramétrico de sólidos en 3D para la creación de prototipos digitales utilizados en el diseño, la visualización y la simulación de productos. Producido por la empresa de *software Autodesk*, se agregó a las Series de Diseño Mecánico de *Autodesk* como una respuesta de la empresa a la creciente migración de su base de clientes de diseño mecánico en 2D hacia la competencia, permitiendo que las computadoras personales ordinarias puedan construir y probar montajes de modelos extensos y complejos. *Autodesk Inventor* se basa en técnicas de modelado paramétrico y se utiliza en el diseño de ingeniería para producir y perfeccionar productos nuevos (4).

Su modelador geométrico permite modelar geometrías, dimensiones y materiales de manera que si se alteran las dimensiones, la geometría se actualiza automáticamente basándose en las nuevas dimensiones. Los bloques de construcción cruciales de Inventor son las partes. Estas se crean definiendo las características (*feature*), que a su vez se basan en bocetos o dibujos en 2D. Luego aplicando a estos bocetos operaciones de creación de sólidos como son la extrusión³, se convierte el boceto en un sólido. La ventaja de este diseño es que todos los bocetos y las características se pueden corregir más adelante, sin tener que hacer de nuevo la partición entera. Este sistema de modelado es mucho más intuitivo que en ambientes antiguos de modelado, en los que para cambiar dimensiones básicas era necesario generalmente suprimir el archivo entero y comenzar de cero. Como parte final del proceso de modelado, las partes se conectan para hacer ensamblajes. Los ensamblajes pueden consistir en piezas u otros ensamblajes. Las piezas se ensamblan agregando restricciones entre las superficies, bordes, planos, puntos y ejes. Algunas de estas restricciones incluyen coincidencia, nivelación, inserción, ángulo, tangente, transicional, movimiento y sistema de coordenadas de usuario (4).

Autodesk Inventor cuenta con un módulo para la gestión de roscas, intuitivo y cómodo para el especialista. Para comenzar con la construcción de la rosca, se debe tener como base una superficie cilíndrica o cónica ya sea exterior o interior, sobre la cual se aplicará la rosca que se pretende crear. Luego de haber seleccionado la superficie, el *software* permite seleccionar, en una ventana de diálogo que muestra al usuario, toda una serie de parámetros imprescindibles para la creación de la rosca. Si todos los parámetros están correctamente seleccionados, automáticamente *Inventor* genera, sobre la superficie seleccionada una textura ajustada a los parámetros seleccionados e indicando de esta forma la existencia de una rosca en la superficie cilíndrica seleccionada (4).

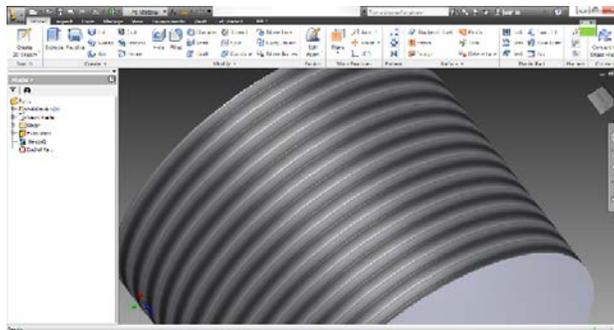


Fig. 7 Visualización de una rosca con *Autodesk Inventor*.

³ Operación 3D que genera un prisma a partir de una cara base.

1.2.2. SolidWork

SolidWork es un programa de diseño asistido por computadora para modelado mecánico desarrollado en la actualidad por *SolidWork Corp.*, para el sistema operativo *Microsoft Windows*. Es un modelador de sólidos paramétrico que permite modelar piezas y conjuntos, y extraer de ellos tanto planos como otro tipo de información necesaria para la producción. Posteriormente todas las extracciones (planos y ficheros de intercambio) se realizan de manera automatizada (5).

La construcción de un modelo en *SolidWork* por lo general comienza con un boceto 2D. El boceto puede contener objetos geométricos a los cuales se les agregan dimensiones para definir el tamaño y la ubicación de los mismos dentro del boceto. A estos objetos geométricos se le pueden definir atributos mediante operaciones llamadas relaciones. La naturaleza paramétrica de *SolidWork* significa que las dimensiones en el dibujo pueden ser controlados independientemente, o por relaciones con otros parámetros dentro o fuera del boceto. Finalmente, los dibujos o bocetos pueden ser transformados a sólidos mediante operaciones de construcción de sólidos. Las vistas se generan automáticamente a partir del modelo sólido, y las notas, dimensiones y tolerancias se pueden agregar fácilmente a los dibujos cuando sea necesario (5).

La creación de una rosca con *SolidWork* es un proceso sencillo y cómodo, el *software* permite insertar desde el menú de la barra de herramientas una espiral o “*hélix*” y a partir de la misma comenzar con la construcción de la rosca. Esbozando un cilindro, el cual debe tener un diámetro igual a la hélice construida anteriormente, este se convertirá en el núcleo de dicha rosca. Si todo está correcto hasta este punto, el *software* muestra una ventana para la selección de algunos parámetros como la altura de la rosca y el número de revoluciones que tendrá la misma. Después es necesario seleccionar el ángulo de inicio, *SolidWork* también permite al usuario la selección de dicho parámetro. Luego, para la terminación de dicha rosca es necesario definir en el punto inicial de la espiral la forma a la cual se le aplicará la operación de “barrido” sobre la hélice creada. De maneras muy similares se pueden crear tanto roscas internas como externas, aunque *SolidWork* propone otros métodos un poco más complejos para la creación de roscas, todo depende de la exactitud y el nivel de realismo que se desea obtener con la rosca que se diseña. (6).

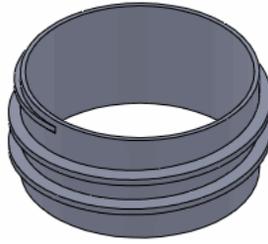


Fig. 8 Visualización de una rosca con *SolidWork*.

1.2.3. CATIA

CATIA es un *software* CAD que ofrece funcionalidades avanzadas, como ingeniería inversa, modelado de superficies, propagación rápida de las modificaciones del diseño, herramientas de diagnóstico en tiempo real y visualización de gama alta (7).

CATIA ofrece un ambiente productivo de diseño 2D que ofrece bocetos con una dinámica variacional y la creación de geometrías explícitas. Esto incluye todos los elementos necesarios para crear vistas 2D (puntos, líneas, círculos, elipses, splines, rectángulos, arcos y contornos 2D). Además el *software* propone algunos contornos predefinidos para acelerar la productividad (ranura, hexágono, rectángulo). También presenta diferentes tipos de superficies (cilindro, cono, esfera, toroide, plano) y de curvas (curvas resueltas, aristas, arcos, *P-lines*) (8). También posee una transición fácil y sin problemas para metodologías de diseño basadas en 2D a 3D. La geometría 2D pueden ser reutilizada para generar planos 3D copiando y pegando como un contorno 2D. Las vistas generadas a partir de 3D permiten combinarse con vistas 2D (7).

Teniendo como base una superficie cilíndrica previamente creada, se puede aplicar una rosca sobre la misma. *CATIA* permite una serie de operaciones para la construcción de una rosca. En la superficie cilíndrica seleccionada se debe crear un eje y a partir del mismo se construye una espiral o hélice la cual se modifica posteriormente cuando se le brinda la opción al usuario de seleccionar los parámetros pertinentes como la altura, el paso, la cantidad de revoluciones y la orientación de la misma. Teniendo todos estos datos el *software* permite ajustar el ángulo y la curva central de la rosca y aplicando entonces la operación antes mencionada se genera, de manera geométrica y ajustada a los parámetros seleccionados, una rosca en la superficie cilíndrica creada en un inicio.

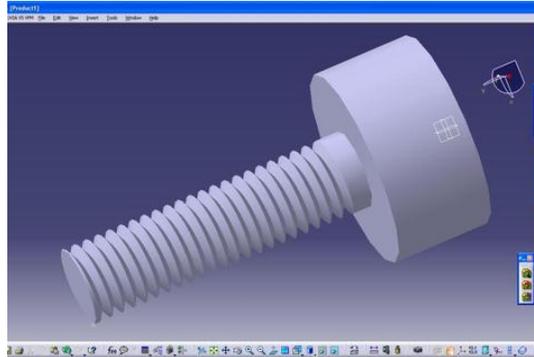


Fig. 9 Visualización de una rosca con CATIA.

1.2.4. VariCAD

VariCAD es un *software* CAD 2D / 3D pensado principalmente para el diseño de ingeniería mecánica proporcionando herramientas para actualizaciones 2D después de los cambios de los modelos 3D (9).

VariCAD basa su modelación en la construcción de bocetos 2D a partir de entidades básicas como puntos, líneas, arcos, entre otros. A partir de estos bocetos con operaciones 3D de construcción como el *extrude* o la revolución se pueden obtener sólidos a los cuales se les pueden aplicar diferentes operaciones como biselado, creación de agujeros, operaciones booleanas entre otros. Para el diseño 2D también se utiliza restricciones además de poder quitar grados de libertad a determinados objetos (9).

Las roscas en *VariCAD* se pueden crear sobre una superficie cilíndrica exterior ya existente, como un orificio o mediante la inserción de un cilindro. Para esto se comienza por crear el hilo o hélice de la rosca. Si los hilos están creados correctamente se pasa a la selección de un tipo de rosca, para lo cual se muestra al usuario una ventana de diálogo permitiendo la selección del tipo de rosca deseado. *VariCAD* presenta tres operaciones para la creación de roscas y estas son: Roscado del Agujero, Cilindro Roscado (para la creación de tornillo), y Rosca Exterior. Aplicando una de estas tres operaciones a la hélice definida se crea la rosca sobre dichos hilos (10).

1.2.5. Salome Geometric

Salome es un *software* de código abierto que provee una plataforma genérica para pre y post procesamiento para la simulación. Se basa en una arquitectura abierta y flexible de

componentes reutilizables. Es una solución multiplataforma y se distribuye como *software* de código abierto bajo los términos de la licencia GNU LGPL. Soporta la interoperabilidad entre *software CAD* de modelado y cálculo (*CAD-CAE* enlace). Facilita la integración de nuevos componentes en sistemas heterogéneos para el cálculo numérico (11).

Salome posee un módulo destinado a la gestión de las operaciones de modelado geométrico, este módulo se llama *Geometry Module (GEOM)*. Este componente proporciona funciones versátiles para la creación, visualización y modificación de los modelos geométricos de *CAD*. Este módulo permite la realización de varias operaciones como son la visualización de modelos en 3D, y dentro de este la selección, el cambio del color de un modelo, entre otros. También se gestiona la creación de objetos geométricos básicos como son el punto, la línea, el círculo, el arco, un vector, un plano de trabajo, un sistema de coordenadas local, entre otros. Permite la creación de primitivas 3D básicas como el cubo, el cilindro o la esfera. Además permite realizar las operaciones de modelado tales como la extrusión, la revolución, el relleno de superficie con aristas entre otras. Permite aplicar operaciones booleanas a los objetos como la fusión, cortar y la sección, y además las operaciones de transformación de los objetos como la rotación, la escalada, la multi-traduccion y la multi-rotación (12). A pesar de que *Salome Geometric* es un *software* para el diseño de piezas mecánicas, no cuenta entre sus módulos con uno dedicado a la gestión de roscas en superficies cilíndricas.

1.2.6. FreeCAD

FreeCAD es un modelador 3D *CAD* de propósito general basado en tecnologías libres (GPL y licencia LGPL). Está dirigido a la ingeniería mecánica y diseño de los productos, pero también se ajusta a una gama más amplia de aplicaciones alrededor de la ingeniería, como la arquitectura o de otras especialidades de la ingeniería. *FreeCAD* está basado en un modelador paramétrico con una arquitectura de *software* modular, lo que hace que sea fácil para adicionar funcionalidades sin modificar el núcleo del sistema (13).

El modelado geométrico con *FreeCAD* permite dibujar rápidamente simples objetos 2D y ofrece varias herramientas para su modificación. También proporciona un sistema completo de ajuste, y varias utilidades para administrar los objetos y escenarios. Además el *software* permite la construcción de primitivas como: el cubo, el cono, el cilindro, la esfera, especificando en cada uno de ellos las dimensiones. *FreeCAD* posee herramientas para la creación de objetos como son: línea, círculo, arco, rectángulo, polígono entre otros y herramientas para la modificación de

los objetos antes mencionados, entre estas herramientas están: la intersección, el recorte, la extrusión, la rotación, el barrido, entre otras (14).

FreeCAD presenta un comando especial llamado “Barrido Helicoidal” que está disponible para la creación de roscas. Esta operación es una opción ampliamente utilizada para hacer cortes de roscas. Esta opción también podría ser utilizada para la fabricación de muelles helicoidales. Una vez que se selecciona la operación de barrido, se muestra un cuadro de diálogo para la selección de algunas variables necesarias para la ejecución de la operación. Entonces es necesario definir la trayectoria y la sección sobre la cual se va a ejecutar la operación, se ejecuta la operación de barrido y como resultado se genera una rosca de manera geométrica y ajustada a las medidas definidas anteriormente (15).

1.2.7. BRL-CAD

BRL-CAD es un *software* para el modelado sólido y diseño asistido por computadora (*CAD*). Incluye un editor interactivo de geometría, trazado de rayos de apoyo para la representación gráfica y el análisis geométrico. El paquete entero se distribuye en código fuente y de forma binaria (16).

BRL-CAD soporta simultáneamente dos métodos de interacción, uno con una línea de comandos y el otro con una interfaz gráfica de usuario. Aunque se ha utilizado para una amplia variedad de aplicaciones de ingeniería y gráficos, el propósito principal del paquete sigue siendo el soporte balístico y análisis electromagnéticos. Un punto fuerte reside en su capacidad para construir y analizar modelos realistas de objetos complejos utilizando un conjunto pequeño de formas primitivas. Para ello, se emplean operaciones básicas booleanas de unión, substracción e intersección. *BRL-CAD* también propone el modelado de geometrías sólidas constructivas el cual se centra en todo el volumen y el contenido de los objetos (16).

Las bibliotecas *BRL-CAD* están diseñadas principalmente para el modelador geométrico. Cada biblioteca está diseñada para un propósito específico: la creación, edición, trazado de rayos, geometría y la manipulación de imágenes. La aplicación también ofrece una serie de herramientas y utilidades que se ocupan principalmente de la conversión geométrica, la conversión de formato de imagen y la línea de comandos orientados a la manipulación de imágenes. *BRL-CAD* no cuenta con un módulo para la gestión y visualización de roscas (16).

1.3. ELEMENTOS DE *OPENCASCADE*

El sistema GALBA-CAD está implementado bajo la tecnología *OpenCASCADE* la cual es una biblioteca de código abierto, en lenguaje C++, que consiste en clases y paquetes que ofrece soluciones en el área de:

- Modelado de superficies y sólidos: para modelar cualquier tipo de objeto.
- Visualización en 3D y 2D: mostrar y animar objetos.
- El intercambio de datos: importar y exportar formatos.
- Desarrollo rápido de aplicaciones: para manipular los datos de aplicaciones personalizadas.

OpenCASCADE está diseñado para el desarrollo industrial y el modelado 3D, la simulación numérica y aplicaciones de visualización que requieren una garantía de calidad y fiabilidad. Las bibliotecas de la tecnología *OpenCASCADE* se distribuyen en código abierto para plataformas múltiples y permite el uso comercial incondicional del código. Cada módulo contiene varias bibliotecas, cada biblioteca contiene clases agrupadas en paquetes, como se puede observar en la siguiente figura (17).

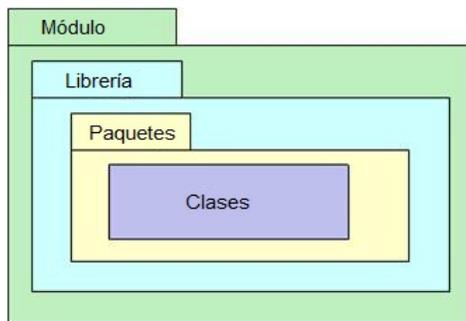


Fig. 10 Estructuras por paquetes de *OpenCASCADE*.

Las ventajas de usar *OpenCASCADE* como tecnología de modelado en 3D, son (18):

- El acceso al código fuente.
- No hay derechos de licencia.
- La continuidad del producto.
- La confiabilidad.

OpenCASCADE está estructurado por módulos, los cuales poseen todos los paquetes y clases. Cada uno de estos módulos principales posee una función específica, y cada uno trabaja sobre un área determinada. Los principales módulos de *OpenCASCADE* son los siguientes:

- *Foundation Classes.*
- *Modeling Data.*
- *Modeling Algorithms.*
- *Visualization.*
- *Application Framework (OCAF).*
- *Data Exchange.*

1.3.1. Manipulación de geometrías

La tecnología *OpenCASCADE* contiene tres paquetes fundamentales para el manipulación de las geometrías: Utilidades Geométricas (*Geometry Utilities*), Geometría 2D (*2D Geometry*) y Geometría 3D (*3D Geometry*). Cada una de ellas contiene diferentes paquetes los cuales brindan una serie de servicios para facilitar el manejo y construcción de las entidades geométricas.

➤ **Utilidades Geométricas (*Geometry Utilities*)**

El paquete *Geometry Utilities* proporciona clases y algoritmos para los siguientes servicios:

- Creación de formas mediante la interpolación y aproximación.
- La construcción directa de formas.
- Cálculo de las coordenadas de los puntos en las curvas 2D y 3D.
- Cálculo de extremos entre las formas.

➤ **Geometría 2D (*2D Geometry*)**

El paquete *Geom2d* contiene algoritmos para la construcción de geometrías del paquete *gp* (geometría básica) en el espacio 2D. Dichos objetos son no persistentes y se guardan por referencia en lugar de ser guardados por valor. Proporciona además el paquete *TColGeom2d*, que ofrece instancias de matrices unidimensionales e instancias de secuencias de curvas del paquete *Geom2d*. Todos estos objetos están disponibles tanto para el almacenamiento por referencia como por valor (19).

➤ **Geometría 3D (3D Geometry)**

OpenCASCADE ofrece en el paquete *3D Geometry* una serie de estructuras de datos para objetos en tres dimensiones y las propiedades locales de estos objetos. Contiene el paquete *Geom*, el cual define los objetos geométricos no persistentes y almacenados por referencia en el espacio 3D y contiene todas las funciones básicas para las transformaciones geométricas como la identidad, rotación, traslación, espejo, escala, entre otros. Además contiene el paquete *TCo/Geom*, el cual ofrece instancias de matrices bidimensionales e instancias de secuencias de curvas del paquete *Geom*. Todos estos objetos están disponibles tanto para el manejo por referencia como por valores (19).

1.3.2. Manipulación de topologías

Una topología se considera como el límite (en términos de frontera) de un objeto geométrico, el uso principal de las topologías es para la descripción de las restricciones de un objeto dado y para describir cómo los objetos están conectados entre sí. Las entidades topológicas son llamadas formas (*shape*) (19). Una forma está definida por:

- Un sistema de coordenadas locales.
- Un objeto *TShape*.
- Una orientación.

El sistema de coordenadas locales se utiliza para describir la ubicación de una forma topológica en diferentes posiciones a partir de la posición en la cual fue definida.

Un objeto *TShape* es un puntero que describe el objeto en su sistema de coordenadas predeterminado. Este objeto nunca se utiliza directamente, es manejado a través de la clase *TopoDS_Shape* la cual se proporciona en el paquete *TopoDS*. Dicha clase se especializa en varias clases que son las encargadas de manejar cada una de las formas topológicas existentes, las cuales son: (19)

- Compuesto (*Compound*): Grupo de cualquier tipo de objetos topológicos.
- Sólido compuesto (*CompSolid*): Conjunto de sólidos conectados por sus caras.
- Sólido (*Solid*): Una parte del espacio limitado por las caras. Tiene tres dimensiones.
- Cáscara o Caparazón (*Shell*): Conjunto de caras conectadas por sus bordes que puede ser abierta o cerrada.

- Cara (*Face*): En 2D es parte de un plano; en 3D que es parte de una superficie. Tiene dos dimensiones.
- Alambre (*Wire*): Un conjunto de aristas conectadas por sus vértices que puede ser un contorno abierto o cerrado dependiendo de si los bordes están unidos o no.
- Borde (*Edge*): Un elemento topológico correspondiente a una curva contenida. Un borde se limita generalmente por vértices. Tiene una dimensión.
- Vértice (*Vertex*): Un elemento topológico correspondiente a un punto. Tiene dimensión cero.

La orientación es una noción generalizada del sentido de la dirección. Esto se utiliza cuando una forma limita con un dominio geométrico y es estrechamente ligada a la noción de límite. Esto solo sucede en los tres casos siguientes:

- Curva limitada por un vértice.
- Superficie limitada por un borde.
- Espacio limitado por una cara.

Entre las entidades topológicas existe la conectividad entre las formas. El concepto de conectividad plantea que dos formas topológicas pueden conectarse si ellas comparten una misma sub-forma. Un ejemplo de conectividad es el siguiente gráfico:

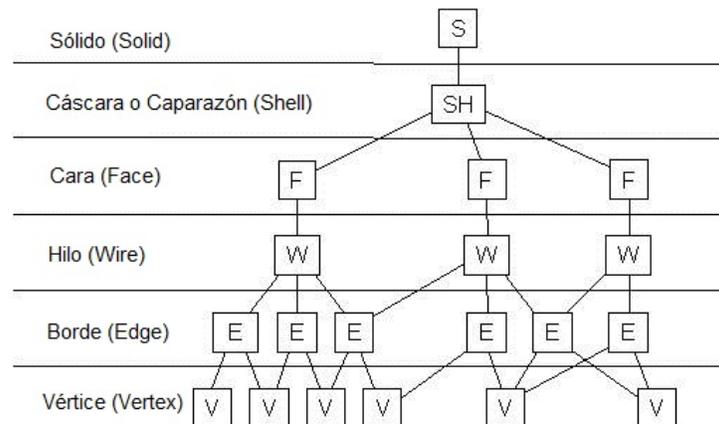


Fig. 11 Conectividad entre formas topológicas.

Otra de las formas de representar una topología es mediante la representación de fronteras con el paquete *BRep* (*boundary representation*). *BRep* da una descripción completa de un objeto mediante la asociación de la información topológica y geométrica.

1.3.3. Visualización de objetos

La visualización se gestiona a través de los componentes de presentación y selección. Para hacer que la gestión de estas funcionalidades sea más intuitiva y transparente *OpenCASCADE* proporciona los servicios de aplicaciones interactivas (AIS por sus siglas en inglés). Además *OpenCASCADE* contiene clases y paquetes para implementar objetos en tres dimensiones que puedan presentarse y seleccionarse, además de clases y algoritmos genéricos para implementar dicha presentación (20).

➤ Estructura de una presentación

La visualización con la tecnología *OpenCASCADE* de un objeto en la pantalla consta de tres tipos de entidades: (20)

- Un objeto presentable (*AIS_InteractiveObject*): suministrar la estructura gráfica del objeto a ser presentado.
- Un visor: permite manipular interactivamente vistas del objeto.
- Un contexto interactivo (*AIS_InteractiveContext*): controla todo el proceso desde la presentación.

➤ El principio de selección

Los objetos que se pueden seleccionar gráficamente se muestran como conjuntos de primitivas sensibles, que proporcionan zonas sensibles en el espacio gráfico 2D. Estas zonas están ordenadas de acuerdo con su posición en la pantalla cuando se inicia el proceso de selección. La posición del ratón está también asociada a una zona sensible. Cuando se mueve dentro de la ventana donde se muestran los objetos, las áreas tocadas por la zona del ratón se analizan. Los objetos contenidos en estas zonas se resaltan o selecciona.

➤ Servicios de Aplicaciones Interactivas (AIS)

Las Aplicaciones de Servicios Interactivas (AIS por sus siglas en inglés) ofrecen un conjunto de servicios generales más allá de los que ofrecen los paquetes básicos de selección y presentación. AIS permite manejar las presentaciones y selecciones de manera dinámica en un visor simple.

Una de estas clases fundamentales del paquete AIS es *AIS_InteractiveObject*, la cual define una clase de objetos con servicios de visualización y selección. Las entidades que se visualizan y seleccionan son objetos interactivos. Otra de las clases fundamentales es *AIS_InteractiveContext* la cual permite gestionar el comportamiento gráfico y la selección de objetos interactivos en uno o más visores.

1.3.4. Marco de trabajo para el desarrollo de aplicaciones

Dentro de los módulos fundamentales de *OpenCASCADE* se encuentra el marco de trabajo para el desarrollo de aplicaciones (*Application Framework*) conocido como *OCAF* por sus siglas en inglés. Este módulo propone una herramienta para el desarrollo rápido de aplicaciones, proporcionando una arquitectura basada en el modelo aplicación / documento. Esta arquitectura consiste de bibliotecas de clases desarrolladas en C++. La clase principal de aplicación, está a cargo de la gestión de documentos durante la sesión de trabajo. Los servicios que brinda esta clase incluyen (21):

- Creación de nuevos documentos.
- Abrir y salvar documentos.
- La inicialización de vistas de documentos.

El documento es el contenedor de los datos de la aplicación. En *OCAF*, las referencias se implementan en forma de etiquetas. Los datos de la aplicación se adjuntan a estas etiquetas como atributos. Por medio de estas etiquetas y una estructura de árbol se mantienen unidas las referencias de un modelo (en la forma de las etiquetas) en un único recipiente denominado documento.

Los documentos ofrecen acceso a la estructura de datos y la gestión de los siguientes elementos (21):

- Gestionar la notificación de los cambios.
- Actualización de los enlaces externos.
- Gestionar el ahorro y la restauración de los datos.
- Almacenar los nombres de extensiones de *software*.
- Administrar transacciones de comandos.
- Administrar opciones Deshacer y Rehacer.

Dentro de un documento hay un conjunto de etiquetas organizadas en una estructura en forma de árbol, las mismas tienen las siguientes características (21):

- La primera etiqueta de un documento es la raíz del árbol.
- Cada etiqueta tiene un identificador (más conocido como *tag*) el cual es un valor entero.
- Las sub-etiquetas que se derivan de alguna etiqueta se denominan sus hijos.
- Cada etiqueta que no es raíz tiene un padre.
- Las etiquetas que tienen el mismo padre son hermanos.
- Etiquetas hermanos no pueden compartir el mismo identificador.
- Una etiqueta se define únicamente por una entrada expresada como una lista de identificadores de los padres desde la raíz: la lista de las etiquetas se escribe de derecha a izquierda: identificador de la etiqueta, identificador de su padre, identificador del padre de su padre, ..., hasta 0 (identificador de la etiqueta raíz) (21).

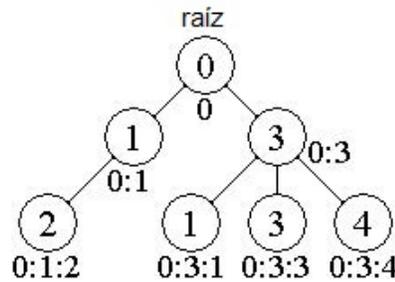


Fig. 12 Designación de los identificadores de las etiquetas.

Cada etiqueta puede tener una lista de datos que contienen atributos, y varios atributos pueden adjuntarse a una etiqueta. Cada atributo se identifica por un identificador y en una etiqueta no debe tener más de un atributo con el mismo identificador.

Los documentos agregan una lista de atributos, que representan datos típicos utilizados en sistemas CAD. Estos datos incluyen una amplia gama de atributos estándares correspondientes a los siguientes tipos (21):

- Atributos primitivos: como enteros, reales, cadenas, tipos de matrices, etc.
- Atributos *shape*: contiene la geometría de las formas, su referencia y evolución.
- Atributos geométricos: como puntos, ejes, planos, restricciones, etc.
- Atributos de función: para reconstruir los objetos después de que han sido modificados (parametrización de los modelos).

- Atributos de visualización: permiten que los datos se visualicen en una vista 2D o 3D.
- Atributos definidos por los usuarios: son creados y configurados por los usuarios para guardar datos específicos y personalizados.

Una de las ventajas de *OCAF* es que de todos los tipos de atributos se puede crear nuevas instancias de ellos, adjuntarlos y eliminarlos de etiquetas.

CONCLUSIONES DEL CAPÍTULO

En el desarrollo de este capítulo se realizó un estudio de varios términos y conceptos importantes para la realización de esta investigación. Teniendo en cuenta el estudio sobre las partes de la rosca y su clasificación se destaca a las roscas de perfil triangular como una de las más utilizadas en la industria por proporcionar mayor fijación y acoplamiento. A partir de los conceptos vistos sobre los sistemas roscados y sus características, el Sistema Internacional constituye el más utilizado dado que la mayoría de la tornillería actual se desarrolla bajo este sistema.

Considerando el estudio realizado sobre las características de los sistemas CAD líderes del mercado, se aprecia que la creación de la rosca mediante texturas que propone *Autodesk Inventor*, es más factible dado que el cual el usuario solo tiene que seleccionar los datos de la rosca y lo abstrae de todo el proceso de creación interno. En cuanto a la visualización de la rosca, a partir de los software analizados, se considera que si bien la visualización por geometría brinda un mayor nivel de realismo, el hecho de que el usuario tenga que definir elementos para la construcción de la rosca hace el proceso de creación engorroso especialmente para usuarios sin experiencia y sin dominio de aspectos matemáticos fundamentales. Se considera también que el realismo de la rosca no es una prioridad para el sistema GALBA-CAD, dado que si bien la geometría de la rosca se puede utilizar para establecer compatibilidad entre roscas externas e internas (dígase por ejemplo la rosca de un tornillo y la rosca de una tuerca), esta misma compatibilidad se puede determinar a través de los datos propios de la rosca. Tomando en cuenta las consideraciones anteriores se concluye que la visualización por textura que realiza *Autodesk Inventor*, es la más apropiada para el módulo a desarrollar.

Basado en el estudio realizado sobre la tecnología *OpenCASCADE* y sus elementos fundamentales dentro de los cuales se puede apreciar el Framework de Aplicaciones que propone la tecnología, se considera que el mismo proporciona una mejor organización de los

datos almacenándose en un estructura en forma de árbol además de la utilidad de comandos y operaciones como rehacer y deshacer que permiten una mejor utilidad del *software* desarrollado bajo esta tecnología.

CAPÍTULO 2: SOLUCIÓN PROPUESTA

En el presente capítulo se presenta la solución técnica del presente trabajo de diploma, contiene además la descripción de los requisitos funcionales y no funcionales, se describen los casos de uso con el objetivo de explicar con un mayor nivel de detalles lo que la aplicación debe hacer. Se presentan los diagramas realizados en la fase de diseño tales como diagrama de clases, modelo de dominio entre otros.

2.1. SOLUCIÓN PROPUESTA

A continuación se describe la solución propuesta separado en sub-epígrafes, donde se expone con más detalles aspectos tales como la descripción de normas y estándares para la fabricación de roscas, como se organizan los datos en la estructura de OCAF, la manera en que se logra la visualización de la rosca y como se logra establecer la comunicación con la base de datos.

2.1.1. Normas y estándares

En el marco del módulo que se desea desarrollar es necesario, para el diseño y posterior fabricación de una rosca, regirse por normas y estándares establecidos internacionalmente. El módulo desarrollado maneja estas normas mediante la conexión con la base de datos del sistema GALBA-CAD, la cual actualmente solo tiene incluida tres normas:

- *ISO Metric Profile.*
- *ISO Metric Trapezoidal Threads.*
- *ISO Pipe Threads.*

Si bien no son estas las únicas normas existentes, representan algunos de los tipos de roscas más comunes en la industria como las triangulares, las trapezoidales y las redondas. Se analizará en cada una de estas normas su designación y los valores asociados a esta, pues estos valores son los que brindaran la información necesaria para la creación y visualización de la rosca.

2.1.1.1. Rosca Métrica ISO (ISO Metric Profile)

El estándar ISO para roscas métricas se caracteriza por un hilo simétrico en forma de V, cuyos flancos forman un ángulo de 60° entre sí, y que es truncado⁴ en su parte superior en $1/8$ de la altura y en su parte inferior en $1/4$ de la altura. Una rosca métrica ISO se designa por la letra M seguida por el valor del diámetro nominal o diámetro mayor y el paso, ambas longitudes expresadas en milímetros y separados por el signo de multiplicación x. Una designación completa comprende además, las clases de tolerancias para el diámetro de la cresta y el diámetro del paso, ambos símbolos seguidos de una letra g o h que indica la posición de la tolerancia. Estas letras indican si la roca es interna cuando se expresan en mayúscula (G o H) y externa si se expresan en minúscula (g o h). Cuando los símbolos de las tolerancias del diámetro de la cresta y el paso sean idénticos, el símbolo no necesita repetirse y solo debe darse una vez (22).

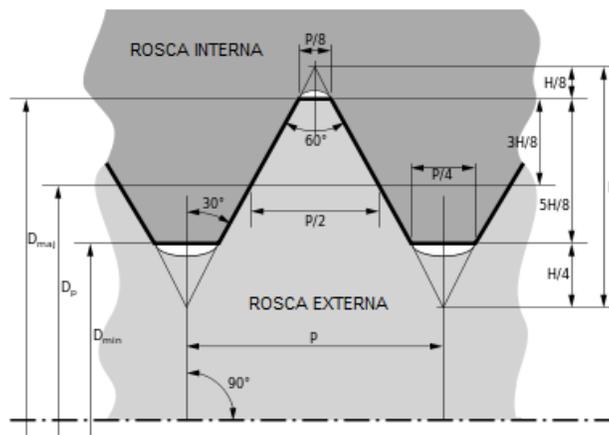


Fig. 13: Rosca Métrica ISO.



Fig. 14: Designación de la rosca métrica.

⁴ Que se corta a una distancia determinada en las bases superiores o inferiores o en ambas.

2.1.1.2. Rosca Métrica Trapezoidal ISO (*ISO Metric Trapezoidal Threads*)

En una rosca trapezoidal ISO, el ángulo de la rosca mide 30° y se designa por las letras TR seguida por el valor del diámetro nominal o diámetro mayor y el paso, ambas longitudes expresadas en milímetros y separados por el signo de multiplicación x. Una designación completa comprende además de la designación básica, la simbología LH para indicar que es una rosca con sentido a izquierda, cuando este no aparece se asume entonces que la rosca tiene sentido a derecha.

Cuando dicha designación no contiene ningún sufijo se asume que la rosca es de una sola entrada. Al contener un sufijo la designación cambia, siendo entonces que el número después del signo x es el avance de la rosca y el símbolo que se encuentra entre paréntesis es el paso. Dividiendo el avance entre el paso se obtiene la cantidad de entradas de la rosca (23).

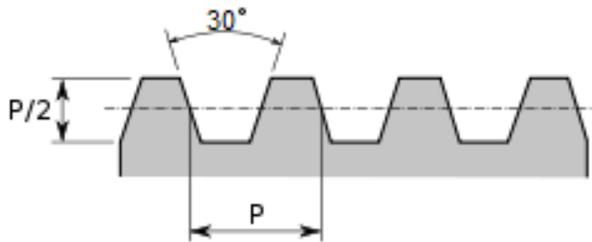
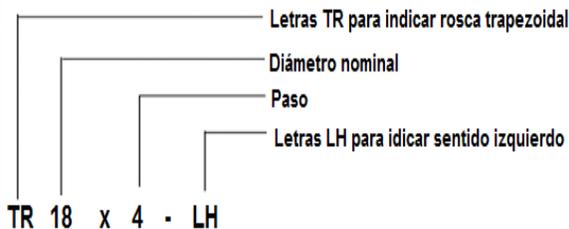


Fig. 15 Rosca trapezoidal.

Designación de rosca trapezoidal de una entrada



Designación de rosca trapezoidal para más de una entrada

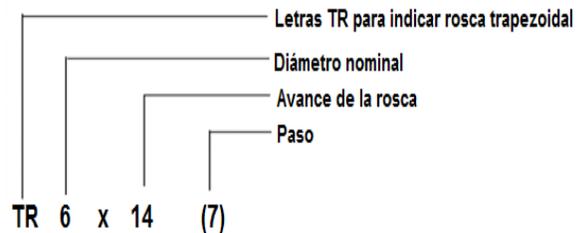


Fig. 16: Designación de la rosca trapezoidal.

2.1.1.3. Rosca de tubería ISO (*ISO Pipe Threads*)

Una rosca para tubos ISO se conforma de una hélice en forma de V truncada en $1/6$ de la altura en la base superior e inferior por arcos circulares que terminan tangencialmente en los flancos, el ángulo entre dichos flancos es de 55° . Este tipo de rosca se denota con la letra G, la cual indica roscas externas e internas de esta norma para el estándar ISO 228, luego le sigue el diámetro o tamaño nominal expresado a forma de fracción. Este diámetro no representa el diámetro nominal de la rosca sino el de la tubería sobre la cual se va a aplicar la rosca. Las roscas son normalmente en sentido a derecha, cuando la rosca es en sentido a izquierda se anexan a la designación las letras LH. El paso se expresa en pulgadas, y representa la cantidad de hilos en una pulgada. Para calcular dicho paso en milímetros, se utiliza la relación: Paso en milímetros = $25.4 /$ paso en pulgadas (24).

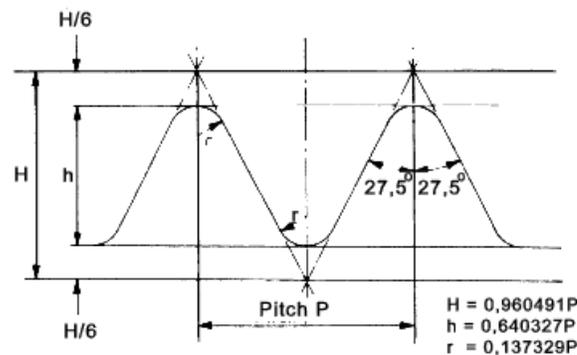


Fig. 17: Rosca de tuberías.

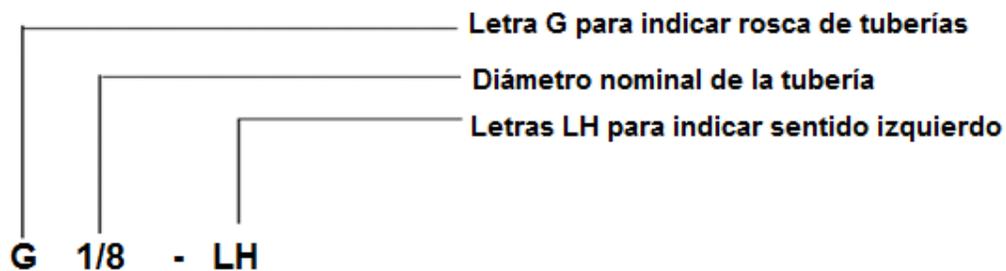


Fig. 18: Designación de la rosca de tuberías.

2.1.2. Organización y persistencia de los datos de la rosca

El sistema GALBA-CAD utiliza el enfoque basado en documentos como propone el *framework* de aplicación de *OpenCASCADE* para guardar la información de las entidades 2D y 3D, sólidos, bocetos entre otros elementos. Cada uno de estos elementos, contiene información que se representa en un árbol mediante etiquetas y atributos. Para proporcionar un acceso más sencillo y legible en el código fuente, así como garantizar la flexibilidad de la información de cada tipo de elemento se crearon nodos que representan etiquetas, sus etiquetas internas y correspondientes atributos.

Se definió también en el sistema GALBA-CAD el concepto de nodo contenedor que representa una etiqueta cuyos hijos son nodos del mismo tipo de elementos. Algunos ejemplos de nodos contenedores son el nodo contenedor de bocetos, operaciones 3D, para sólidos, entre otros. Para guardar la información correspondiente a la rosca en un documento del sistema, se crea un nodo con la estructura que se muestra en la siguiente figura:

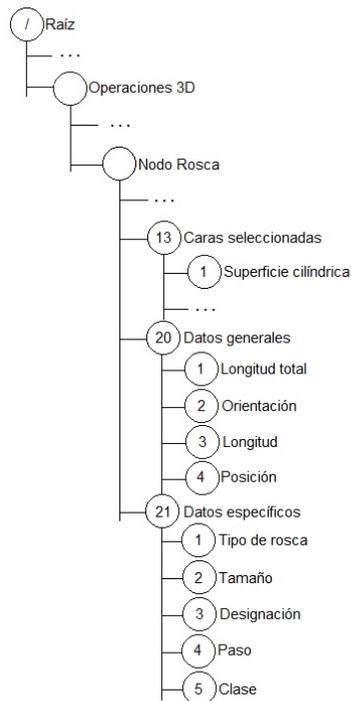


Fig. 19 Ubicación de los datos en el nodo correspondiente a la rosca.

Para poder organizar la información correspondiente a la rosca de una manera más eficiente se dividieron los datos de la misma en datos generales y datos específicos. Dentro de los datos

generales se encuentra la orientación y posición de la rosca así como la longitud donde se va aplicar la rosca o si se aplicará sobre toda la superficie (longitud total), estos datos son comunes para cualquier tipo de rosca y son guardados como nodos hijos del nodo denominado datos generales (ver Fig. 14).

Una especificación de una rosca en particular contiene mucha más información que la que se maneja en la aplicación como datos específicos, pero solo se necesitan utilizar estos datos (tipo, tamaño, designación, paso, clase) pues a partir de ellos se puede identificar una rosca en específico dentro de los valores establecidos en las normas y con esto se puede acceder al resto de la información que brinda la especificación. Estos datos antes mencionados también se guardan como nodos hijos del nodo denominado datos específicos (ver Fig. 14)

2.1.3. Visualización de la rosca

Para la visualización de la rosca se propone la utilización de una textura previamente elaborada, que representa un paso de la rosca a visualizar. La visualización completa se logra con la repetición de la textura antes mencionada en el parámetro V de la superficie cilíndrica parametrizada sobre la cual quedará visualizada la rosca.

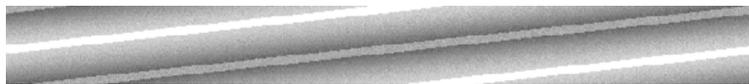


Fig. 20: Textura original que representa un paso.

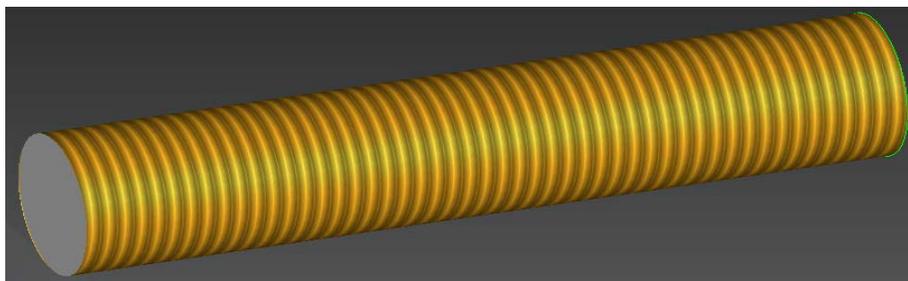


Fig. 21: Textura original visualizada después de las repeticiones.

La cantidad de repeticiones que deben realizarse de la textura original dependerá de la altura de la superficie y del paso de la rosca que se esté creando. Dicha cantidad de repeticiones puede calcularse de la siguiente manera:

$$\text{Repetición en V} = \text{Altura de la superficie} / \text{Paso de la rosca}$$

La altura de la superficie cilíndrica en la cual se va a crear la rosca y que se utiliza en la relación anterior se calcula utilizando el parámetro V de la superficie parametrizada (ver Fig. 17) en cuestión de acuerdo a la siguiente expresión:

$$\text{Altura de la superficie} = \text{Ultimo parámetro de V} - \text{Primer parámetro de V}$$

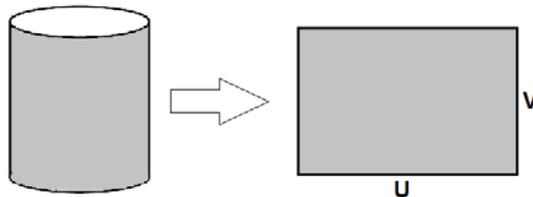


Fig. 22: Representación de parámetros U y V de la superficie lateral de un cilindro.

Para el cálculo de la altura se utiliza la API de *OpenCASCADE BRepAdaptor_Surface*, que permite obtener una superficie geométrica parametrizada (*Geom_Surface*) a partir de una topología en forma de cara (*TopoDS_Face*), así como acceder a los valores de los parámetros U y V de la superficie geométrica. A continuación se presenta un fragmento de código en C++ para calcular la altura del cilindro:

```
BRepAdaptor_Surface surfAdaptor (cylFace);  
double fvp, lvp, cylHeight;  
fvp = surfAdaptor.FirstVParameter();  
lvp = surfAdaptor.LastVParameter();  
cylHeight = lvp - fvp;
```

Después de conformada la textura, esta se asocia a la superficie cilíndrica seleccionada mediante la creación de un objeto interactivo de *OpenCASCADE (AIS_TextureShape)*. Este objeto a su vez se relaciona con un atributo de visualización (*TPrsStd_AISPresentation*) que se guarda en el nodo correspondiente a la rosca mencionado en el epígrafe anterior. Para poder lograr esta relación, se crea un *driver* de visualización que se ejecuta cada vez que se indique la operación *Display()* al atributo. Este *driver* es guardado en la tabla de *driver* de la aplicación general y asociado al atributo de visualización de la rosca.

A continuación se presenta el algoritmo general que sigue el *driver* para la obtención del objeto interactivo *AIS_TextureShape* y su asociación con el atributo *TPrsStd_AISPresentation* de la rosca. Este algoritmo se implementa en el método *Update()* del *driver* mencionado anteriormente.

1. Actualización de la superficie cilíndrica existente mediante mecanismos de parametrización.
2. Ejecución del *TextureThreadBuilder*.
3. Obtener la textura generada para la rosca a partir del *builder* ejecutado.
4. Actualización del *AIS_InteractiveObject*, parámetro del método *Update()*, con la textura generada.

2.1.4. Conexión con la Base de Datos

Como se menciona al inicio de este capítulo, las normas y estándares proporcionan especificaciones para las roscas utilizadas en la fabricación de piezas mecánicas. Los datos correspondientes a las normas se publican en forma de tablas y para su uso por el sistema GALBA-CAD se almacenan en una base de datos. Para la elaboración de la misma se utiliza el gestor de bases de datos Postgres.

Dado que no se incluía en el alcance de la investigación el diseño y elaboración de la base de datos para roscas, el módulo propuesto se comunica con una base de datos existente del sistema GALBA-CAD. Para establecer la conexión se utiliza el driver de Qt, QPsql, para acceder a bases de datos creadas con Postgres y que se habilita en Ubuntu instalando la biblioteca *libqt-sql-psql*.

Para lograr una mayor flexibilidad y reutilización de los resultados de las consultas realizadas a la base de datos, se utiliza la arquitectura modelo/vista que propone el *framework* Qt. Esta arquitectura separa la forma en que se almacenan los datos de la forma en que se le presentan al usuario. Estos datos se almacenan en forma de modelos que pueden ser mostrados en varias vistas al mismo tiempo, lo cual posibilita una mayor flexibilidad y extensibilidad a los datos mencionados (25).

2.2. ESPECIFICACIÓN DE REQUISITOS

Hacer una identificación correcta de los requisitos funcionales que debe cumplir el sistema es muy importante. Una vez que se han definido los principales conceptos relacionados con el campo de acción, se definen los requerimientos funcionales y no funcionales que el módulo que se va a desarrollar debe cumplir.

2.2.1. Requisitos funcionales

Los requisitos funcionales son capacidades o condiciones que el sistema debe cumplir. El propósito fundamental de la captura de requisitos es guiar el desarrollo hacia un sistema que satisfaga las necesidades del cliente (26). De acuerdo con los objetivos propuestos, la aplicación debe ser capaz de:

RF 1: Seleccionar superficie cilíndrica.

RF 2: Especificar datos de la rosca.

RF 2.1: Seleccionar norma y estándares de la rosca a crear.

RF 2.2: Insertar o seleccionar parámetros correspondiente a la norma seleccionada.

RF 2.3: Establecer la longitud de la rosca.

RF 2.4: Previsualizar rosca al variar norma y parámetros.

RF 3: Crear y visualizar rosca en superficie cilíndrica seleccionada de acuerdo a la norma escogida y parámetros definidos.

RF 4: Editar rosca creada.

RF 4.1: Modificar norma y parámetros de la rosca seleccionada.

RF 5: Eliminar rosca creada.

2.2.2. Requisitos no funcionales

Los requisitos no funcionales son propiedades o cualidades que debe cumplir el producto. Debe pensarse en ellos como las características que hacen al producto atractivo, usable, rápido, entre otros posibles aspectos a tener en cuenta (26). La aplicación debe cumplir los siguientes requisitos:

Diseño e Implementación

- Debe ser implementado en lenguaje C++ estándar y bajo paradigmas de *software* libre.
- Se regirá por la filosofía de Programación Orientada a Objetos.

Software

- La aplicación debe ser desarrollada en la distribución de *GNU / Linux Debian*.
- Se debe utilizar *QTcreator* o *Eclipse CDT* como *IDE* de desarrollo y *QTDesigner* para el diseño de interfaces gráficas.

Apariencia o Interfaz Externa

- La aplicación debe tener una interfaz sencilla y amigable.
- Los botones expresarán su función, ya sea mediante su texto o la imagen que acompaña a este.

2.3. MODELO DE CASOS DE USO

2.3.1. Actor del sistema

Se considera actor del sistema a toda persona o sistema que interactúa con la aplicación y se beneficia con el resultado de dicha interacción. En el módulo desarrollado se detectó el siguiente actor:

Actor	Descripción
Especialista	Interactúa directamente con la aplicación. Establece y selecciona los datos necesarios para la ejecución de las operaciones.

Tabla 1: Descripción de actor del sistema.

2.3.2. Diagrama de Casos de Uso

Un diagrama de casos de usos permite a los desarrolladores y a los clientes llegar a un acuerdo sobre los requisitos, es decir, sobre las condiciones y posibilidades que debe cumplir el sistema. El diagrama de casos de uso contiene actores, casos de uso y las relaciones entre ellos, además proporciona la entrada fundamental para el análisis, el diseño, y las pruebas (26). El diagrama de casos de uso para la aplicación a desarrollar es el siguiente:

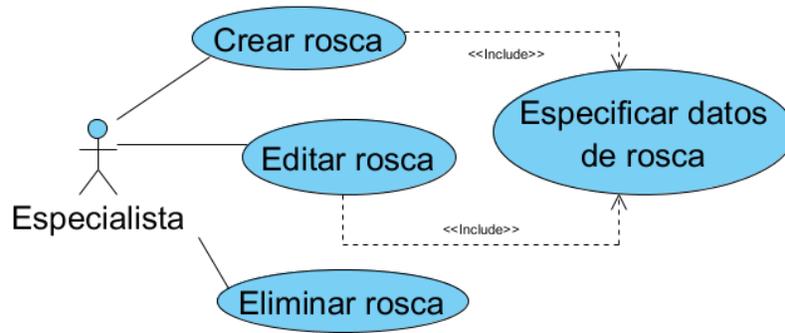


Fig. 23 Diagrama de Casos de Uso del sistema.

2.3.3. Descripción de los Casos de Uso

La descripción de los casos de uso permite comprender los procesos que se encuentran asociados a cada uno de ellos. A continuación se muestran las descripciones de los casos de uso.

Caso de Uso	Crear rosca.	
Actores	Especialista.	
Resumen	El especialista selecciona la operación a realizar y especifica los datos para la ejecución de dicha operación. El sistema realiza la operación seleccionada.	
Precondiciones	Debe existir una superficie cilíndrica creada.	
Referencias	RF 2, RF 2.1, RF 2.2, RF 2.3, RF 2.4, RF 3	
Prioridad	Crítico.	
Flujo Normal de Eventos		
Sección "Crear rosca"		
	Acción del Actor	Respuesta del Sistema
1.	El especialista selecciona la operación "Thread".	
2.		2.1. El sistema verifica que exista una superficie cilíndrica creada. 2.2. El sistema muestra un cuadro de dialogo para la selección de las especificaciones de la rosca.

3.	<p>3.1. El especialista selecciona la superficie cilíndrica donde se creará la rosca.</p> <p>3.2. El especialista selecciona los parámetros necesarios para crear la rosca. (Ver caso de uso incluido “Especificar datos de rosca”).</p> <p>3.3. El especialista acepta los datos seleccionados.</p>	
4.		<p>4.1. El sistema crea y visualiza la rosca con los datos seleccionados.</p> <p>4.2. Termina el caso de uso.</p>
Flujo Alternativo 2.1.a No existe superficie cilíndrica.		
1.		El sistema muestra un mensaje de error indicando que no existe una superficie cilíndrica sobre la cual crear la rosca.
2.	El especialista acepta el mensaje de error.	
3.		Termina el caso de uso.
Poscondiciones	Debe quedar creada la rosca de acuerdo a las especificaciones seleccionadas.	

Tabla 2: Descripción del Caso de Uso “Crear rosca”.

Caso de Uso	Editar rosca.	
Actores	Especialista.	
Resumen	El especialista selecciona la operación a realizar y modifica los parámetros de la rosca creada. El sistema visualiza los cambios y realiza la operación.	
Precondiciones	Debe existir en la aplicación al menos una rosca creada.	
Referencias	RF 2.4, RF 4, RF 4.1	
Prioridad	Crítico.	
Flujo Normal de Eventos		
Sección “Modificar parámetros de rosca”		
	Acción del Actor	Respuesta del sistema

1.	El especialista selecciona la rosca a modificar y activa la opción "Edit thread".	
2.		2.1. El sistema verifica que se haya seleccionado una rosca. 2.2. El sistema muestra un cuadro de dialogo para la selección de las especificaciones que se desean modificar.
3.	3.1. El especialista selecciona los parámetros necesarios para la modificación de la rosca. (Ver caso de uso incluido "Especificar datos de rosca"). 3.2. El especialista acepta los datos seleccionados.	
4.		4.1. El sistema crea y visualiza la rosca modificada de acuerdo a los datos seleccionados. 4.2. Termina el caso de uso.
Flujo Alternativo 2.1.a No se seleccionó una rosca.		
1.		El sistema muestra un mensaje de error indicando que no hay una rosca seleccionada.
2.	El especialista acepta el mensaje de error.	
3.		Termina el caso de uso.
Poscondiciones	Se debe modificar la rosca de acuerdo a los cambios aplicados.	

Tabla 3: Descripción del Caso de Uso "Editar rosca".

Caso de Uso	Eliminar rosca.
Actores	Especialista.
Resumen	El especialista selecciona la operación a realizar y determina que rosca será eliminada de la aplicación. El sistema realiza la operación.
Precondiciones	Debe existir en la aplicación al menos una rosca creada.

Referencias	RF 5, RF 5.1	
Prioridad	Crítico.	
Flujo Normal de Eventos		
Sección “Eliminar rosca”		
	Acción del Actor	Respuesta del sistema
1.	El especialista selecciona la rosca a eliminar y activa la opción “Delete thread”.	
2.		2.1. El sistema verifica que se haya seleccionado una rosca. 2.2. El sistema elimina de la aplicación la rosca seleccionada. 2.3. El sistema elimina de la aplicación cualquier operación que dependa de la rosca eliminada. 2.4. Termina el caso de uso.
Flujo Alternativo 2.1.a No se seleccionó una rosca.		
1.		El sistema muestra un mensaje de error indicando que no hay una rosca seleccionada.
2.	El especialista acepta el mensaje de error.	
3.		Termina el caso de uso.
Poscondiciones	Debe quedar eliminada la rosca de la aplicación.	

Tabla 4: Descripción del Caso de Uso “Eliminar rosca”.

Caso de Uso	Especificar datos de rosca.
Actores	Especialista.
Resumen	El especialista selecciona las especificaciones de la rosca. El sistema visualiza dichas especificaciones.
Precondiciones	Deben estar activadas alguna de las opciones “Thread” o “Edit thread” y debe haber una superficie cilíndrica o una rosca seleccionada para cada opción respectivamente.
Referencias	RF 2, RF 2.1, RF 2.2, RF 2.3, RF 2.4

Prioridad	Crítico.	
Flujo Normal de Eventos		
Sección “Especificar datos de rosca”		
	Acción del Actor	Respuesta del sistema
1.	El especialista especifica la distancia de la rosca y desactiva la opción “Full length”.	
2.		El sistema activa las opciones “Ofsset” y “Length”.
3.	3.1. El especialista introduce la distancia a partir de la cual se comienza a visualizar la rosca (ofsset). 3.2. El especialista introduce la distancia de la rosca (length). 3.3. El especialista selecciona el tipo de norma de la rosca. 3.4. El especialista selecciona el tamaño de la rosca. 3.5. El especialista selecciona la designación de la rosca. 3.6. El especialista selecciona la clase de la rosca. 3.7. El especialista selecciona la orientación de la rosca.	
4.		El sistema previsualiza la rosca a partir de los datos especificados.
5.	El especialista acepta los datos.	
6.		6.1. El sistema verifica que los datos especificados sean correctos. 6.2. El sistema visualiza la rosca de acuerdo a los datos especificados. 6.3. Termina el caso de uso

Flujo Alternativo 6.1.a Los datos introducidos son incorrectos.		
1.		El sistema muestra un mensaje de error indicando que los datos introducidos en las distancias son incorrectos.
2.	El especialista acepta el mensaje.	
3.		Retorna al paso 3 del flujo normal de eventos.
Poscondiciones		Deben quedar seleccionadas las especificaciones de la rosca.

Tabla 5: Descripción del Caso de Uso “Especificar datos de rosca”.

2.4. DISEÑO DEL SISTEMA

2.4.1. Modelo de dominio

El modelo de dominio captura los tipos más importantes de objetos en el contexto del sistema. Los objetos del dominio representan las entidades que existen o los eventos que suceden en el entorno en el que trabaja el sistema (26). Permite definir el alcance de la aplicación, quiénes y qué elementos están involucrados en el desarrollo de la misma, representa un modelo conceptual de las entidades que participan en el negocio y las relaciones entre ellas.

A continuación una breve descripción de las entidades involucradas:

- **Especialista:** Persona que interactúa con el sistema a través de la interfaz gráfica. Se encarga de seleccionar la superficie cilíndrica sobre la cual se visualiza la rosca y establecer los datos de especificación de la misma.
- **Superficie cilíndrica:** Superficie correspondiente a un sólido sobre la cual se visualiza la rosca. Es seleccionada por el especialista.
- **Especificaciones de rosca:** Conjunto de datos necesarios para la gestión de la rosca y que son definidos por el especialista.
- **Interfaz gráfica:** Componente visual que permite la interacción del especialista con el sistema. Los datos de especificación se definen a través de la interfaz gráfica.

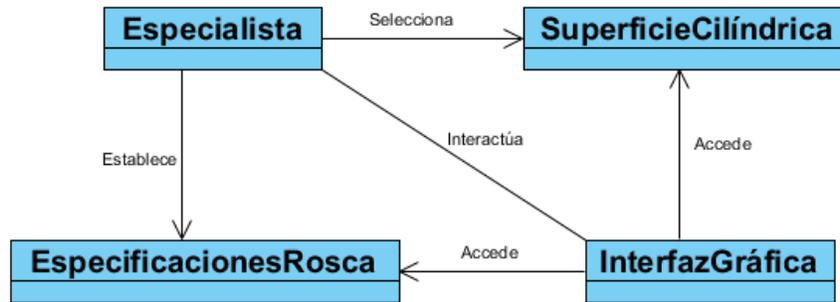


Fig. 24 Modelo de dominio.

2.4.2. Diagrama de paquetes

Los paquetes se pueden utilizar para mostrar agrupaciones lógicas de objetos. Cada paquete agrupa un conjunto de responsabilidades. Las clases necesarias para el funcionamiento de la herramienta a desarrollar tienen responsabilidades que las distinguen unas de otras por lo que se puedan clasificar y agrupar por paquetes. Como el módulo a desarrollar formará parte de un sistema, debe mantener la estructura y organización que posee el mismo.

A continuación se muestra la agrupación en paquetes de las clases utilizadas en el desarrollo de la aplicación.

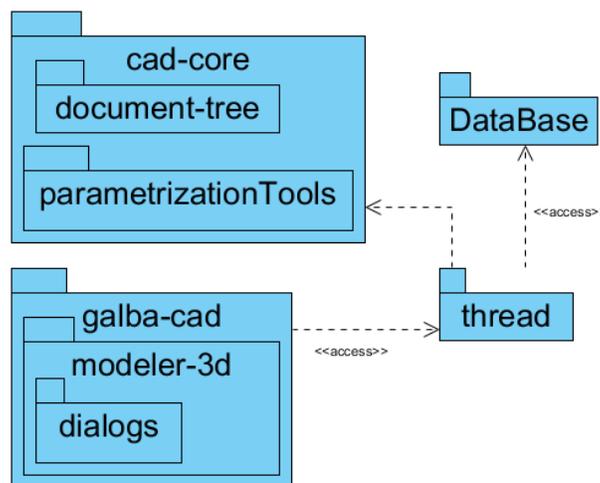


Fig. 25 Diagrama de paquetes del diseño.

2.4.3. Diagrama de clases

El diagrama de clases describe la estructura de la aplicación, muestra el diseño conceptual de la información que se maneja, los componentes que se encargan de las funcionalidades y la relación entre uno y otro. (Ver diagrama ampliado en anexo 1)

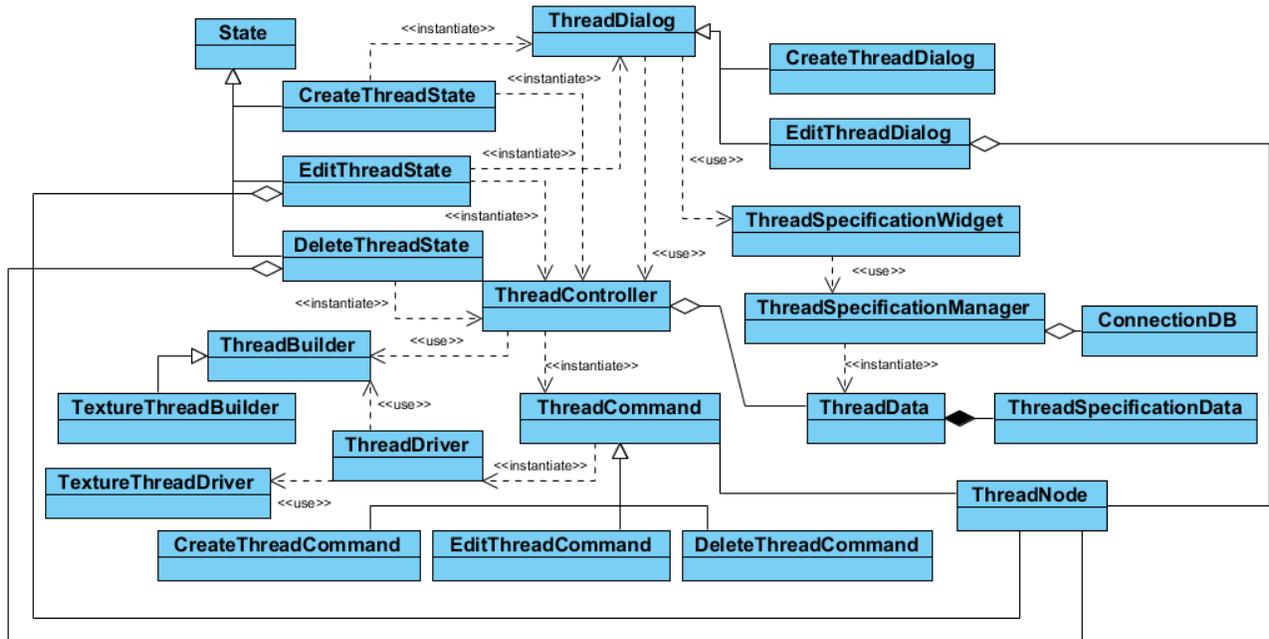


Fig. 26 Diagrama de clases.

CONCLUSIONES DEL CAPÍTULO

A partir de la solución propuesta y el estudio realizado se puede apreciar que la visualización mediante la utilización de texturas propuesta, puede ser utilizado no solo para las superficies cilíndricas sino también para crear roscas en las superficies cónicas, dado que el cálculo de la altura de la superficie y de la cantidad de repeticiones necesarias de la textura es similar en ambos casos.

Teniendo en cuenta el estudio de las normas y estándares utilizados en la industria para la construcción y especificación de roscas, la utilización de los datos: tipo de rosca, tamaño, designación y clase, es suficiente para acceder al resto de los datos de una especificación completa de la rosca para cualquier norma seleccionada.

Considerando el estudio y posterior utilización del enfoque modelo / vista que propone el framework Qt, se aprecia la flexibilidad y extensibilidad que este enfoque brinda a los datos que

se obtienen como resultado de las consultas a la base de datos, permitiendo que estos datos puedan mostrarse en varias vistas a la vez.

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

En el presente capítulo se exponen los principales elementos de la implementación y las pruebas realizadas a la aplicación. Se presentan los diagramas correspondientes al modelo de implementación y el diagrama de componentes. Además se detallan los casos de pruebas realizados al módulo propuesto. Se describen también los resultados obtenidos de la presente investigación.

3.1. IMPLEMENTACIÓN DE LA SOLUCIÓN PROPUESTA

3.1.1. Diagrama de componentes

Un componente es el empaquetamiento físico de los elementos de un modelo. Los componentes pueden ser ejecutables, archivos de código fuente, librerías, tablas de datos, documentos, etc. Dichos componentes tienen relaciones con los elementos del modelo que implementan (26). (Ver diagrama ampliado en anexo 2)

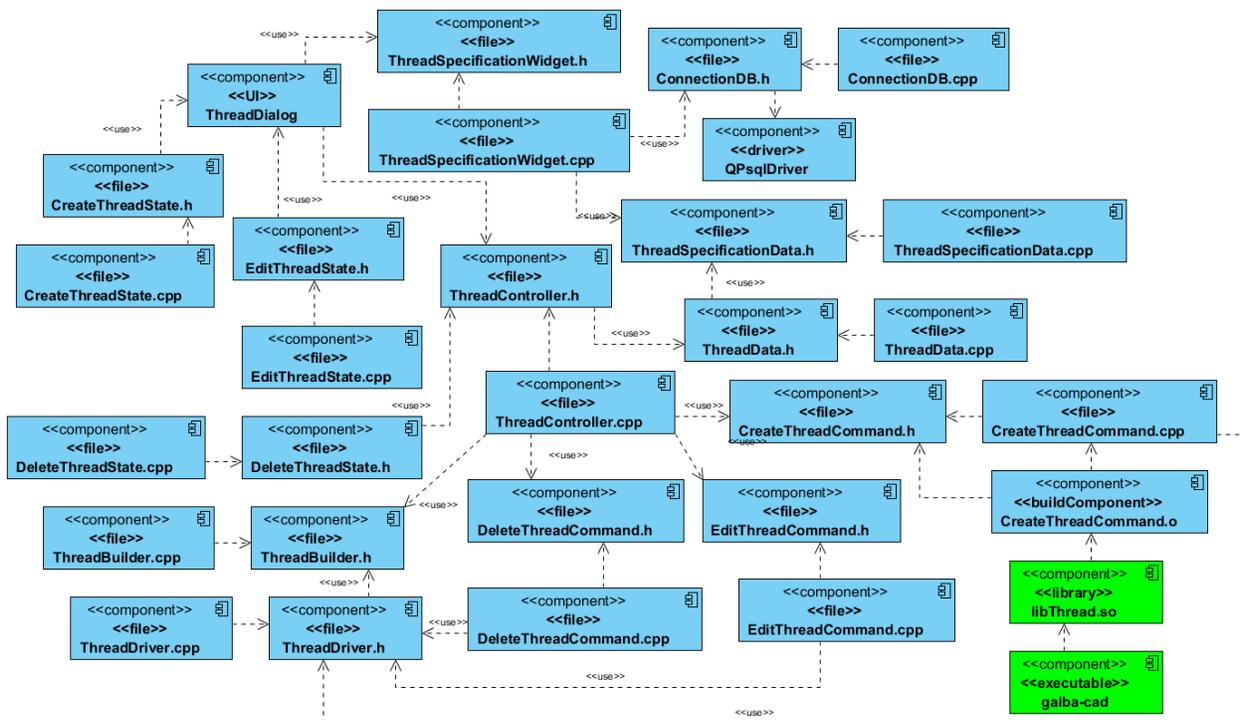


Fig. 27 Diagrama de componentes.

3.2. RESULTADOS

Después de terminada la implementación de la solución propuesta, se obtiene como resultado un módulo de gestión y visualización de roscas con cuatro funcionalidades fundamentales, la creación, edición y eliminación de roscas, así como la previsualización tras cada paso del proceso de creación y edición y la visualización final, una vez creada la rosca. El módulo desarrollado se integró al sistema GALBA-CAD, logrando de esta manera agregar nuevas funcionalidades al sistema y aumentando su usabilidad. A continuación se describen cada una de las funcionalidades del módulo desarrollado.

3.2.1. Crear una rosca

Con el desarrollo de esta funcionalidad se permite la creación de una rosca en el sistema GALBA-CAD así como asociar la misma a superficies cilíndricas seleccionadas previamente. A continuación se describen los pasos básicos para llevar a cabo esta operación.

1. Accionar la opción “Thread” de la barra superior del sistema. Se mostrará al usuario un cuadro de diálogo para la selección de algunos parámetros necesarios para la creación de la rosca. (ver Fig. 23)
2. Para lograr la previsualización de la rosca en correspondencia con los datos que se seleccionen, como segundo paso, se debe seleccionar la superficie cilíndrica sobre la cual se quiere crear la rosca. (ver Fig. 23)
3. Seleccionada la superficie cilíndrica, se pasa a seleccionar entonces los parámetros correspondientes a la rosca que se desee crear. A medida que se va seleccionando algún parámetro el módulo permite la previsualización de la rosca en correspondencia con el dato seleccionado. Los datos de especificación de la rosca son actualizados automáticamente en dependencia de los valores que se vayan seleccionando. (ver Fig. 23)
4. Una vez que han sido seleccionado los parámetros correspondientes, se acciona el botón “OK”. De esta manera se crea la rosca internamente en el sistema y se adiciona una entrada en el historial del documento, indicando que la rosca ha sido creada. (ver Fig. 24)

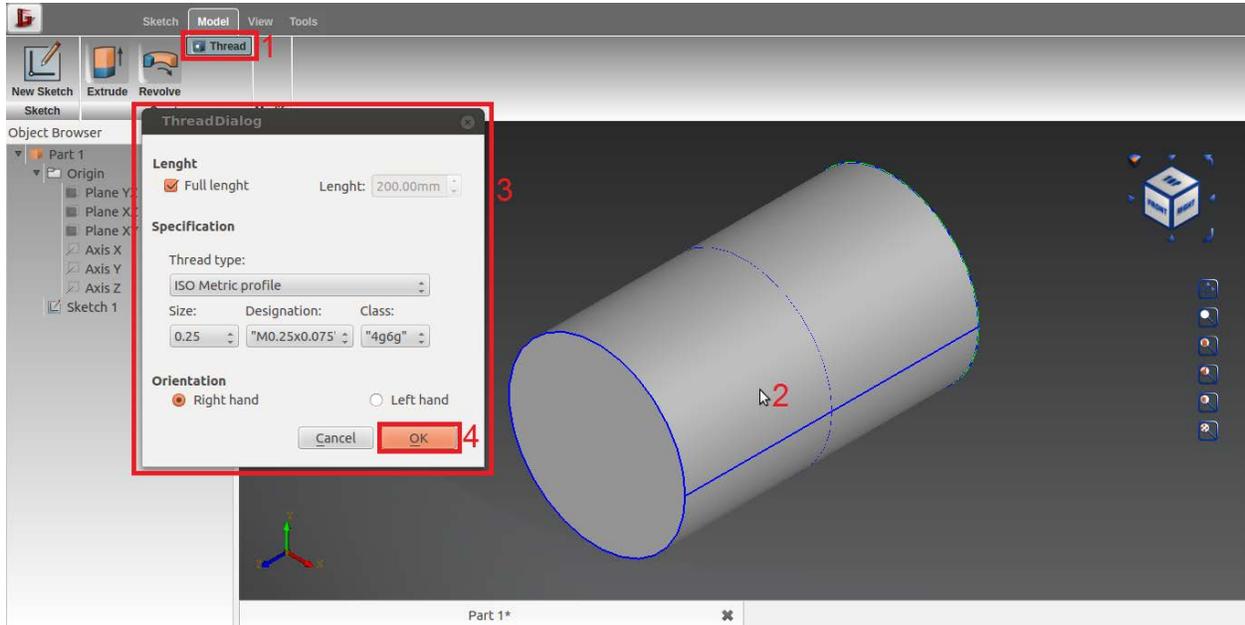


Fig. 28: Pasos para la creación de una rosca en el sistema GALBA-CAD.

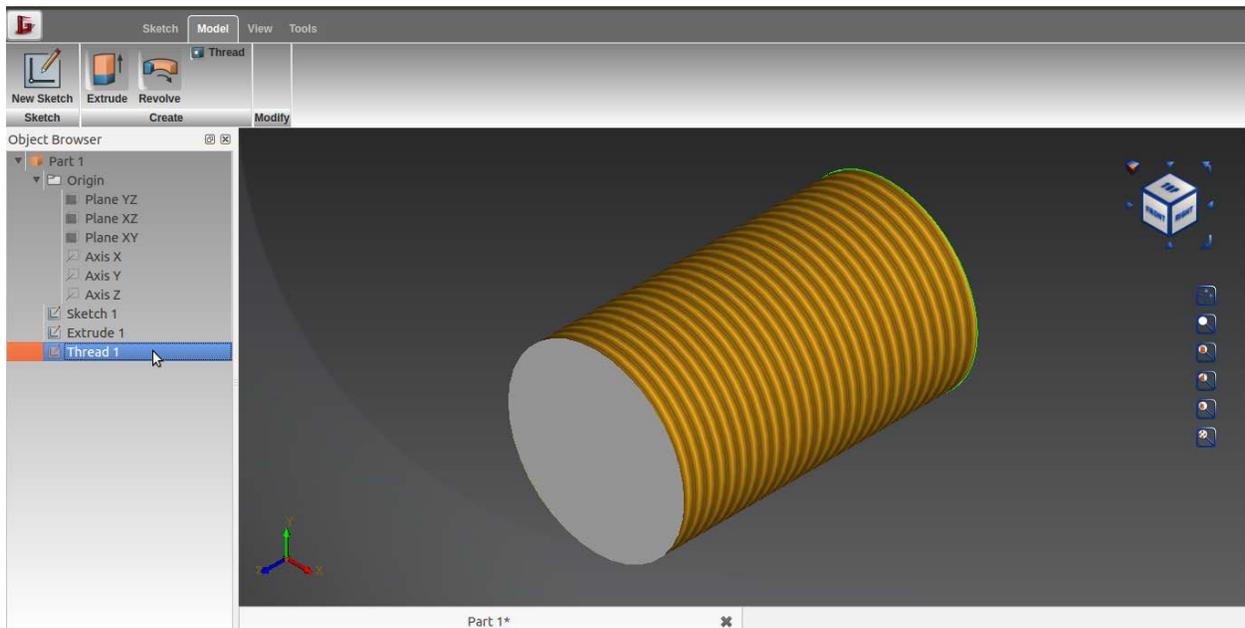


Fig. 29: Rosca creada en el sistema GALBA-CAD.

3.2.1. Editar una rosca

La funcionalidad de edición de una rosca permite cambiar los parámetros de una rosca existente, sin la necesidad de crear una nueva rosca. Los pasos para la edición de una rosca son muy sencillos y se describen a continuación.

1. Seleccionar en el historial del documento la rosca que se desee editar. Habilitar la opción “Edit” haciendo clic derecho sobre la entrada en el historial y escogiendo la opción correspondiente. Se mostrará al usuario un cuadro de diálogo con los datos de la rosca seleccionada. (ver Fig. 25)
2. Modificar los parámetros correspondientes, igual que en el proceso de creación, el módulo permite la previsualización de los datos que se seleccionen y la actualización automática de los mismos. Una vez que se modificaron los parámetros se acciona el botón “OK”. El sistema visualiza la rosca modificada de acuerdo a los parámetros seleccionados.

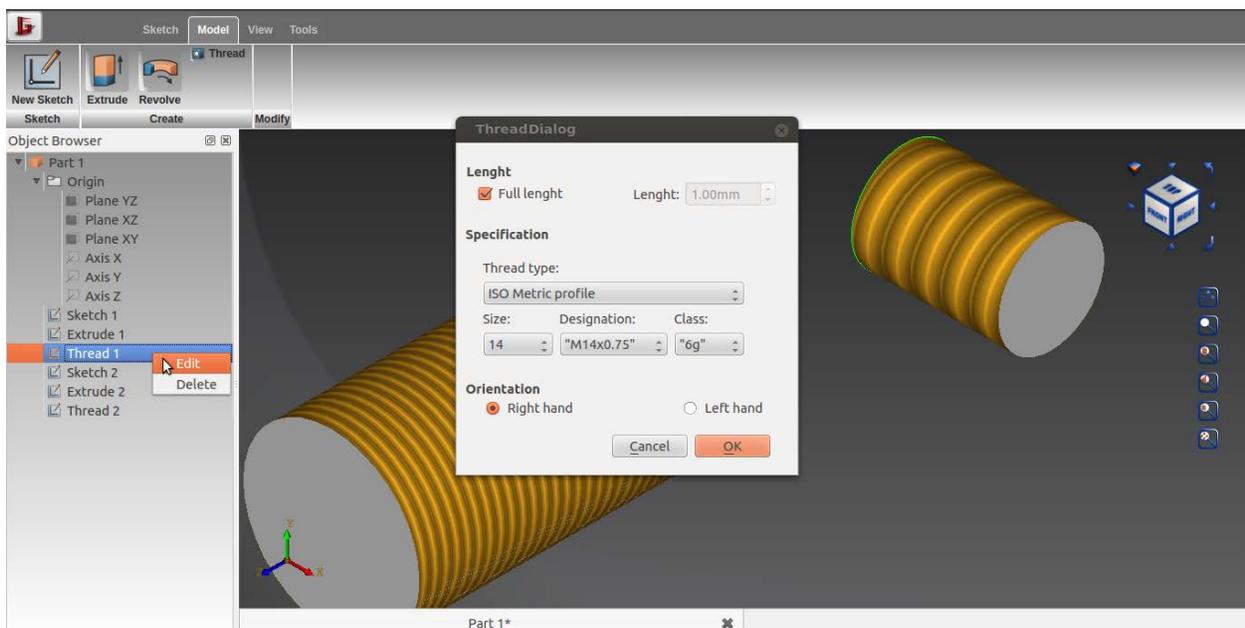


Fig. 30: Pasos para editar una rosca creada en el sistema GALBA-CAD.

3.2.1. Eliminar una rosca

Con la funcionalidad de eliminar una rosca se completa el proceso de gestión de roscas del módulo desarrollado. Esta funcionalidad permite eliminar del sistema una rosca determinada. A continuación se describen los pasos para ejecutar esta funcionalidad.

1. Seleccionar en el historial del documento la rosca que se desee eliminar. Habilitar la opción “Delete” haciendo clic derecho sobre la entrada en el historial y escogiendo la opción correspondiente. (ver Fig. 26)
2. De manera inmediata se elimina la visualización de la rosca seleccionada, y se elimina la entrada de la rosca en cuestión en el historial del documento. (ver Fig. 27)

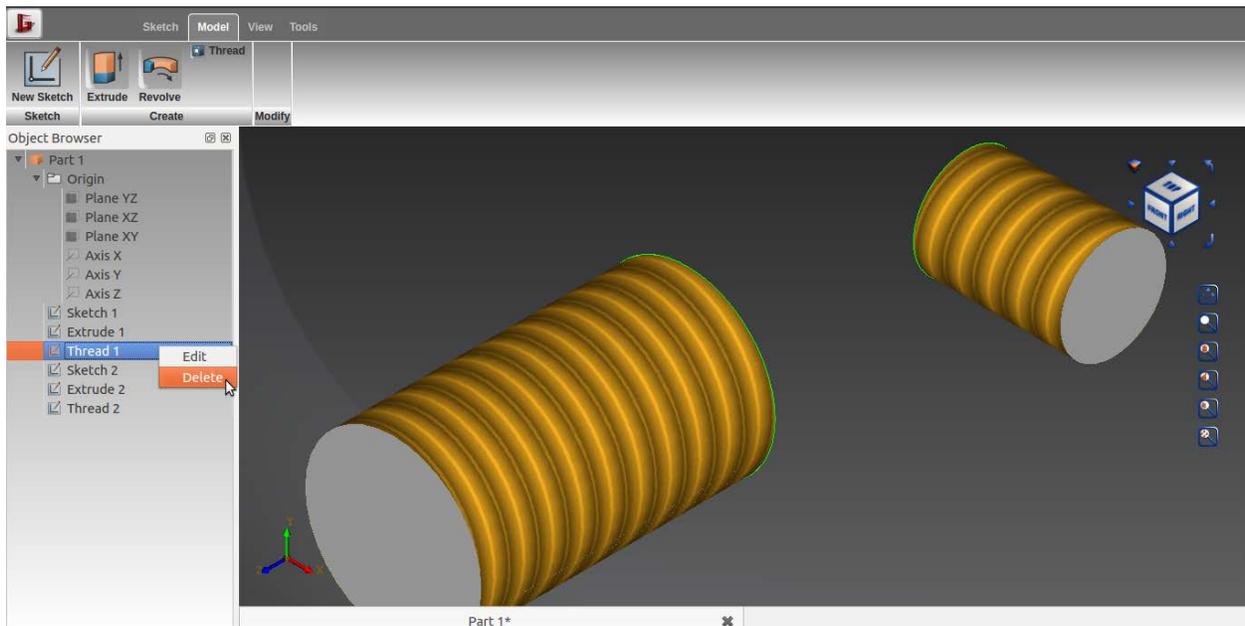


Fig. 31: Pasos para eliminar una rosca en el sistema GALBA-CAD.

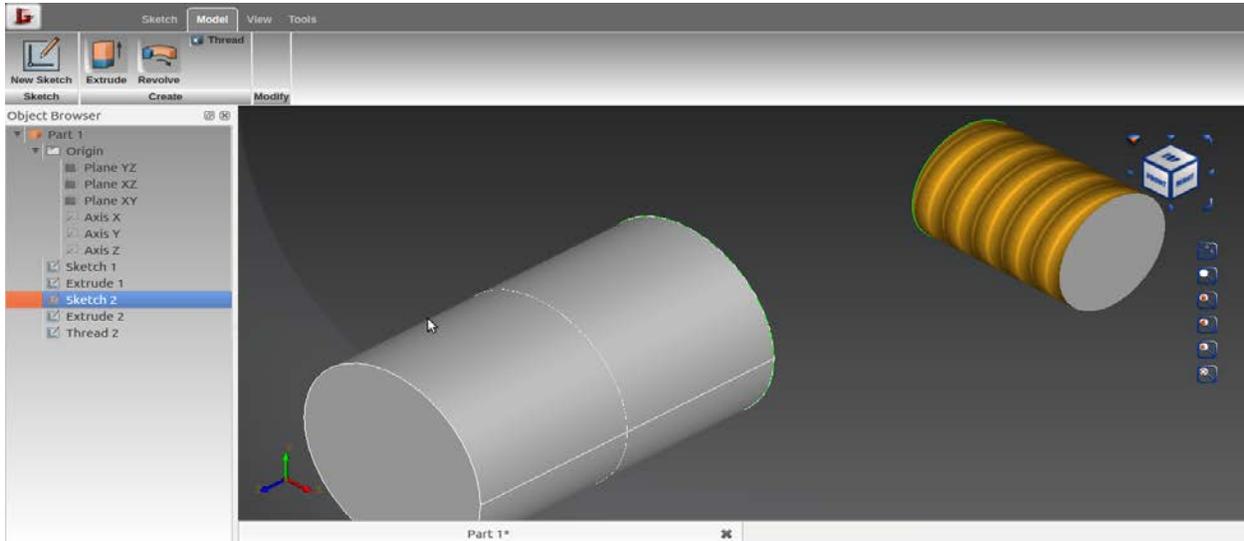


Fig. 32: Rosca eliminada del sistema GALBA-CAD.

3.3. PRUEBAS AL MÓDULO DESARROLLADO

3.3.1. Casos de prueba

Las pruebas de especificación o pruebas de caja negra se realizan para probar el comportamiento externo de la aplicación sin tener en cuenta como se ha implementado. Un caso de prueba muestra una forma de probar la aplicación incluyendo la entrada o resultado con el que ha de probarse y las condiciones bajo las cuales ha de probarse en un estado determinado (26).

3.3.1.1. Caso de prueba # 1

Escenario	Descripción	No. Prueba	Variable	Respuesta del sistema	Flujo central
1. Seleccionar superficie cilíndrica.	Verificar la correcta selección de una superficie cilíndrica.	1	N/A	Previsualiza la rosca de acuerdo a los datos especificados sobre la superficie seleccionada.	1. Seleccionar una superficie.

Tabla 6: Caso de prueba # 1.

3.3.1.2. Caso de prueba # 2

Escenario	Descripción	No. Prueba	Variable	Respuesta del sistema	Flujo central
1. Seleccionar la norma de la rosca.	Verificar la actualización de los valores del campo "Tamaño" de acuerdo a la norma seleccionada.	1	N/A	Activa el primer elemento de los valores del campo "Tamaño".	1. Seleccionar una de las normas de las disponibles en el listado.
2. Seleccionar el tamaño de la rosca.	Verificar la actualización de los valores del campo "Designación" de acuerdo al tamaño seleccionado	1	N/A	Activa el primer elemento de los valores del campo "Designación". Previsualiza la rosca según los datos especificados.	1. Seleccionar uno de los tamaños de los disponibles en el listado.
3. Seleccionar la designación de la rosca.	Verificar la actualización de los valores del campo "Clase" de acuerdo a la designación seleccionada y la posición de la rosca (Externa o Interna).	1	N/A	Muestra los valores del campo en dependencia de la posición de la rosca (Externa o Interna). Activa el primer elemento de los valores del campo "Clase".	1. Seleccionar una de las designaciones de las disponibles en el listado.
4. Seleccionar la clase de la rosca.	Verifica la previsualización de la rosca de acuerdo a los	1	N/A	Previsualiza la rosca según los datos especificados.	1. Seleccionar una de las clases de las disponibles en

	parámetros especificados.				el listado.
5. Seleccionar la orientación de la rosca.	Verificar la selección de uno de los dos campos de orientación y la previsualización de la rosca de acuerdo a los parámetros especificados.	1	N/A	Deshabilitar el campo que no ha sido seleccionado. Previsualizar la rosca según la orientación seleccionada.	1. Seleccionar una de las orientaciones de las disponibles.

Tabla 7: Caso de prueba # 2.

3.3.1.3. Caso de prueba # 3

Escenario	Descripción	No. Prueba	Longitud	Respuesta del sistema	Flujo central
1. Establecer longitud de la rosca.	Verificar la activación del campo "Longitud Total"	1	N/A	Deshabilita el campo "Longitud". Previsualiza la rosca sobre toda la superficie.	1. Activar el campo "Longitud Total".
	Verificar que el campo "Longitud Total" está inactivo y los valores introducidos son correctos.	2	V 1.50mm	Previsualiza la rosca solo para la distancia especificada.	1. Insertar la longitud deseada en el campo "Longitud"
	Verificar que el	3	I	Muestra un	

	campo "Longitud Total" está inactivo y los valores introducidos son incorrectos.	4	abC	mensaje de error indicando que los datos introducidos no son válidos.	
			I		
			@*/		
		5	I		
			0		
		6	I		
	-2.25mm				
2. Visualización de la rosca de acuerdo a la longitud seleccionada.	Verificar que se visualiza la rosca en la longitud seleccionada	1	V	Visualizar la rosca solo para la longitud del cilindro seleccionada.	1. Insertar la longitud deseada en el campo "Longitud" 2. Hacer clic en el botón "OK"
			5 mm (La altura total del cilindro es de 10 mm)		

Tabla 8: Caso de prueba # 3.

No	Nombre del campo	Clasificación	Valor nulo	Descripción
1	Longitud	Campo de texto	No	Valor correspondiente a la distancia sobre la cual se quiere aplicar la rosca.

Tabla 9: Descripción de variables del caso de prueba # 3.

3.3.1.4. Caso de prueba # 4

Escenario	Descripción	No. Prueba	Variable	Respuesta del sistema	Flujo central
1. Crear y visualizar rosca.	Verificar que se muestre el diálogo para la selección de los datos con	1	N/A	Muestra un diálogo para la selección de los parámetros correspondientes a la creación de la	1. Seleccionar la opción "Rosca".

	los parámetros seleccionados por defecto.			rosca con parámetros ya seleccionados por defecto.	
	Verificar que se visualiza el resultado final de la operación “Rosca” una vez pulsado el botón “OK”	2	N/A	Visualiza la rosca después de seleccionados todos los parámetros correspondientes.	1. Seleccionar los parámetros correspondientes. 2. Pulsar botón “OK”.

Tabla 10: Caso de prueba # 4.

3.3.1.5. Caso de prueba # 5

Escenario	Descripción	No. Prueba	Variable	Respuesta del sistema	Flujo central
1. Editar rosca.	Verificar que se muestra el dialogo con los valores correspondientes a la rosca que se desea editar seleccionados.	1	N/A	Muestra el dialogo con los valores de la rosca que se desea editar seleccionados.	1. Seleccionar la operación 3D rosca que se desea modificar. 2. Activar la opción “Editar Rosca”.
	Verificar que se visualiza la rosca modificada con los nuevos valores una vez pulsado el botón “OK”.	2	N/A	Modifica y visualiza la rosca en correspondencia con los parámetros modificados.	1. Modificar los parámetros correspondientes. 2. Pulsar botón “OK”.

Tabla 11: Caso de prueba # 5.

3.3.1.6. Caso de prueba # 6

Escenario	Descripción	No. Prueba	Variable	Respuesta del sistema	Flujo central
1. Eliminar rosca	Verificar que se elimina la operación 3D rosca seleccionada.	1	N/A	Elimina la visualización de la rosca seleccionada. Elimina los datos guardados en el sistema, que corresponden a la rosca a eliminar. Elimina la operación 3D rosca seleccionada de la lista de operaciones 3D.	1. Seleccionar la operación 3D rosca que se desea eliminar. 2. Activar la opción "Eliminar Rosca".

Tabla 12: Caso de prueba # 6.

3.4. RESULTADOS DE LAS PRUEBAS.

En la tabla que aparece a continuación se muestra un resumen de los resultados de las pruebas realizadas al módulo. Se puede apreciar para cada escenario las pruebas que lograron resultados satisfactorios y cuales no obtuvieron los resultados esperados.

No. Caso de prueba	Escenario	No. Prueba	Resultado
1	1. Seleccionar superficie cilíndrica.	1	Satisfactorio
2	1. Seleccionar la norma de la rosca.	1	Satisfactorio
	2. Seleccionar el tamaño de la rosca.	1	Satisfactorio
	3. Seleccionar la designación de la rosca.	1	No satisfactorio
	4. Seleccionar la clase de la rosca.	1	Satisfactorio
	5. Seleccionar la orientación de la rosca.	1	Satisfactorio
3	1. Establecer longitud de la rosca.	1	Satisfactorio

		2	Satisfactorio
		3	Satisfactorio
		4	Satisfactorio
		5	Satisfactorio
		6	Satisfactorio
	2. Visualización de la rosca de acuerdo a la longitud seleccionada.	1	No satisfactorio
4	1. Crear y visualizar rosca.	1	Satisfactorio
		2	Satisfactorio
5	1. Editar rosca.	1	Satisfactorio
		2	Satisfactorio
6	1. Eliminar rosca	1	Satisfactorio

Tabla 13: Resultados de las pruebas.

Para el caso de prueba número dos, la prueba uno del escenario tres, no obtuvo resultados satisfactorios, dado que al seleccionar un valor de designación se muestran todas las clases correspondientes a dicha designación sin tener en cuenta la posición de la superficie. Esta no conformidad surge a partir de que no hay un mecanismo implementado que determine la posición de la superficie seleccionada, pues implica una investigación que no se considera dentro del alcance de la tesis.

Para el caso de prueba tres, la prueba uno del escenario dos, el sistema tuvo un comportamiento diferente a la respuesta esperada. Para un valor de longitud menor que la altura total de la superficie se obtiene la visualización de la textura en toda la superficie. Este problema viene dado porque la implementación actual de la clase *AIS_TextureShape* utilizada no permite la visualización de la textura para una longitud determinada.

CONCLUSIONES DEL CAPÍTULO

Terminado el desarrollo del módulo propuesto se puede concluir que:

Las operaciones de creación, edición y eliminación de roscas pueden ser reutilizables en otras operaciones 3D como la operación de creación de orificios. La implementación del enfoque modelo / vista permitió la extensibilidad de los datos resultados de las consultas.

A partir de los casos de prueba realizados se comprobó que la implementación realizada satisface los requerimientos del sistema excepto la visualización de la rosca en una longitud determinada menor que la longitud total de la superficie del cilindro y la correcta actualización de los valores de la clase de la rosca a partir de la posición, dígase externa o interna, de la misma.

El proceso de visualización de texturas utilizando la clase AIS_TextureShape impide la visualización en una porción de la superficie relacionada, dado que para cada triángulo producto de la triangulación a la superficie, asocia una coordenada de textura.

Para valores altos del paso de la rosca la visualización de la textura correspondiente presenta problemas, afectando notablemente al realismo de la misma.

CONCLUSIONES

A partir del estudio realizado para el desarrollo de esta investigación y una vez concluido el desarrollo del módulo de gestión y visualización de roscas, se obtuvieron las siguientes conclusiones:

- Las roscas de perfil triangular son una de las más utilizadas en la industria, además el Sistema Internacional constituye el más utilizado dado que la mayoría de la tornillería actual se desarrolla bajo este sistema.
- La visualización por textura como propone *Autodesk Inventor* fue la más apropiada para el módulo desarrollado dado que abstrae al usuario de todo el proceso interno de creación de la rosca. Esta forma de visualización puede adaptarse tanto para superficies cilíndricas como para superficies cónicas dado que el proceso de visualización es similar en ambos casos.
- A partir de los datos: tipo de rosca, tamaño, designación y clase, se puede acceder al resto de los datos que incluye una especificación completa de la rosca.
- La implementación del enfoque modelo / vista permitió la extensibilidad de los datos resultados de las consultas a la base de datos.
- El módulo de gestión y visualización de roscas, puede ser reutilizable en otras operaciones 3D como la creación de orificios.
- El proceso de visualización de texturas utilizando la clase *AIS_TextureShape* impide la visualización en una porción de la superficie relacionada.
- Para valores altos del paso de la rosca la visualización de la textura correspondiente presenta problemas, afectando notablemente al realismo de la misma.

RECOMENDACIONES

Los objetivos de este trabajo de diploma han sido logrados, pero a lo largo de su desarrollo, han surgido ideas que podrían implementarse en un futuro, de esta manera puede lograrse una aplicación más útil y efectiva, para lo cual se recomienda:

- Asociar una textura a cada estándar de rosca usado en la aplicación, así como incorporar un modelo de iluminación para lograr mayor realismo en la visualización mediante texturas.
- Corrección de la visualización de la textura para:
 - Evitar el color amarillo.
 - Visualizar la rosca en una longitud menor que la longitud total de la superficie cilíndrica. Valorar la reimplementación de la clase *AIS_TextureShape*.
- Implementar un mecanismo para determinar si la superficie cilíndrica seleccionada por el usuario es interna o externa, respecto al sólido que la contiene.

REFERENCIAS BIBLIOGRÁFICAS

1. **Esteban José Domínguez Soriano, Julián Ferrer Rui.** *Mecanizado básico.* s.l. : Editex, 2008. págs. 108-113. ISBN/8497714180, 9788497714181.
2. **scribd.com.** Definiciones y generalidades sobre roscas. [En línea] [Citado el: 2 de 11 de 2012.] www.scribd.com/doc/57960662/TIPOS-DE-ROSCAS.
3. **fastenal.com.** Article Screw Threads Design. [En línea] [Citado el: 14 de 11 de 2012.] www.fastenal.com/content/feds/pdf/Article%20-%20Screw%20Threads%20Design.pdf.
4. **Autodesk.com.** [En línea] [Citado el: 19 de Noviembre de 2012.] http://wikihelp.autodesk.com/Inventor/enu/2012/Help/0073-Autodesk73/0308-Parts308/0353-Part_fea353/0355-Create_f355/0369-Thread_f369.
5. **SolidWork.com.** SolidWork. [En línea] [Citado el: 13 de Diciembre de 2012.] <http://www.solidworks.com/>.
6. **engineering.union.edu.** How to create helical features. [En línea] [Citado el: 13 de 12 de 2012.] engineering.union.edu/~tchakoa/mer101/HOW%20TO%20CREATE%20HELICAL%20FEATURES.pdf.
7. **3ds.com.** La experiencia del producto digital. [En línea] [Citado el: 13 de Diciembre de 2012.] <http://www.3ds.com/es/products/catia/>.
8. **Maruf.ca.** The Objects of CATIA Geometric Modeler. [En línea] [Citado el: 3 de Diciembre de 2012.] <http://www.maruf.ca/files/caadoc/CAAGobTechArticles/GeoObjects.htm>.
9. **Varicad.com.** VariCAD 3D / 2D CAD software for mechanical engineering. [En línea] [Citado el: 16 de Enero de 2013.] <http://www.varicad.com/en>.
10. —. 3D Comprehensive Shapes. [En línea] [Citado el: 3 de Diciembre de 2012.] http://www.varicad.com/userdata/files/manual/en/3D_Comprehensive_Shapes.htm#FTag33.
11. **Salome-platform.org.** Welcome to Salome. [En línea] [Citado el: 20 de Enero de 2013.] <http://www.salome-platform.org/>.

12. —. About geometry. [En línea] [Citado el: 20 de Enero de 2013.] <http://www.salome-platform.org/about/geometry>.
13. **Sourceforge.net**. Draft Module. [En línea] [Citado el: 10 de Diciembre de 2012.] http://sourceforge.net/apps/mediawiki/free-cad/index.php?title=Draft_Workbench.
14. **Sourceforge.net/**. Part Module. [En línea] [Citado el: 10 de Diciembre de 2012.] http://sourceforge.net/apps/mediawiki/free-cad/index.php?title=Part_Workbench.
15. **Brighthubengineering.com**. How to create screw threads in pro engineer. [En línea] [Citado el: 16 de Noviembre de 2012.] <http://www.brighthubengineering.com/cad-autocad-reviews-tips/26983-how-to-create-screw-threads-in-pro-engineer/>.
16. **BRLCAD.org**. Overview BRLCAD. [En línea] [Citado el: 17 de Enero de 2013.] <http://brlcad.org/wiki/Overview>.
17. **OpenCASCADE.org**. Open CASCADE Technology, 3D modeling & numerical simulation. [En línea] [Citado el: 17 de Enero de 2013.] <http://www.opencascade.org/>.
18. —. Advantages. [En línea] [Citado el: 17 de Enero de 2013.] <http://www.opencascade.org/occt/advantages/>.
19. **OpenCASCADE Technology**. *Modeling Data*. 2011.
20. —. *Visualization*. 2011.
21. —. *Application Framework*. 2011.
22. **Bhandari, V. B.** *Design Of Machine Elements*. 2da Edición. s.l. : Tata McGraw-Hill Education, 2007. pp. 248-249. ISBN/0070611416, 9780070611412.
23. —. *Design Of Machine Elements*. 2da edición. s.l. : Tata McGraw-Hill Education, 2007. págs. 202-204. ISBN/0070611416, 9780070611412.
24. **David A. Madsen, David P. Madsen**. *Engineering Drawing & Design*. 5ta Edición. s.l. : Cengage Learning, 2011. págs. 412-413. ISBN/1111309574, 9781111309572.
25. **qt-project.org**. Model/View Tutorial. [En línea] [Citado el: 9 de 5 de 2013.] <http://qt-project.org/doc/qt-4.8/modelview.html>.

26. Ivar Jacobson, Grady Booch, James Rumbaugh. *El proceso unificado de desarrollo de software*. s.l. : Pearson Educación S.A, 2000. ISBN/847859-0362.

GLOSARIO DE TÉRMINOS

API: Interfaz de programación de aplicaciones o Application Programming Interface por sus siglas en inglés.

Boceto 2D: Son una parte del proceso de diseño. Proporcionan las formas básicas a las que luego se le aplican operaciones de transformación para llevarlo a un boceto en 3D.

Cuadro de diálogo: Componente de una aplicación informática a través del cual el usuario interactúa con la aplicación. Está formada por ventanas, botones, menús, entre otros elementos.

Driver: Dentro del contexto de clases y paquetes de la tecnología OpenCASCADE, un driver se considera una clase que realiza operaciones bien definidas y que es utilizado por un mecanismo de comandos.

Entidades geométricas: Geometrías 2D (puntos, líneas, arcos, círculos, entre otros) y geometrías 3D (sólidos) utilizados por un modelador geométrico.

Extrusión: Operación 3D que genera un prisma a partir de una cara base.

Feature: Características de las entidades y objetos geométricos.

Filete: La unión de los flancos, el fondo y la cresta forman en su conjunto lo que es conocido como el prisma de la rosca o también llamado comúnmente filete.

Framework: Marco de trabajo.

IDE: Integrated Development Environment (Ambiente de desarrollo integrado).

Modelador geométrico: Parte de un sistema CAD que permite crear objetos en tres dimensiones a partir de las figuras básicas como rectas, círculos, triángulos y rectángulos entre otras.

Multiplataforma: Término utilizado frecuentemente en informática para indicar la capacidad o características de poder funcionar o mantener una interoperabilidad de forma similar en diferentes sistemas operativos o plataformas.

Previsualización: Proceso que se realiza cada vez que se cambia un datos para observar cómo se ajusta la textura a los datos que se vayan seleccionando.

Software: Término genérico que designa al conjunto de programas de distinto tipo (sistema operativo y aplicaciones diversas) que hacen posible operar con el ordenador.

Textura: Imagen que puede ser mapeada en una superficie determinada.

Tolerancia: Margen de error que admiten las medidas.

Truncado: Que se corta a una distancia determinada en las bases superiores o inferiores o en ambas.

Visor: Componente visual en el cual se visualizan las entidades geométricas y permite la interacción del usuario con los objetos visualizados.

