

CAPÍTULO 4

CATIA V5

GESTION DEL CONOCIMIENTO

MODULO KNOWLEDGEWARE

4.1. INTRODUCCIÓN

Para la aplicación del proyecto usaremos Catia v5, clasificado en capítulos anteriores como uno de los programas para desarrollar el conocimiento basado en la ingeniería.

Este programa se encuentra dividido en módulos que desarrollan un aspecto diferente del diseño del producto. El más interesante para este proyecto es el *Knowledgware*. Analizaremos sus submódulos y herramientas más importantes para el almacenamiento del conocimiento referente a nuestra aplicación.

4.2. CATIA V5.

Catia es un Programa de Diseño Integral, desarrollado por Dassault Systemes y distribuido IBM. El objetivo era crear una solución de Diseño que abarcara el **CAD/CAM/CAE/KBE/PDM** (Diseño /Mecanizado /Cálculo de Estructuras/Gestión del conocimiento/Gestión del producto). Para ello se han basado en Catia V4, líder en los sectores europeos aerospacial, automóvil y utilaje.

Catia V5 funciona bajo Windows lo que permite al usuario, integrar en su entorno de trabajo, todas las ventajas de este sistema operativo, sacrificando por el momento la estabilidad que tenía el entorno Unix.

El programa está concebido para satisfacer las necesidades de todos sus clientes, suministrándoles todas las herramientas que necesiten en su trabajo, por ello se ha pensado en un entorno modular, en el que el usuario compra lo que necesita y dónde siempre queda la puerta abierta a nuevos módulos.

¿Porqué tantos módulos?

La respuesta es que a pesar de Catia V5 ha nacido completamente en el entorno Windows, mira siempre hacia Catia V4 que tiene 160 productos. Los clientes presionan a Dassault para que en cada nueva Release se programen y mejoren módulos de V4.

Catia es un buen programa de modelado industrial, y lo están utilizando *PSA Citroën, Renault, Mercedes Benz, BMW,...* aunque en la versión 4. Se espera que en el plazo de uno o dos años, toda la industria del automóvil se pase a la versión 5.



Figura 4.1. Catia V5

4.2.1. Diseño Mecánico.

El módulo de **modelado sólido**, proporciona todas las herramientas para la creación de piezas, el análisis ángulos de salida y desmoldeos, análisis de curvaturas, propiedades físicas, etc. Este módulo está directamente relacionado con el de **ensamblajes**. Ya que casi siempre nuestra pieza ha de ir incluida en un producto final. Catia incorpora el método de diseño Top- Down, *arriba-abajo*, en el que se pueden diseñar piezas en el entorno de conjunto filtrando la geometría que consideremos conveniente del Ensamblaje.



Figura 4.2. Diseño Mecánico.

La **generación de planos** es automática, fácil e intuitiva. Si el diseño se ha parametrizado correctamente, la aplicación recuperará las cotas del modelo 3D.

Las vistas, se hacen de forma automática, podemos incluir holguras tolerancias, cambiar el formato de las cotas, según muchos de los estándares.

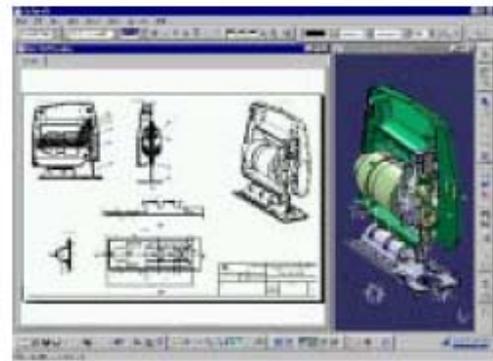


Figura 4.3. Generación de planos

Funciona muy bien generando planos de montaje, ya que en un mismo plano podemos tener distintas configuraciones de producto, que se corresponderán con las distintas fases de montaje de este.

Tenemos también dos **módulos de chapa**.

Un módulo de diseño con muchas operaciones específicas en chapa y comandos directos. Permite la ingeniería simultánea entre las representaciones plegadas y desplegadas de la pieza.

Un módulo de producción con el que se puede estudiar la viabilidad de cualquier pieza para ser construida en chapa.



Figura 4.4. Conformado de chapa

Otro módulo interesante es el de Diseño de **estructuras metálicas**. Con él se pueden construir directamente estructuras lineales, curvas y placas. Los perfiles se escogen de una librería en la que están los principales estándares del mercado (IPN, IPE, UPN, UAP, CAE,...).

Catia cuenta además con un **módulo de moldes**.

La información en el árbol en este caso se ordena automáticamente, y en la salida a planos tendremos los componentes perfectamente gestionados (casquillos, guías, expulsores, boquillas, etc,...). Sin embargo el programa ofrece poca ayuda a la hora de determinar el plano de partición, tarea que hasta al momento ha de realizarla el usuario.



Figura 4.5. Moldes

Esta herramienta utilizada conjuntamente con los módulos de mecanizado, puede convertirse en una opción interesante para empresas dedicadas a los moldes, pero por el momento aun ha de mejorar bastante.

4.2.2. Diseño de Formas y Estilo.

Esta es la parte más fuerte de CATIA V5. Para empezar Catia dispone de un módulo específico de Digitalizado, que aunque por si mismo se queda muy corto en opciones, es sencillo e intuitivo. Al estar perfectamente integrado con los demás módulos, tenemos toda la potencia de las superficies paramétricas.

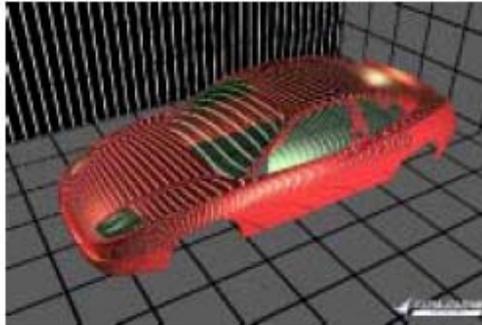


Figura 4.6. Diseño de formas

Con este tipo de superficies se pueden hacer modificaciones en los perfiles en cualquier momento, y estas modificaciones se propagan a todo el diseño. Disponemos de herramientas de análisis de superficies, inflexión de curvas, curvaturas, distancias.

Catia controla todos los parámetros de unión en superficies, como la tensión en la superficies de unión, la continuidad, de estas, factores de suavizado, etc. El módulo más importante en este sector es el GSD (Generative Shape Desing). En este entorno el usuario puede crear formas avanzadas basadas en la combinación de la estructura alámbrica y las múltiples características de las superficies. Proporciona un completo conjunto de herramienta de para crear y modificar las superficies mecánicas utilizadas en el diseño de formas complejas y piezas híbridas. En nuevas versiones y en configuraciones de programa avanzadas se influye un módulo llamado **superficies de clase A**, con las que se crea geometría con unas especificaciones muy exigentes, con las que satisfacer a la industria automovilística y aeronáutica. Véase la calidad en las superficies de esta aeronave, y la integración con el chasis.



Figura 4.7. Superficies avanzadas

Para modeladores más avanzados Catia dispone de un módulo de superficies de estilo libre, en el que se trabaja con NURBS y Bezier, los diseñadores pueden manipular de forma directa las superficies y conservar las asociaciones de forma subyacente. Vinculado a este módulo tenemos otro con el que podemos crear geometría a partir de modelos 2D. Este módulo, es especialmente útil cuando detrás de un Diseño hay un trabajo grande de bocetado por parte de los estilistas.

4.2.3. Análisis.

Realiza análisis de esfuerzos y de vibración en conjuntos para conjuntos con el fin de generar diseños de alta calidad en poco tiempo. No es un programa estricto de cálculo por elementos finitos, ya que hay muchos parámetros que no se pueden controlar, pero simplifica enormemente el análisis, sacrificando precisión. El programa genera reportes en modo automático, con representaciones gráficas de las tensiones y deformaciones. Las nuevas actualizaciones del programa incorporan un módulo de mallado, que permite calcular algunos parámetros y realizar cálculo a superficies.

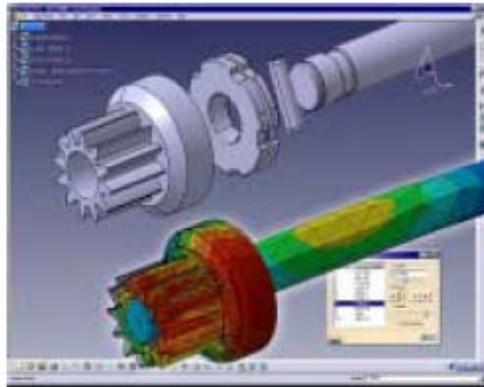


Figura 4.8. Análisis

4.2.4. Mecanizado.

CATIA V5, dispone de tres módulos de mecanizado, que generan el programa para nuestra máquina de control numérico que podemos escoger en la base de datos, donde están los principales postprocesadores del mercado.

Los módulos de los que dispone el programa son: Mecanizado prismático en 2,5 ejes, de superficies 3 ejes, y superficies en 5 ejes.



Figura 4.9. Mecanizado

4.2.5. Equipamiento e Ingeniería de sistemas.

Conforman un conjunto de aplicaciones orientadas a los ingenieros, que sirven de ayuda para sacar toda la información posible a un producto.

Pongamos el ejemplo de un ingeniero que quiere cablear un coche. Utilizando el módulo de **ruteado eléctrico**, el ingeniero puede obtener información de la cantidad de cable que va a necesitar de cada tipo, si habrá interferencias en las distintas zonas del coche, si habrá o no problemas a la hora de cambiar ese componente...

Podemos optimizar también el ruteado de sistemas de tuberías, diseñar y revisar sistemas eléctricos, etc.

Otros módulos incluidos en esta sección son:

- Diseño de placas electrónicas y dispositivos eléctricos, donde estamos auxiliados por una extensa librería de componentes estándar, como conectores, resistencias, condensadores, etc.
- Librería de componentes eléctricos: crea y gestiona catálogos de dispositivos eléctricos.
- Diagramas de sistemas, son diagramas 2D de sistemas de tuberías.
- Instrumentación y tuberías: Crea y gestiona diseños lógicos de sistemas de tuberías utilizando las convecciones, terminología y practicas estándar en el sector.
- Diagramas HVAC: Crea y gestiona diseños lógicos de climatización y ventilación.
- Disposición en Planta: Permite organizar y distribuir una planta de producción.

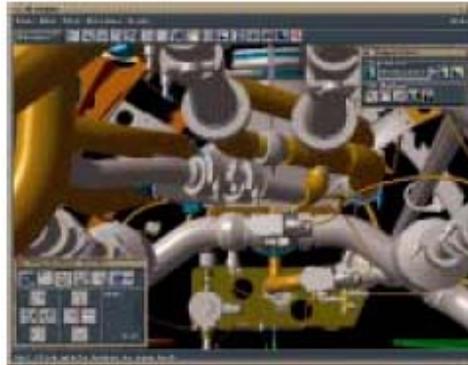


Figura 4.10. Equipamientos

4.2.6. Síntesis de Productos.

- **Simulador de montaje de conjuntos:** Este módulo dispone de las herramientas necesarias para hacer una validación del montaje y ensamblaje de piezas. Genera información útil sobre el espacio reservado a las operaciones de desmontaje, con el fin que se tenga en cuenta en futuras modificaciones del diseño.
- **Simulador de Mecanismos:** Define mecanismos utilizando gran variedad de tipos de acoplamientos o bien generándolos automáticamente a partir de restricciones de conjuntos mecánicos. Comprobamos las interferencias las distancias mínimas. Está pensado para todo tipo de industrias.
- **Render:** Este módulo permite situar piezas o conjuntos en un entorno, en el que manipulando puntos de luz y las propiedades de los materiales creamos imágenes fotorrealísticas de nuestras piezas.



Figura 4.12. Síntesis de productos

- **Gestión del conocimiento:** También llamado KBE. Las piezas generadas por ingeniería basada en el conocimiento, incorporan en al Diseño, el comportamiento, las fórmulas, las reglas, y todo lo que el ingeniero considere necesario, en la realización de una pieza de ese tipo. Los diseñadores de una empresa no tienen los conocimientos técnicos del ingeniero. Si un producto se diseña bien por este método, el programa le guiará avisándole cuando no cumpla las especificaciones que se han impuesto a esa pieza.

4.3. GESTION DEL CONOCIMIENTO (KNOWLEDGEWARE)

El conocimiento es la acumulación de experiencia y datos que se usan de forma directa. El producto es conocimiento. Los modelos y dibujos del producto son parte de los datos que lo definen, pero observando estos datos no llegamos a saber cómo o por qué alguien creó el producto de una manera específica.

Es difícil incluir el conocimiento en un plan, porque hay que hacer un esfuerzo extra para escribir fuera o codificar el proceso de decisión para que pueda pasarse como conocimiento de una persona a otra.

Las herramientas que se usan a menudo en el desarrollo de un producto necesitan ser capaces de capturar conocimiento, y deben permitirles a los diseñadores aplicar ese conocimiento a las tareas de la creación de otro nuevo producto.

CATIA Versión 5, proporciona una serie de capacidades muy interesantes que permiten a los diseñadores crear sus propias especificaciones y reglas, salvándolas en documentos, y reutilizándolas de nuevo.

Las aplicaciones que se necesitan para desarrollar soluciones del Conocimiento se han dividido en tres partes:

- El consejero del conocimiento (Knowledge Advisor)
- El conocimiento experto (Knowledge Expert)
- Plantillas de producto del conocimiento (Product Knowledge Template)

4.3.1. KNOWLEDGE ADVISOR

El Consejero del Conocimiento es un producto de CATIA que permite a los usuarios incluir el conocimiento dentro de un plan de trabajo e influenciar en él para ayudar al diseñador a tomar decisiones en el orden de reducir errores o automatizar dicho plan, para que se pueda obtener el máximo de productividad.

Los usuarios pueden aportar conocimiento en un plan de diseño tales como conductas, fórmulas, reglas y chequeos y acudir a ellos cuando se requiere en cualquier momento. Su significado también es accesible: Por ejemplo un intento de chequeo puede resaltar los parámetros involucrados en una comprobación, siendo fácil e inmediato de entender de qué manera una norma se ha violado.

Para abreviar, el Consejero de Conocimiento ayuda a los usuarios a:

- Capturar el conocimiento de la ingeniería como especificaciones que permiten dar a la aplicación una consistencia completa.
- Definir y compartir fácilmente la habilidad de diseño entre todos los usuarios.
- Automatizar la definición del producto.
- Aumentar la productividad.
- Permitir dar una atención temprana a especificaciones del plan de diseño que previenen un costoso rediseño.
- Guiar y ayudar a los usuarios a través de sus tareas del plan de diseño.

Este módulo introduce nuevos conceptos que se van a definir a continuación especificando su funcionalidad y cómo y cuándo se utilizan.

4.3.1.1. El entorno



Los parámetros y las fórmulas se utilizan para crear parámetros nuevos y relaciones entre ellos.



Las tablas de designación nos dan la capacidad de manejar familias de componentes



El inspector del conocimiento pregunta designaciones para determinar y prever los resultados al cambiar algunos parámetros.



El icono de la Regla permite crear reglas, escribir códigos y textos con sintaxis y aplicarlas a nuestro documento.



Los Chequeos al igual que las reglas se crean con un código y se aplican al documento.



El icono de la Reacción permite crear una escritura que especifica cómo cambiar algún rasgo que atribuye cuando un evento ocurre.



El Juego de icono de los Parámetros permite crear juegos de parámetros. Estos juegos de parámetros son agrupados debajo del nodo de los Parámetros



El icono de VB permite que se lancen unos argumentos de ingenio de se han creado en una macro



El icono de Acción permite crear una escritura para la macro.



El icono de la Lista le permite crear una lista de rasgos que se localizará bajo el nodo de los Parámetros en el árbol de la especificación.

4.3.1.2. Parámetros.

Los parámetros son elementos que permiten controlar variables que están destinadas a formar parte de algún aspecto del diseño.

En Catia se describen fundamentalmente dos tipos de parámetros:

- *Los parámetros Intrínsecos al sistema* que se definen automáticamente cuando se van generando objetos y geometrías y que permanecen en la estructura interna de Catia. No son editables ni podemos hacer que aparezcan en el árbol, solo son visibles cuando desplegamos el icono de la fórmula donde aparecen todos estos parámetros que se han ido generando a medida que vamos creando elementos.

- *Los parámetros del Usuario* que son los que se crean desde el exterior y que pueden ser de muchos tipos, parámetros de medida, de masa, reales, incluso booleanos. Pueden tener un solo valor, o múltiples valores, o llevar insertada alguna fórmula.

Con el icono *Formula*, , se despliega la caja de diálogo donde se puede seleccionar el tipo de parámetro y darle un valor. Este parámetro puede ser renombrado y cuando sea creado aparece en el árbol en una lista donde se incluyen todos los parámetros. Lo más importante que tiene su uso es la posibilidad que da al diseñador de variar su valor después de ser creado.

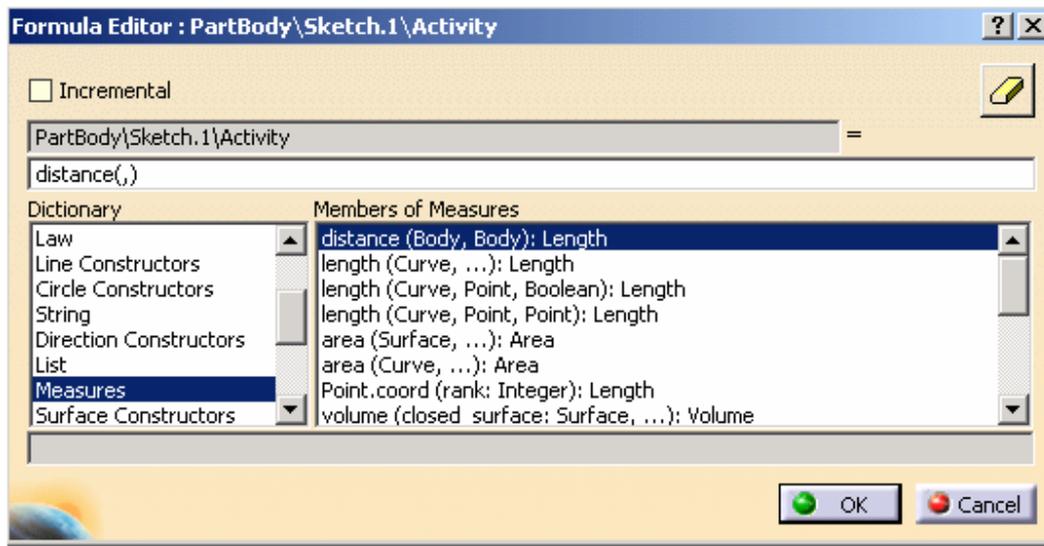


Figura 4.13. Parámetros.

- *New Parameter of type*: aparecen todos los tipos de parámetros que tiene Catia y hay que seleccionar antes el que se quiere y luego crearlo.
- *With*: los parámetros pueden tener un solo valor o más de un valor que se le asignan con una tabla.
- *Filter*: se puede hacer un filtro por tipo de parámetro para mejorar el proceso de búsqueda.

Un parámetro puede llevar como definición una fórmula y esto nos evita tener que estar haciendo cálculos innecesarios y repetitivos. Para añadir una fórmula se tiene el botón *Add Formula* donde aparece el editor de fórmulas que tiene Catia.

Las funciones principales de los parámetros son:

- Pueden servir para agregar información específica a un documento.
- Pueden definirse o pueden ser reprimidos por relaciones.
- Pueden usarse como los argumentos de una relación.

Una relación dada puede tomar como sus argumentos ambos tipos de parámetros (intrínseco y usuario).

4.3.1.3. Fórmulas.

Las fórmulas son los elementos de enlace entre los parámetros y los elementos que conforman el diseño.

Catia tiene su propio editor de fórmulas que contiene no sólo una lista con todos los parámetros que contiene el modelo, sino también con una serie de funciones que se pueden utilizar para definir la fórmula. Dependiendo de qué categoría los objetos son referenciados dentro de la fórmula.

Para insertar una definición en el editor de fórmulas, se puede o bien hacer doble -click del objeto dentro del diccionario o desde el árbol.

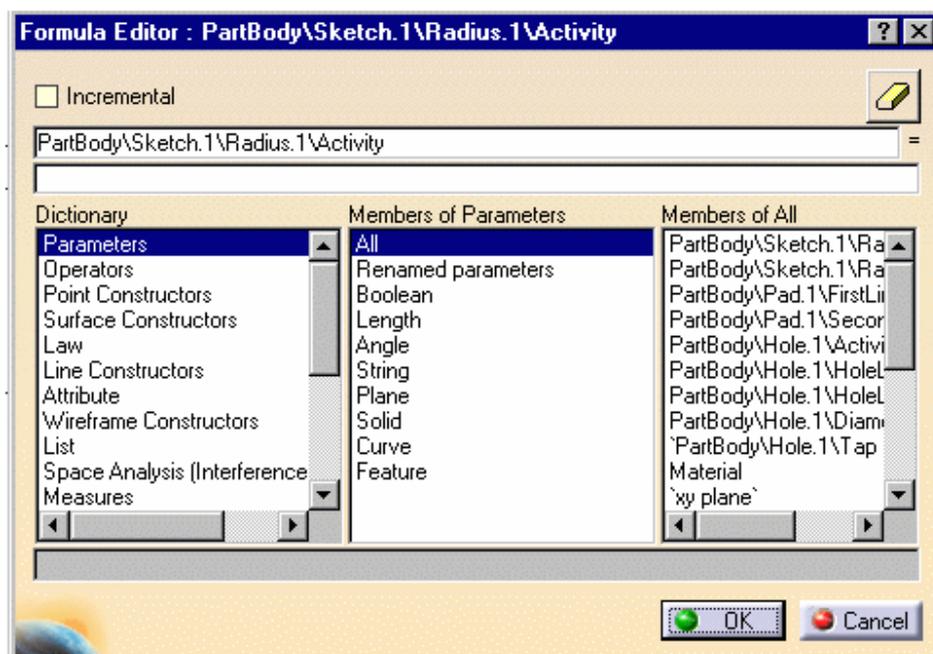


Figura 4.14. Fórmulas

Cuando se selecciona una de las fórmulas que vienen en el diccionario viene especificado el tipo de argumento que es necesario para su aplicación.

Por ejemplo, si se selecciona el apartado de *Measures*,

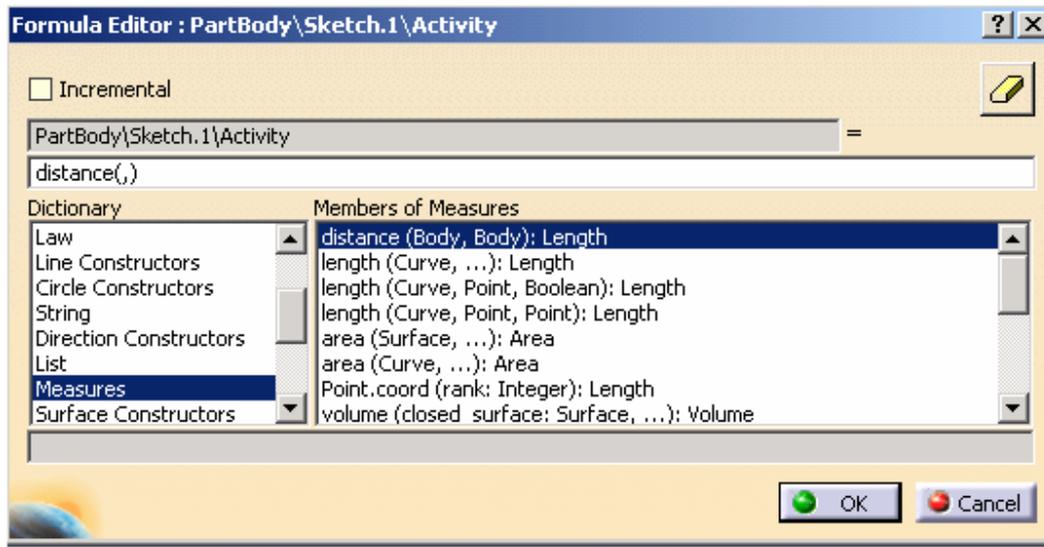


Figura 4.15. Editor de fórmulas

medidas, tenemos varias posibilidades, pero al elegir una te indica que tipo de elementos son necesarios para su construcción, en este caso dos Bodies.

Hay que tener mucho cuidado cuando creamos la fórmula que las unidades de los parámetros que incluimos sean las correctas.

4.3.1.4. Listas.

La ventana de Edición de Lista le permite al usuario que maneje los objetos que él quiere agregar a la lista que se está creando.

Se accede pulsando el botón del icono *List*, .

La ventana contiene cuatro botones diferentes.

- **Add** (añadir) Le permite al usuario que agregue los artículos que él seleccionó en el árbol de la especificación o en la geometría a la lista.
- **Remove** (eliminar) Le permite al usuario que quite artículos de la lista.
- **Move Up** (mover hacia arriba) Le permite al usuario que suba artículos en la lista.
- **Move Down** (mover hacia abajo) Le permite al usuario que baje artículos en la lista.

El Número de campo de los Elementos despliega el número de artículos contenido en la lista.

El área opuesta a los cuatro despliegues de los botones, los artículos que pertenecen a la lista.

Si se selecciona un artículo en la Lista, y en otro artículo en el árbol de la especificación o en la geometría se pulsa el botón *Agregue*, el artículo de la Lista se reemplazará con el que se ha agregado.

4.3.1.5. Lazos.

Los lazos pueden manejar la creación, modificación y agrupaciones de juegos de rasgos que se guardan en el modelo resultante como un solo rasgo. Un cambio en sus especificaciones tendrá como consecuencia la modificación esperada en el modelo.

Los lazos pueden ser instanciados a través de un PowerCopy o una UserFeature lo que hace más sencilla su utilización
Aparece la siguiente ventana de diálogo.

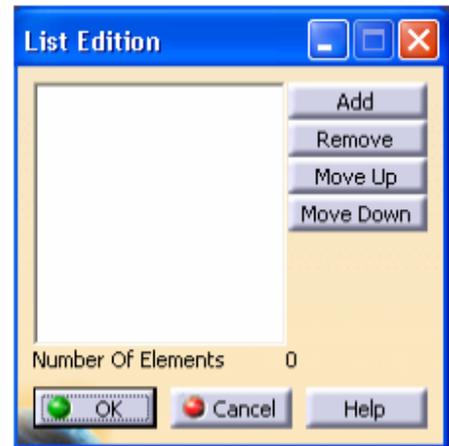


Figura 4.16. List

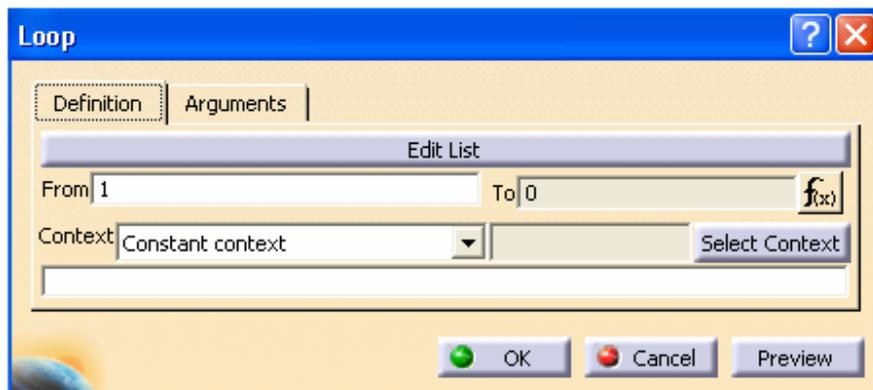


Figura 4.17. Definición del lazo

En el contexto podemos tener:

- Contexto constante: el contexto es una porción del documento.
- Iterante: El elemento itinerante puede ser basado en un número constante o en una lista de rasgos.

Estos elementos se van seleccionando directamente del árbol.

Una vez seleccionados los objetos podemos editar las fórmulas:

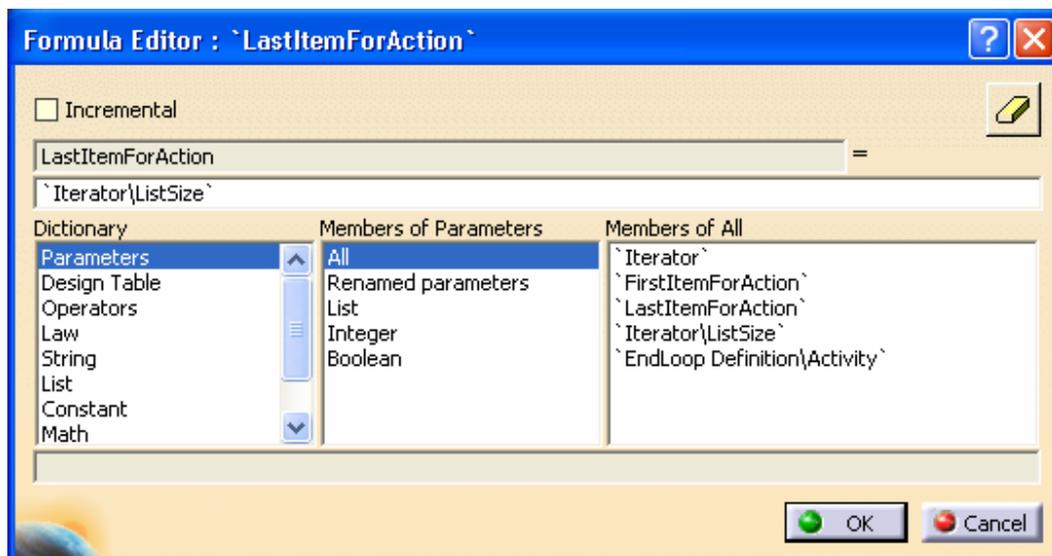


Figura 4.18. Editor de fórmulas

Con esta función lo que conseguimos es ir aplicando acciones o fórmulas en vez de a objetos independientes a listas de objetos que se han sido seleccionadas previamente.

En los argumentos se especifica el Pad al que queremos añadir el valor de una variable con lo que se crea un nuevo pad dentro de nuestro OpenBody con el nuevo valor que le hemos asignado y además se va a crear otro nuevo Body que también contiene al nuevo Pad.

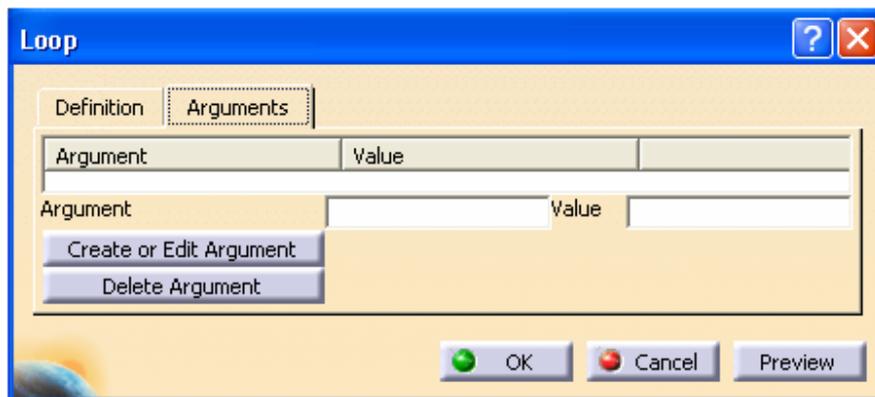


Figura 4.19. Argumentos del lazo

4.3.1.6. Tablas

Cuando tenemos una serie de parámetros que hacen referencia a un mismo conjunto de datos, o que la construcción geométrica a la que hacen referencia esté normalizada y se recogen los resultados en una tabla, podemos crearla de tal forma que sus valores cambien en función de los datos contenidos en ella.

Tener los parámetros de esta forma agrupada es de mucha utilidad, ya que se cumplen dos funciones simultáneas, una de ellas es tener una estructura de árbol más ordenada, y la otra reducir el tamaño de elementos que de él cuelgan.

Lo más importante que se consigue teniendo los datos agrupados en tablas es darnos la posibilidad de cambiar los valores o incluir otros valores nuevos sin más que editarla.

La tabla se crea con el icono, *Design table*, .

Se abre la ventana:

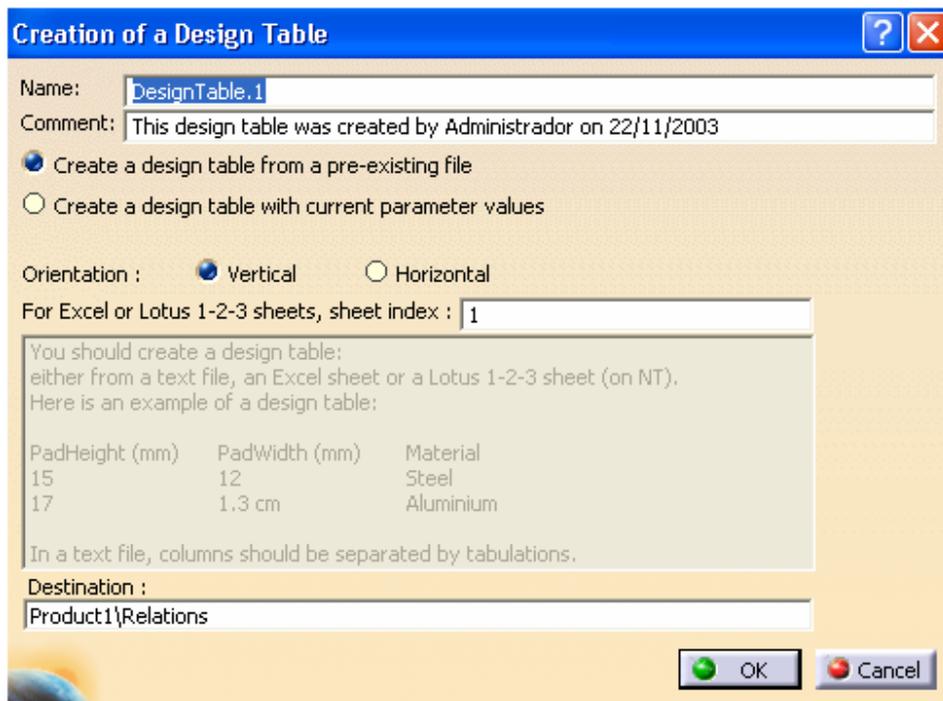


Figura 4.20. Creación de tablas.

Se elige la opción de crear una tabla nueva (*create a design table with current parameter values*) y pinchando en OK se abre la lista de parámetros que tenemos creados anteriormente.

Una vez seleccionados los parámetros, la tabla se almacena en un archivo a parte, esto hay que tenerlo en cuenta a la hora de salvar el modelo porque tendremos que incluir también el documento donde se ha creado la tabla.

El archivo que se crea puede ser del tipo Microsoft Excel (.xls), o bien un archivo de texto (.txt).

El fichero es creado en el directorio elegido, y el documento Catia hará una ligadura a éste.

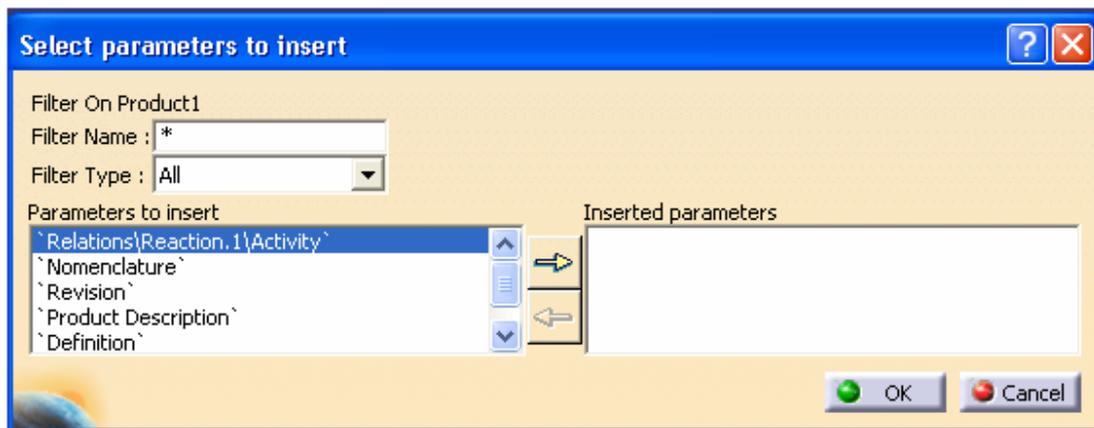


Figura 4.21. tablas

También se puede asociar al archivo Part una tabla existente con la opción (Create design table from a pre-existing file) asociando los parámetros a las columnas de la tabla.

4.3.1.7. Reglas.

Las reglas dan la posibilidad al usuario de obligar al modelo a que siga unas determinadas pautas.

Son fundamentalmente de dos tipos:

- Agrupaciones de fórmulas: se consigue así sobre todo tener un árbol más ordenado y con menos elementos.
- Reglas creadas por el propio usuario con un lenguaje intrínseco de Catia y que sigue la estructura de un lenguaje sencillo de programación. Como el Visual Basic.

Normalmente es conveniente crear una Regla cuando tenemos un modelo que va a seguir unas pautas si cumple unas condiciones, y otras distintas si cumple otras condiciones. Para acciones de más relevancia habría que crear una Macro que es un programa más complicado en Visual Basic y no es objeto de nuestro proyecto.

Creación de una regla, pulsando sobre el icono *Ruler*, .

En la ventana de la regla aparece:

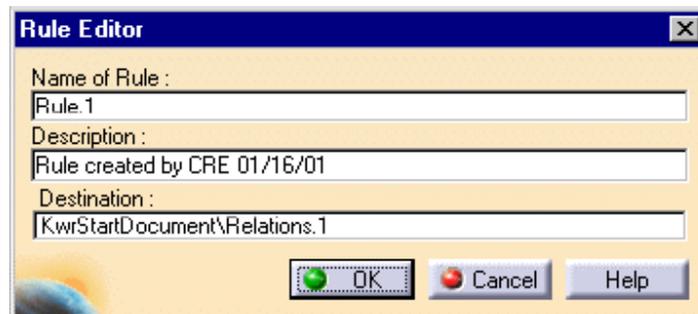


Figura 4.22. Editor de Reglas

- El nombre de la regla: *Rule.1*. La primera regla creada en un documento es por defecto Regla.1. Este nombre es el desplegado en el árbol de la especificación a menos que se modifique el nombre predefinido a la creación.
- El usuario y la fecha de creación.
- El destino: una regla puede hacer referencias a especificaciones de fórmulas que lleve agregado el Part en su raíz o puede incluirse dentro de un nivel más bajo como un Body que esté insertado en el Part principal.

Los *Keywords* son las estructuras básicas de las que disponemos para construir la regla.

Los *Operators* son los operadores booleanos.



Figura 4.23. Keywords and operators

4.3.1.8. Chequeos

Los chequeos sirven para hacer comprobaciones sobre las definiciones que tienen que cumplir los elementos, así se pueden detectar si se está incumpliendo alguna normativa o alguna regla que lleva incluido el modelo.

Un chequeo no modifica el documento al que se aplica. Un chequeo es un rasgo. En el árbol de especificación de documento, se despliega como una relación que puede activarse y puede dejarse fuera de funcionamiento. Como cualquier rasgo, un chequeo puede manipularse de su menú contextual.

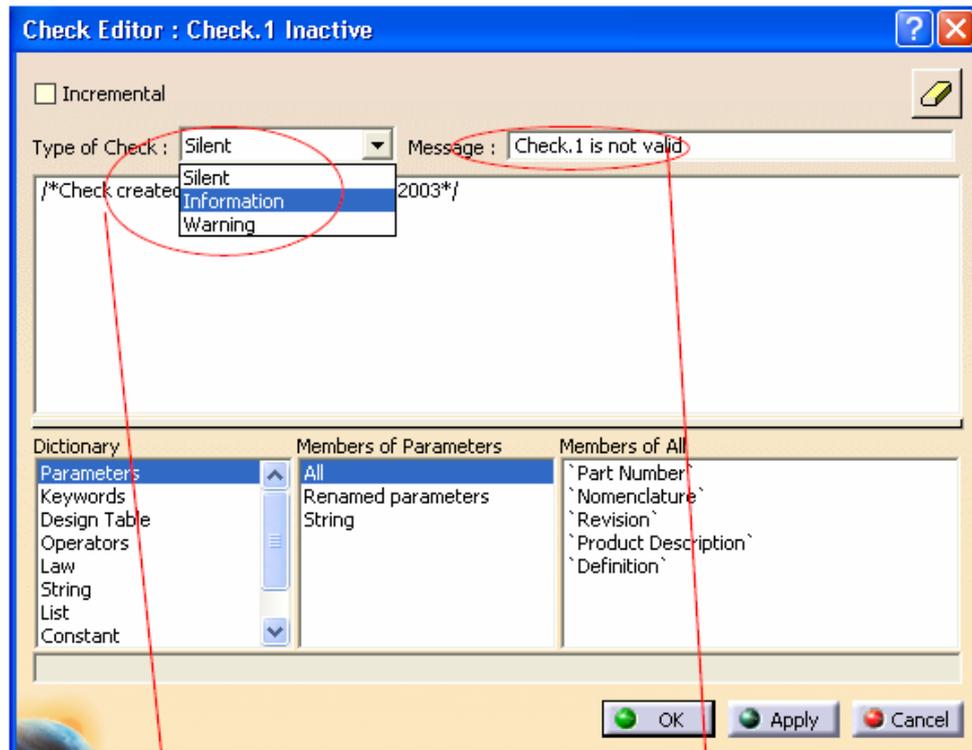
Por ejemplo, si se está trabajando con piezas menores de 20 kilogramos podemos crear un chequeo que detecte si alguna de las piezas sobrepasa este peso y si así se edita un mensaje de aviso.

La ventana de diálogo que se abre en un chequeo es muy parecida a la de la regla, icono *Check*, .



Figura 4.24. Editor de chequeos

También es conveniente cambiar el nombre del chequeo por alguno que dé cierta información sobre el contenido del mismo.



El chequeo fundamentalmente puede ser de estos tres tipos:
Silent: no daría ningún mensaje
Information: solo da un mensaje de información.
Warning: el mensaje es de alarma

Aquí se escribe el mensaje que queremos que aparezca cuando el chequeo se lleve a cabo

Figura 4.25. Parámetros del chequeo

Este mensaje será desplegado siempre que la declaración especificada por el chequeo no se cumple, además de un icono rojo que aparecerá en el árbol. Un icono verde en el árbol de especificación aparece si el chequeo se cumple y ningún mensaje se despliega.

4.3.1.9. Macros con argumentos (VB Script)

Una Macro no es más que una aplicación creada con un lenguaje de programación que podemos construir con un editor de texto que aparece al pulsar el icono *Macros with arguments*, , o bien podemos crearla desde un programa en Visual Basic en un documento aparte y luego llamarla desde Catia a través de una acción.

Para construir una Macro o VB Script el usuario debe conocer el lenguaje de la programación, no es como construir una regla que trae un diccionario con los parámetros y operadores que se utilizan.

Las macros es la herramienta más avanzada del conocimiento ya que permite el uso de la programación y hoy en día es difícil encontrar una acción que no pueda ser programable.

Este editor lo que permite es:

- Especificar argumentos
- Llevar una definición del rasgo delante del editor sólo, o seleccionando el rasgo en el árbol o en el área de la geometría.

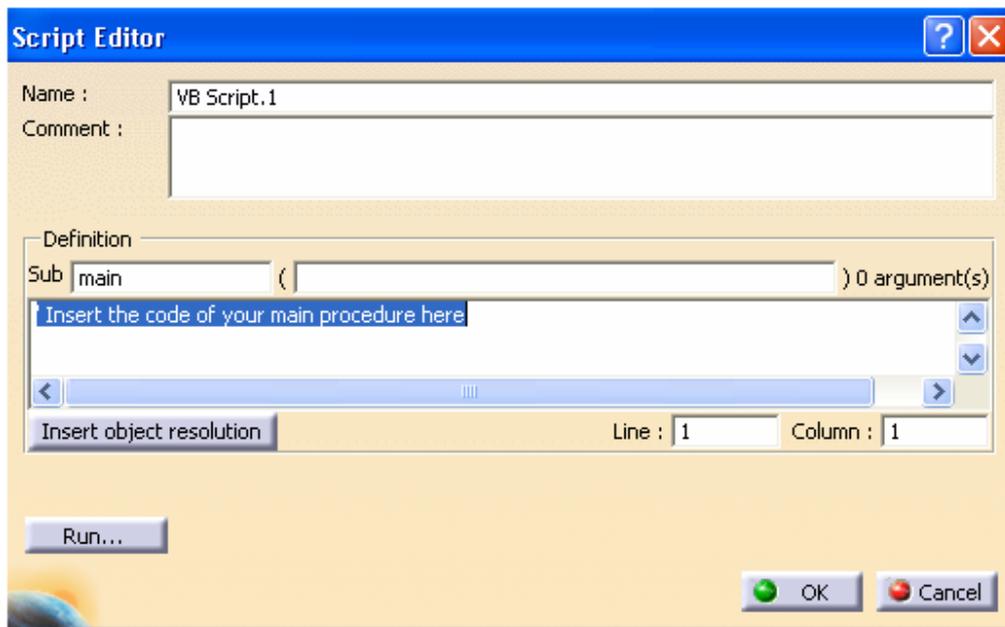


Figura 4.26. Macros, VB Script

4.3.2. KNOWLEDGE EXPERT

Es una herramienta para analizar y manipular todo un producto. Tiene las mismas capacidades que el módulo de Knowledge Advisor pero tratada de una forma un poco diferente, mientras que en el Knowledge Advisor se crean aplicaciones que contienen características o propiedades del modelo y se almacenan dentro del mismo modelo, en el Knowledge Expert el usuario que llamamos “el experto” puede generar una Regla Base.

Esta Regla Base es la que va a funcionar como regla estándar para el resto de usuarios que la vayan a usar y cada vez que quieran crear una regla nueva tienen que ir añadiéndola a la anterior, teniendo al final un conjunto de reglas.

Cada usuario puede manipular a su antojo las reglas que están situadas en el segundo nivel, pero no pueden manipular la Regla Base. Esta sólo puede ser manipulada por el usuario que la creó. Si se desactiva esta regla quedan automáticamente desactivadas todas las demás.

Con esto se consigue ir almacenando información con la finalidad de ir catalogando todo un conjunto de información y normativas que posteriormente pueden ser usadas por los usuarios que tenga la empresa.

Así mismo se crean los Chequeos Genéricos que tienen el propósito de permitir dar diagnósticos y revisar, ficheros para encontrar errores.

El Conocimiento Experto permite utilizar varios lenguajes de programación además del propio lenguaje de escritura que incluye Catia como por ejemplo Visual Basic, CAA....

4.3.2.1. El entorno.

Cuando entramos en el entorno del Módulo del Conocimiento Experto podemos encontrar la siguiente barra de herramientas:



Este es el icono que accede al editor de la Regla Experta. En el editor se puede escribir el código que forma la regla, hacer test incluyendo su sintaxis y añadir la regla creada a la Regla Base.



Accede al editor del Chequeo Experto. Con este icono creamos el chequeo con su correspondiente código y también se añade a la Regla Base.



Crea una Regla de Juego debajo de la Regla Base en las especificaciones del árbol.



Inserta una Regla Base desde un documento externo.



Genera un Chequeo de informe. Utilizar este tipo de chequeo tiene un cierto interés cuando se tiene una Regla Base con un cierto número de chequeos aplicados a características distintas del mismo documento. El informe da información de cumplimiento o no cumplimiento de los distintos chequeos. Esta información depende a su vez del escenario de la Regla Base.

4.3.2.2. Estructura del Conocimiento Experto.

Las reglas y chequeos que se crean en el Conocimiento Experto se distribuyen en una jerarquía donde la raíz es la Regla Base y de ella cuelgan las Reglas que se crean con el Módulo del Knowledge Advisor, las Reglas y Chequeos Expertos, y las Reglas de Juego.



Figura 4.27. Estructura del conocimiento experto

4.3.2.3. Regla base.

El Conocimiento Experto permite crear y manipular tanto relaciones como tipos de características. Estas características particulares son organizadas dentro de una jerarquía. La Regla Base es la raíz de la jerarquía. Las reglas y chequeos expertos son objetos terminales, y entre medio las reglas y chequeos normales que se construyen con el modulo de Knowledge Advisor.

Para tener acceso a seleccionar alguna de estas características no se puede ir directamente a la geometría del dibujo, sino que hay que acceder a través de la especificaciones del árbol de Catia.

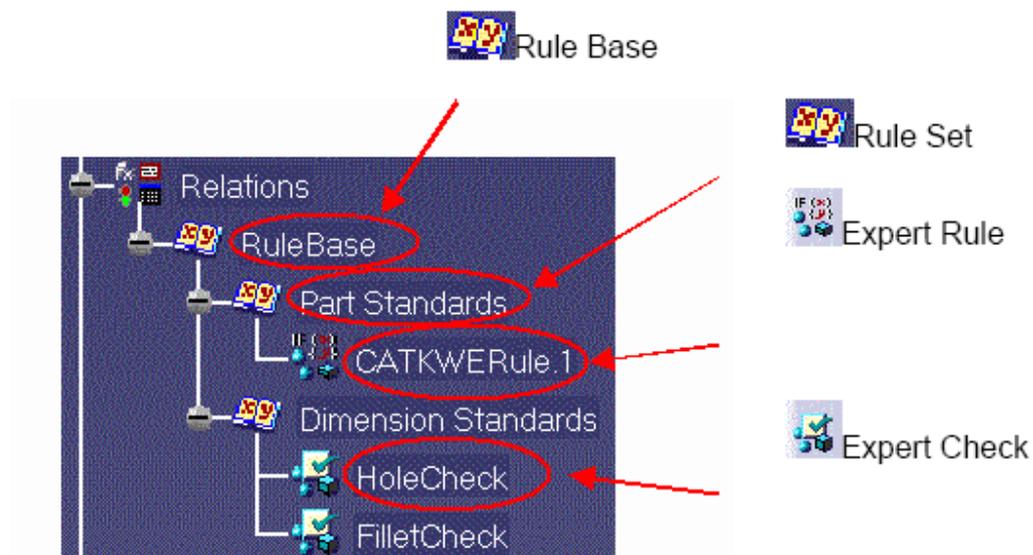


Figura 4.28. Regla base

Las reglas y chequeos son objetos que el usuario crea directamente. Una Regla Base vacía está automáticamente asociada a un documento cuando se accede al entorno de Knowledge Expert, y una regla normal no puede ser creada dentro de este entorno sin estar dentro de una Regla Base.

Al aplicar una Regla Base en otro documento esta tiene que estar guardada en un CatProduct, porque se está haciendo referencia a un conjunto de reglas y chequeos y por lo tanto tiene estructura de producto, sin embargo el documento donde se crea la Regla Base puede ser tanto un CatProduct como un CatPart. Hay que tener cuidado porque en el documento donde queremos aplicar la Regla Base no puede haber otra regla o chequeo con el mismo nombre.

Una vez aplicada, el nuevo documento guarda relación con el antiguo donde ha sido creada la regla de tal manera que si hacemos algún cambio en el documento antiguo quedarán reflejados automáticamente en el nuevo documento.

Hay tres formas de aplicar una Regla Base:

- Aplicar la Regla Base desde un documento o un catálogo y guardarla automáticamente. (*Use Only*)
- Copiar la Regla Base en un documento así como sus componentes (reglas de juego, reglas expertas y chequeos). No se guarda relación alguna con la Regla original. (*Import*)
- Importar la Regla Base con una ligadura a la Regla original, con lo cual, al modificar cualquier parámetro de la regla original se modifica automáticamente la Regla Base. (*Import with link*)

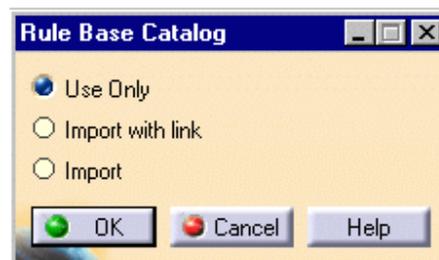


Figura 4.29. Aplicación de una regla base

Otra característica que tiene la Regla Base es que se activa y desactiva en el propio documento.

Hay dos formas de hacerlo:

- En las especificaciones del árbol con el botón derecho sobre la Regla Base seleccionamos: *rule base object -> (De) activate*.
- En la lista de parámetros se selecciona el que define la actividad de la regla 'rule base \ activity' que puede modificarse dándole un valor.

4.3.2.4. Regla de juegos.

El propósito de una Regla de Juego es unir reglas y chequeos que hacen referencia a una característica o conjuntos de características del mismo tipo. Las relaciones cobran significado cuando se van agrupando, así podemos hacer un conjunto de chequeos que hagan referencia a un mismo objeto y darle a ese objeto varias restricciones. El Juego de Reglas no puede ser creado sin la existencia de una Regla Base.

Cuando se crea una Regla de Juego esta se sitúa debajo de la Regla Base en el árbol de especificaciones y así se mantiene la jerarquía. En el editor se puede tanto escribir código como agrupar reglas y chequeos ya creados.



Figura 4.30. Regla de juego.

4.3.2.5. Regla Experta.

Una regla experta es un conjunto de instrucciones las cuales ejecutan un grupo de afirmaciones que dependen de un valor de una o más expresiones y se aplican a todos los elementos referenciados en las instrucciones.

Una Regla Experta permite manipular las características de los elementos. Por ejemplo, algunos valores de las características pueden ser examinados y dependiendo del resultado pueden ser reevaluados. No todos los elementos, ni todas las características pueden ser manipuladas por estas reglas.

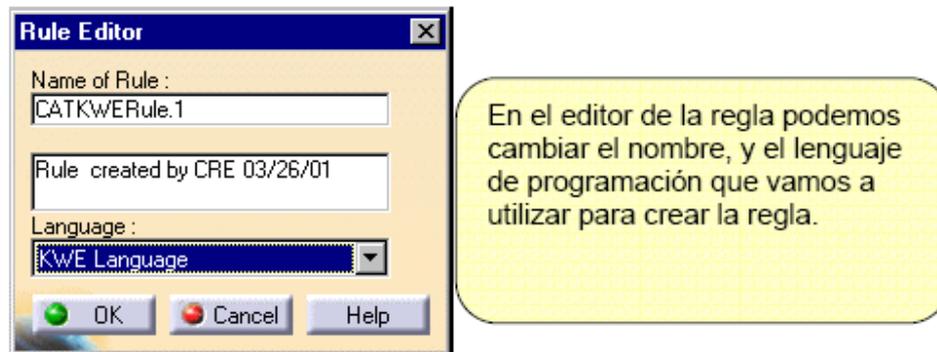


Figura 4.31. Regla Experta.

Para aplicar una Regla Experta que ha sido creada, debe haberse salvado antes la Regla Base en el documento.

Una vez nombrada la regla aparece el siguiente editor de texto:

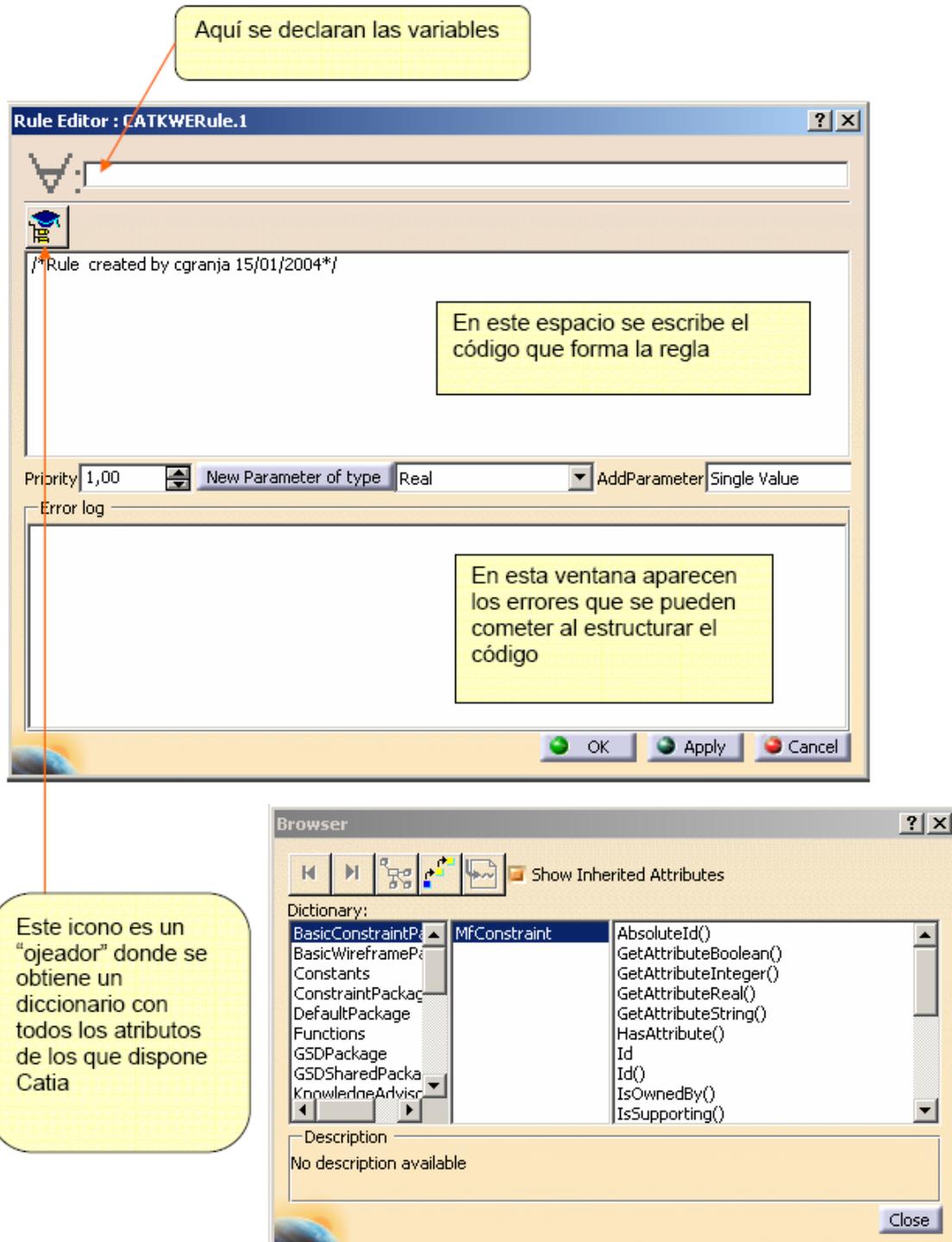


Figura 4.32. Parámetros de una regla experta

En el editor de la regla podemos cambiar el nombre, y el lenguaje de programación que vamos a utilizar para crear la regla.

4.3.2.6. Chequeo Experto

Un chequeo experto es una regla que solo verifica si una condición es verdadera, para todos los objetos que se encuentren en el documento.

No es modificable dentro del documento donde se está aplicando, solo nos da el resultado de si el chequeo has sido válido o no apareciendo el icono del chequeo en el árbol verde o rojo respectivamente.

El editor de un chequeo es exactamente igual que el de una Regla, y se crea de la misma forma, en lo único que varía es en el siguiente menú que se divide en tres áreas:

- *Condition*
- *Correction*
- *Report*

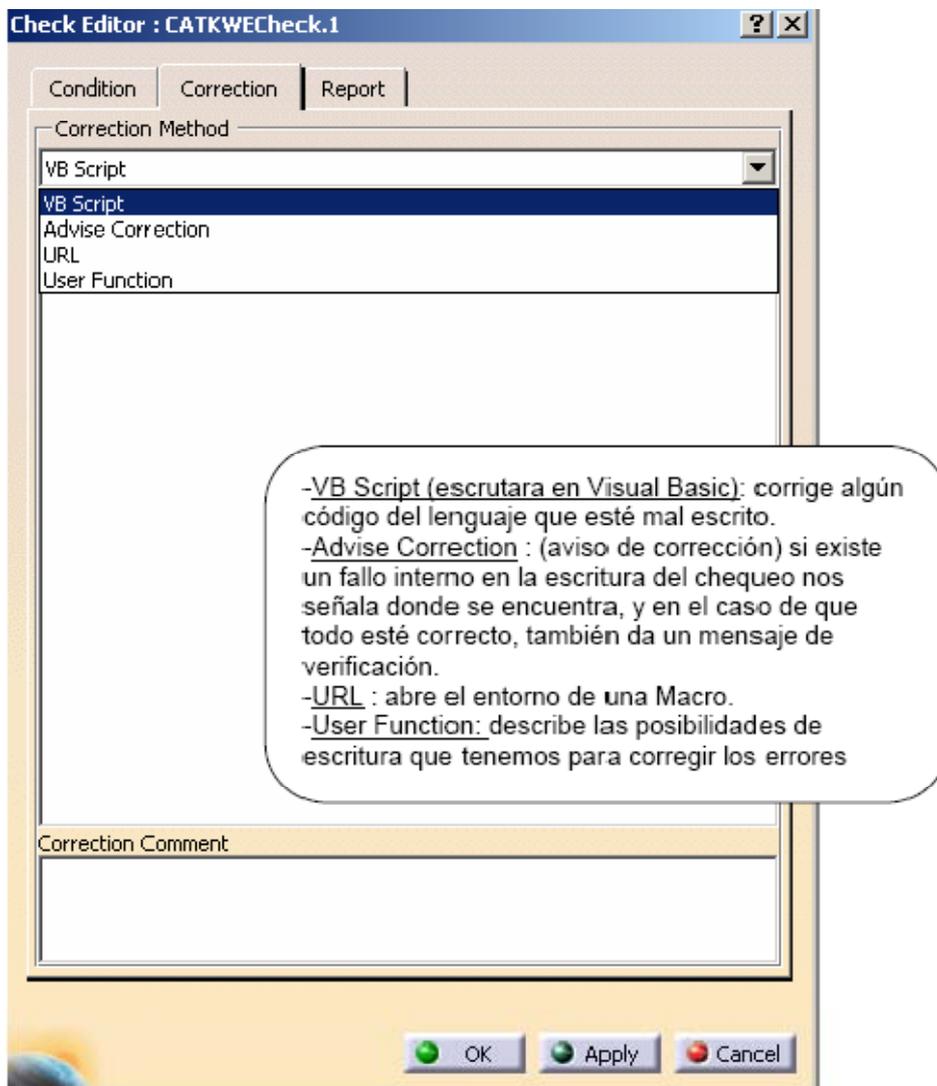


Figura 4.33. Corrección de Chequeo Experto

La herramienta de análisis global se utiliza para desarrollar y evaluar Chequeos Expertos que deben ser localizados en las especificaciones del árbol. Este análisis ayuda a comprender el estado de validación de las

- VB Script (escrutara en Visual Basic): corrige algún código del lenguaje que esté mal escrito.
- Advise Correction : (aviso de corrección) si existe un fallo interno en la escritura del chequeo nos señala donde se encuentra, y en el caso de que todo esté correcto, también da un mensaje de verificación.

- URL : abre el entorno de una Macro.
- User Function: describe las posibilidades de escritura que tenemos para corregir los errores, designaciones que contenga el chequeo y permite navegar por el interior del lenguaje de los chequeos y los componentes que han dado errores.

Las funciones que se pueden realizar son:

- Acceder a la información concerniente a los componentes que han dado errores
- Recoger información concerniente a los objetos y chequeos.
- Hacer automáticamente correcciones si son necesarias.

A la herramienta del análisis global se puede acceder pinchando en el icono de la barra de herramientas obteniendo un análisis en uno de los siguientes estados:



Todos los chequeos están actualizados y pueden ser aplicados.



El chequeo necesita ser actualizado.



Todos los chequeos están actualizados y al menos uno de ellos es incorrecto.

Un ejemplo de ventana de diálogo que se despliega es la siguiente:

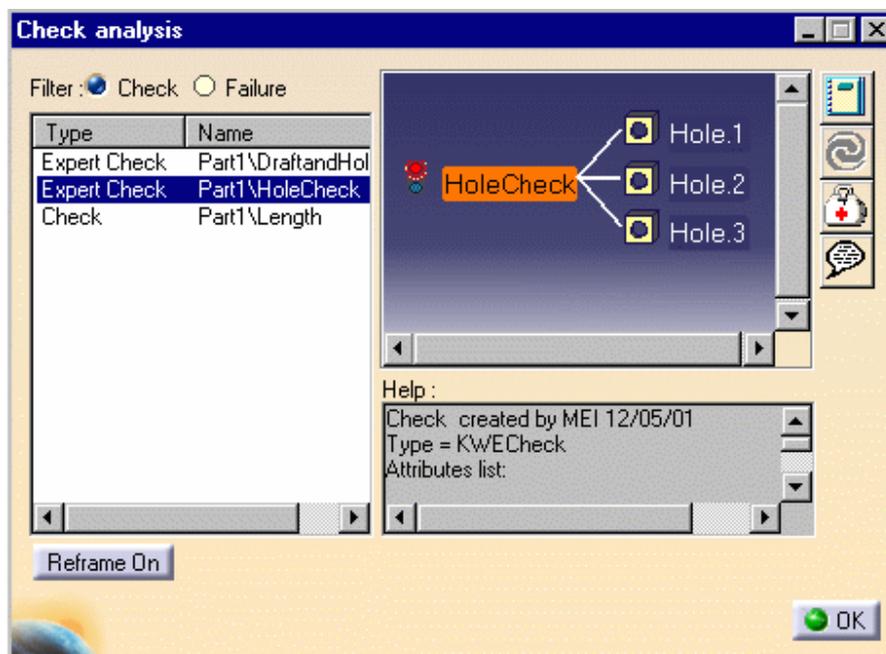


Figura 4.34. Análisis del chequeo

Filtrar una sesión: esta opción permite aplicar un filtro a un chequeo o a los componentes que han dado error para encontrar donde se encuentra y poder corregirlo.

Nos permite dos opciones de corrección:

- *Check* (chequeo): solo los chequeos que han dado error cuando se han actualizado son localizados y desactivados.
- *Failure* (fallo): todos los componentes que han dado error cuando han sido actualizados son desactivados.

4.4. PRODUCT KNOWLEDGE TEMPLATE

El Producto del Conocimiento, parte de la asociación de características técnicas de cualquier nivel de complejidad dentro de las Plantillas de Conocimiento de Producto que permite la captura de la metodología del plan de diseño definida interactivamente en Catia.

Los productos del Conocimiento no sólo pueden contener geometría, sino también cualquier parámetro asociado o relaciones, incluyendo inteligencia de reglas del plan, fórmulas del plan y chequeos, proporcionando la habilidad de encapsular las especificaciones de rasgos haciéndolos sofisticados.

La dirección de conocimiento capturado es más fácil gracias a la habilidad de revisar y mantener plantillas de manera interactiva, como modificar cualquier plan de Catia (no se requiere ninguna habilidad de programación, como en el caso de la utilización de Macros). Esto permite a las plantillas ser adaptadas a los requisitos cambiantes de la empresa, acomodar metodologías mejoradas o requisitos del cliente.

Pueden guardarse Plantillas de Conocimiento de Producto una vez definidas en Catálogos de Catia para permitir facilitar el acceso a los distintos usuarios de una empresa.

Se dispone esencialmente de tres fuentes de trabajo:

- **PowerCopies**
- **UserFeatures**
- **Documents Templates**

Si además de estas aplicaciones introducimos el idioma de escritura que proporciona las Plantillas de Producto de Conocimiento llamado **scripting** que es un idioma simple y declaratorio, el usuario puede describir – en formato de texto - las especificaciones geométricas, transformaciones y especificaciones que nuestro modelo necesite.

Es más para que todavía se pueda hacer la creación de escrituras más fácil pueden usarse Plantillas de Conocimiento simples, poderosas como cualquier objeto de la escritura.

Se recurre a esta forma de diseño cuando el problema no se pueda resolver con PowerCopies, UserFeatures, y Documentos Templates.

4.4.1. El entorno.



Crear un PowerCopy que es un elemento que contiene geometría que va a ser copiada en otro documento.



Crear una UserFeature que es un elemento que realiza la misma función que un PowerCopy pero la geometría no se visualiza en el árbol y no es editable.



Crear un Document Template que es una plantilla que guarda una copia de todo el documento en un Catálogo



Guarda los distintos elementos, PowerCopies, UserFeatures y Documents Template en Catálogos



Sirve para seleccionar alguna aplicación desde un documento que este en sesión en ese momento



Llama a un documento que se encuentre archivado



Contiene todos los catálogos que se han creado en forma de librerías



Sirve para crear alguna aplicación en el lenguaje *scripting*

PowerCopies, UserFeatures y Documents Templates

Estos tres elementos sirven para automatizar los procesos en el diseño y son en esencia los elementos que componen “el conocimiento” en Catia. Analizaremos a continuación las diferencias entre los tres y las ventajas e inconvenientes que tiene usar uno u otro.

4.4.2. PowerCopy

Un PowerCopy es una aplicación que permite al usuario de Catia crear utilidades que adaptan las geometrías a las necesidades del momento.

Su principal función es la de realizar tareas repetitivas que son tediosas y que llevaría realizarlas un tiempo apreciable, tiempo que se ahorraría una vez estuviera construido el PowerCopy.

Se utilizan fundamentalmente cuando tenemos una construcción geométrica que se va a repetir a lo largo del proceso de diseño, así, se construye una sola vez, se inserta dicha geometría en el PowerCopy, y puede ser utilizado en otro modelo tantas veces como sea necesario.

El PowerCopy solamente se puede crear e insertar en el interior de un CatPart, al igual que las UserFeatures que luego trataremos, por lo tanto los elementos que lo componen son principalmente geometrías (planos, superficies, etc...). Se van a posicionar dentro del modelo dependiendo de los datos de entrada.

La forma de construir un PowerCopy y una UserFeature es prácticamente la misma, la diferencia reside en que al incluir elementos en la UserFeature hay que tener mucho cuidado con el orden.

Todo lo que se incluye en un PowerCopy es editable cuando se inserte en el nuevo Catpart, por tanto no tiene la opción de editar u ocultar los elementos (*Outputs*).

4.4.2.1. Construcción de un PowerCopy

Si creamos un PowerCopy con el icono *Create a Power Copy*, , aparece la siguiente ventana:

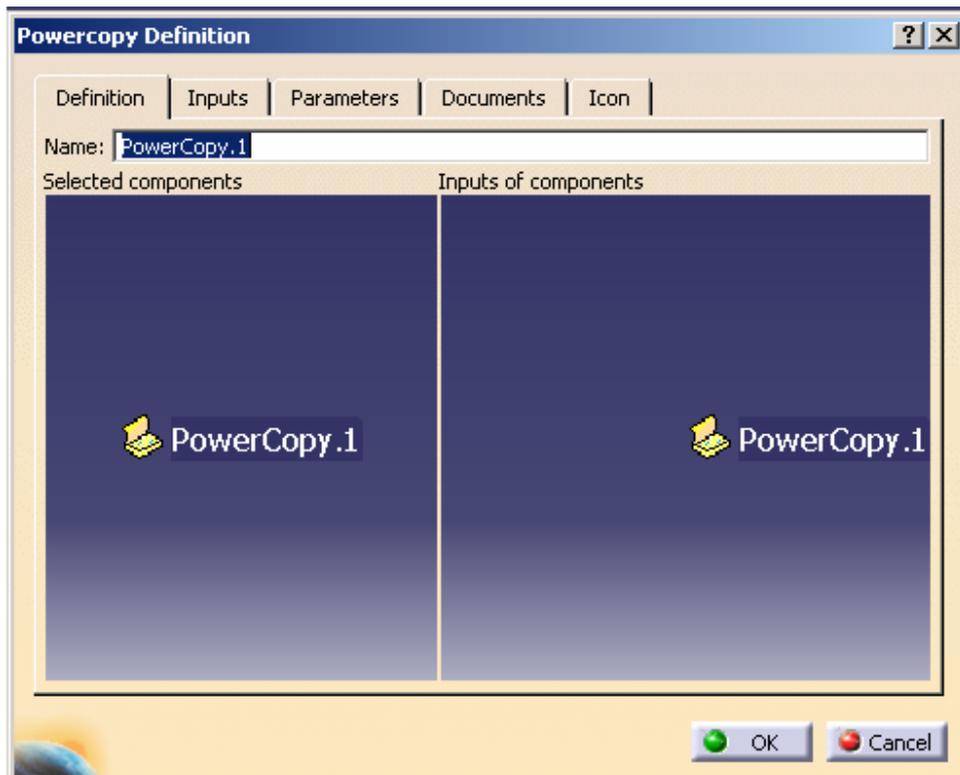


Figura 4.35. PowerCopy.

Al crear un PowerCopy se van seleccionando los elementos en el árbol que queramos generar de nuevo en otro documento y en la parte derecha de la ventana van apareciendo los padres de los elementos que vamos seleccionando y que serán las entradas o *Inputs* que luego tendremos que seleccionar para hacer la réplica del elemento.

Por ejemplo, si construimos una recta entre dos puntos que a su vez están apoyados sobre un plano, al seleccionar la recta como elemento que queremos duplicar, en la ventana de las entradas aparecen los dos puntos que hacen falta para la construcción:

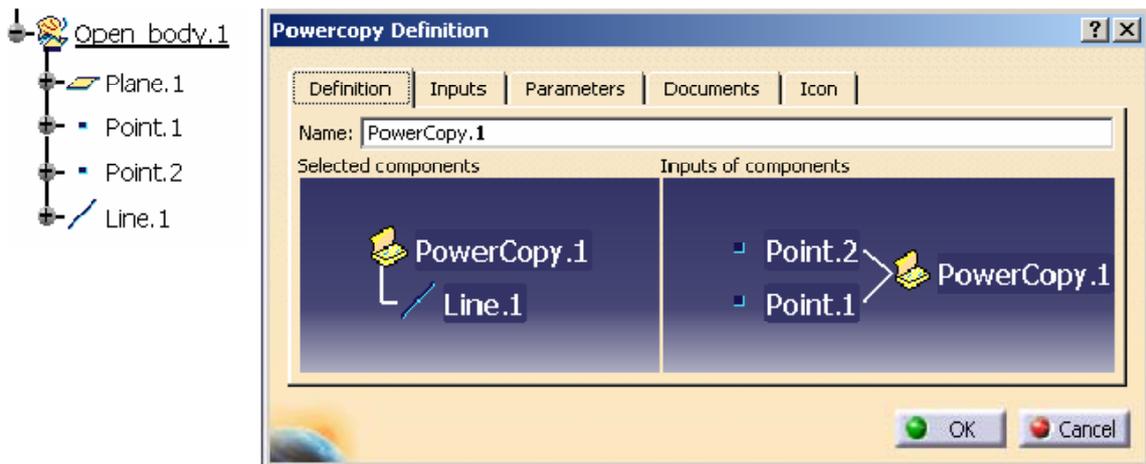


Figura 4.36. Parámetros a introducir en PowerCopy

Y si seleccionamos los puntos la entrada será el plano.

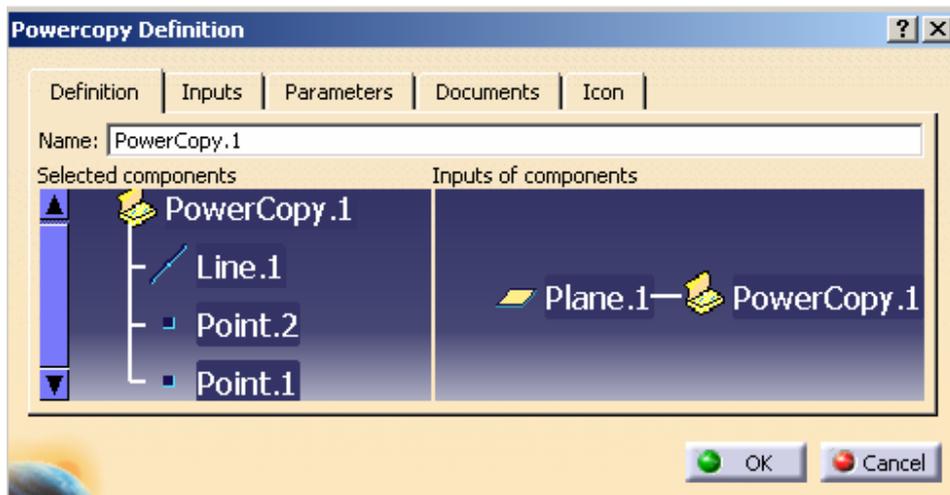


Figura 4.37. Definición en PowerCopy

Documents:

En este apartado aparecen la dirección de las posibles ligaduras externas (*link*) o tablas de diseño que son referenciadas por los elementos incluidos en el PowerCopy.

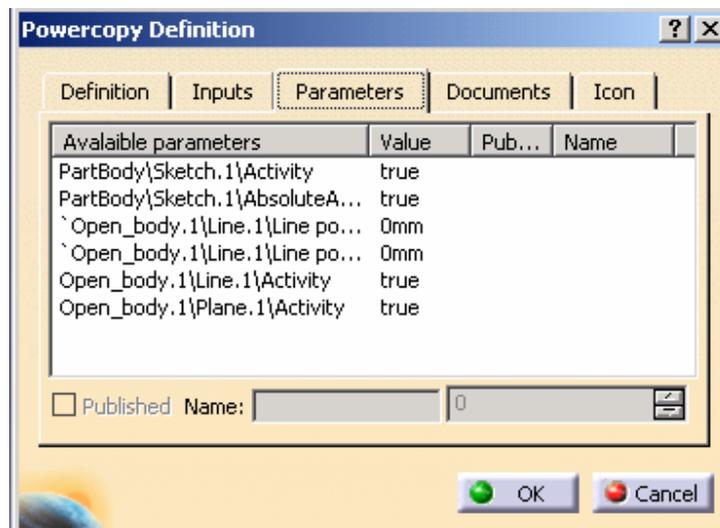


Figura 4.38. Parámetros en PowerCopy

Icon

En esta ventana se puede elegir un icono que identifique el Power-Copy en el árbol. Con el botón *Grab screen* podemos capturar una imagen del Power-Copy que se guardará con la definición.

Con el botón *remove preview* retiramos la imagen.

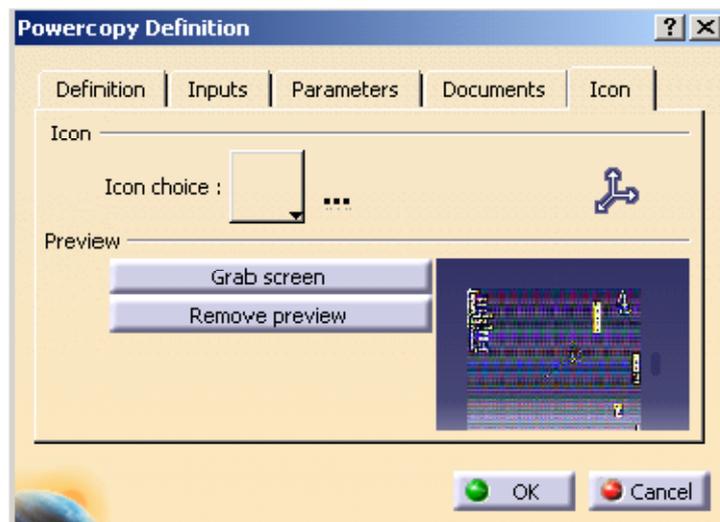


Figura 4.39. Iconos para una PowerCopy

4.4.2.2. Recomendaciones sobre el uso del PowerCopy

- Se pueden incluir más de un PowerCopy en cada modelo.
- Se incluirán como Input los Parámetros que vayan a ser constantes en los modelos donde se van a aplicar : espesores, radios, etc.
- Para incluir los elementos de referencia (geometría alámbrica) que van a formar parte del PowerCopy, se recomienda incluir dichos elementos en un único OpenBody, que será lo que se seleccione (también se incluirán los parámetros).
- Se pueden seleccionar elementos de referencia que se han copiado con *Link*, ligaduras ,de modo que ese Link se mantendrá en el modelo donde quede aplicado.
- En lo que respecta al PartBody, puede que todo forme parte del PowerCopy, o que solo se seleccionen parte de los elementos del mismo. En el caso de que sea todo, no se seleccionará el PartBody, sino elemento a elemento en el orden de creación, ya que de lo contrario nos crearía otro PartBody en el modelo en el cual lo aplicamos.
- El orden recomendable para seleccionar los distintos tipos de elementos que van a formar parte del PowerCopy es el siguiente:
 - Relaciones y reglas
 - Elementos
 - OpenBodies (elementos de referencia, parámetros)
- Una vez seleccionados todos los elementos, se comprobará que los Inputs que aparecen son los que realmente vamos a necesitar.
- Por último publicaremos únicamente los parámetros que pueden variar en función de donde vamos a aplicarlo.
- Cuando se incluyan Sketcher se tendrá en cuenta que:
 - Es preferible no usar proyecciones ni intersecciones.
 - El Sketcher estará completamente definido (en color verde).
 - No crear restricciones con respecto a los ejes H V y a planos de referencia.

4.4.3. UserFeature.

La UserFeature tiene la misma estructura que el PowerCopy y su función también es la misma, lo que ocurre es que la estructura interior está oculta y no se tiene acceso a la misma si no estamos dentro del Catpart donde se ha creado, así se puede decir que es como una caja negra donde sólo se ven las entradas (Inputs) y las salidas (Outputs)

Las entradas serán los elementos en los que nos hemos basado para la construcción de la geometría, por lo que hay que tener mucho cuidado con la reacción padres hijos que tienen todos los elementos ya que es habitual cuando se crean este tipo de aplicaciones que aparezcan Ciclos porque algún elemento no tiene bien definidos cuáles son sus padres y sus hijos.

4.4.3.1. Construcción de un UserFeature

Se ha explicado con anterioridad en qué consiste una UserFeature, cuando se debe utilizar y cuáles son los apartados que la forman. La conclusión más importante que hay que sacar de todo esto es el orden que hay que seguir en la construcción del modelo

que se quiere usar en la UserFeature  y como ir insertando todos los elementos que queremos utilizar posteriormente en la aplicación, así como también hay que tener en cuenta lo que será editable y lo que quedará oculto.

Con respecto al Power-Copy, existe una diferencia en la ventana correspondiente a **Documents** y aparecen dos nuevas solapas: **Outputs** y **Type**.

Definition

Cambiamos el nombre a la UserFeature para que posteriormente sea más fácil de reconocer, la llamamos Geometría.

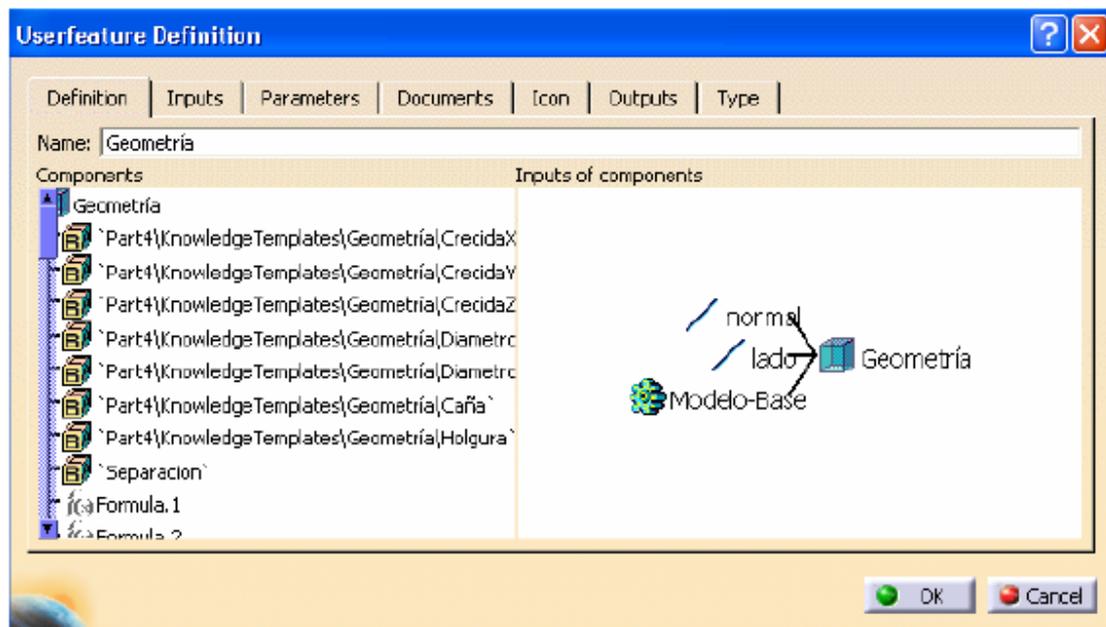


Figura 4.40. Definición en UserFeature

Inputs

A un lado en la mitad izquierda se incluye todo lo que se quiere repetir en el proceso posterior y se va seleccionando en el árbol, en el lado derecho van apareciendo los padres de los elementos que vamos seleccionando, si el árbol está bien construido, y por eso es tan importante el orden de construcción, al final en el lado derecho nos quedarán los elementos iniciales de los que partíamos en un principio para construir el modelo. Estos son nuestros Inputs.

Parameters

Una vez que tenemos todos los elementos incluidos en la UserFeature vamos a editar los parámetros que hemos definido anteriormente.

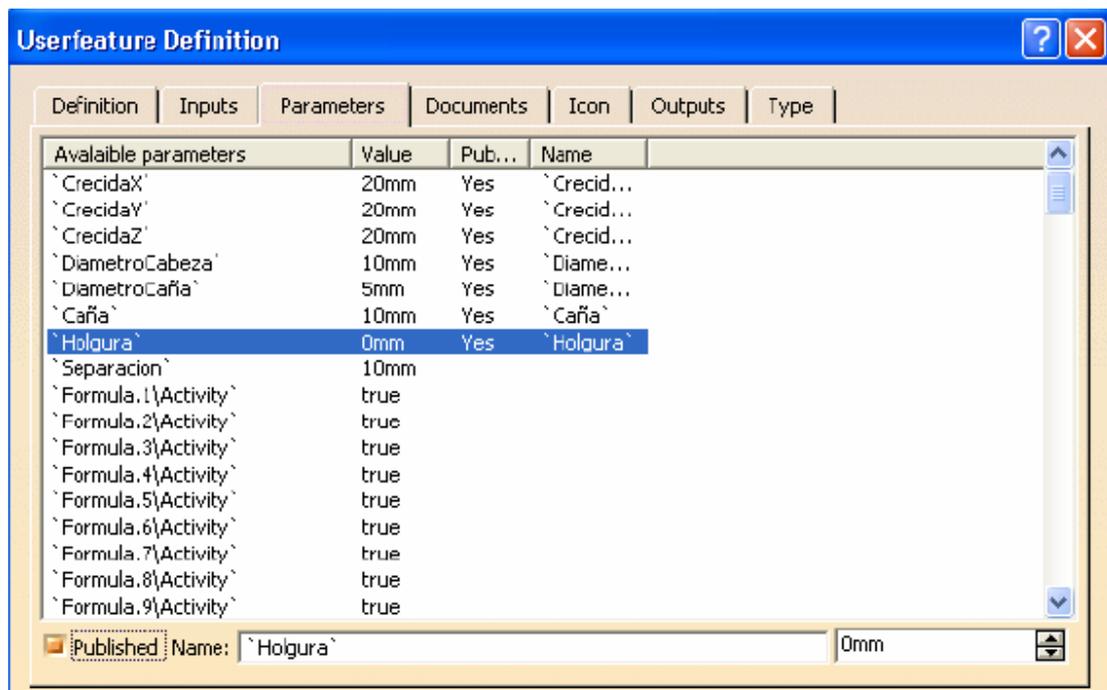


Figura 4.41. Parámetros en UserFeature

Documents

Solo se incluyen tablas de diseño, no pudiendose incluirse elementos copiados con link.

Outputs

En esta ventana se define el resultado que queremos trasladar al modelo en el cual lo queremos aplicar. Al resultado principal (Main result) se le podrán añadir (Add) otros elementos secundarios, los cuales se podrán renombrar. Tenemos las opciones de retirar (Remove) o reemplazar (Replace) cualquier elemento.

Icon

Por último seleccionamos un icono que nos facilite la localización de la UserFeature al igual que hacíamos antes con el nombre.

4.4.3.2. Inserción de una UserFeature

Cuando se crea una UserFeature se guarda en el documento donde ha sido creada. Para aplicarla de nuevo tenemos en el menú de herramientas Insert→ Instantiate From Document → y se abre el documento donde se ha creado la UserFeature. Es importante que ese documento no esté abierto en la sesión, porque entonces no permitiría abrir el documento.

Aparece una ventana donde nos pide las entradas que son necesarias para la creación del elemento que contiene la UserFeature
En el caso de la recta nos pide dos puntos. Al seleccionar dos puntos cualquiera del documento los unirá con una recta.

4.4.4. Diferencias entre los PowerCopies y las UserFeatures.

Ambos elementos se crean y se insertan de la misma forma, pero tienen diferencias entre sí:

- En una UserFeature el orden en que se insertan los elementos es muy importante para que funcione bien, de manera que se va seleccionando en el árbol los hijos que no tienen ningún elemento que pertenezca a él, y sus sucesivos padres, de manera que se van encadenando unos con otros hasta que al final nos quedan como Inputs los elementos que no tienen padre y en los que está basada toda la construcción. En un PowerCopy no importa tanto esta relación padre hijo aunque es conveniente seguir un orden.
- En un PowerCopy se pueden introducir Bodies sin que estos estén operados, en cambio para una UserFeature sólo se pueden introducir mediante operaciones Boleanas.
- Un PowerCopy puede ser modificado una vez creado, y la UserFeature una vez creada no deja cambiar ningún elemento, sino que hay que construirla de nuevo.
- El PowerCopy al insertarse en un modelo CatPart aparece un árbol con una geometría exactamente igual a la creada en el modelo inicial, sin embargo en una UserFeature sólo aparecerán los elementos que fueron editados al crearla, quedando totalmente oculto para el usuario como se ha creado la UserFeature.

4.5. DOCUMENTS TEMPLATES

Este tipo de documento se diferencia de la UserFeature y el PowerCopy fundamentalmente en cuatro cosas:

- Cuando se inserta en un nuevo documento éste tiene que ser un Product no puede insertarse en un Part
- Tiene que llevar asociado un catálogo
- Hace una copia en el Product del modelo donde se ha construido el Template respetando la misma estructura del árbol inicial, y además inserta junto con esta copia un nuevo Body como resultado de una copia de la pieza en forma de sólido cerrado.
- Sólo hace falta para su construcción decirle cuáles queremos que sean los parámetros de entrada o Inputs.

4.5.1. Construcción de un Document Template

Para crear un Documento Template lo primero que hay que hacer es salvar el Part donde lo queremos construir, porque el Documento Template es una copia de todos los elementos que tenemos en el Part, y si éste todavía no está guardado la construcción no es operativa.

Se abre la ventana pulsando en el icono *Create a Document Template*, .

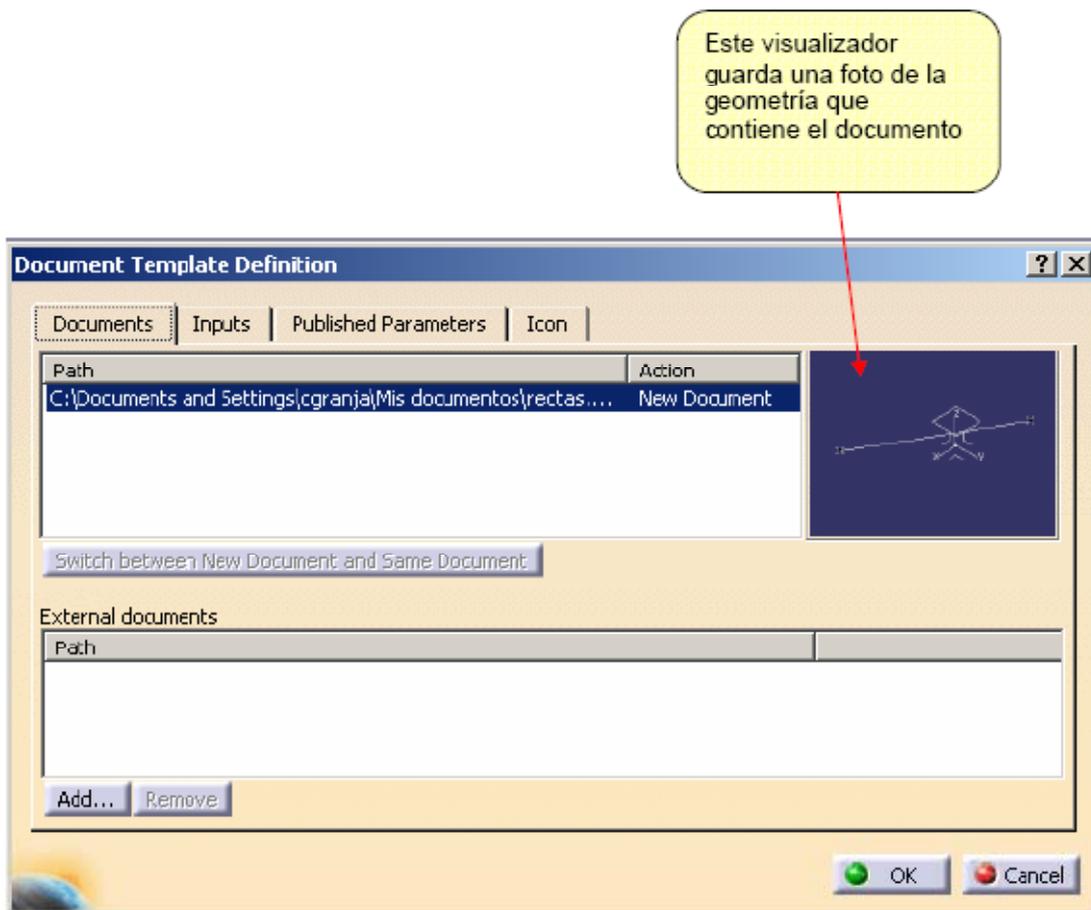


Figura 4.42. Definición de un Document Template

Documents

Hace referencia al documento donde está contenido el Documents Template y la ubicación del archivo donde está guardada.

También se pueden insertar documentos externos.

Inputs

Al igual que en los PowerCopies y las UserFeatures son los elementos de entrada que luego tendremos que seleccionar par aplicar el Document Template.

Published Parameters

Se pueden publicar los parámetros que se quieran dándoles valores, además si editamos la lista se despliega un menú con todos los parámetros que contiene el documento.

Este visualizador guarda una foto de la geometría que contiene el documento
Aquí aparecen todos los parámetros que son publicados.

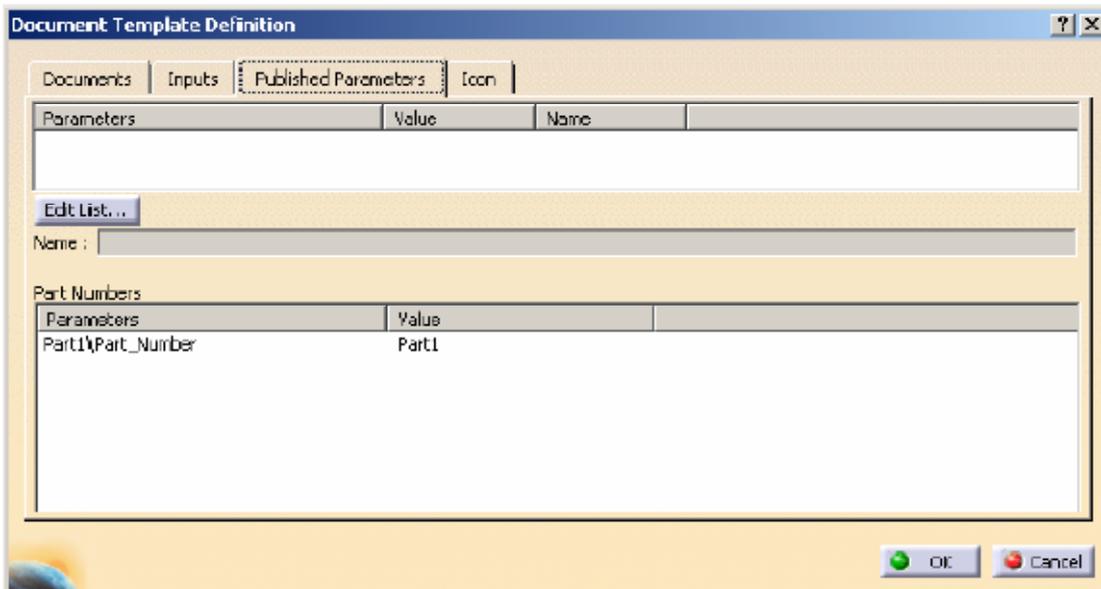


Figura 4.43. Parámetros publicados en Document Template

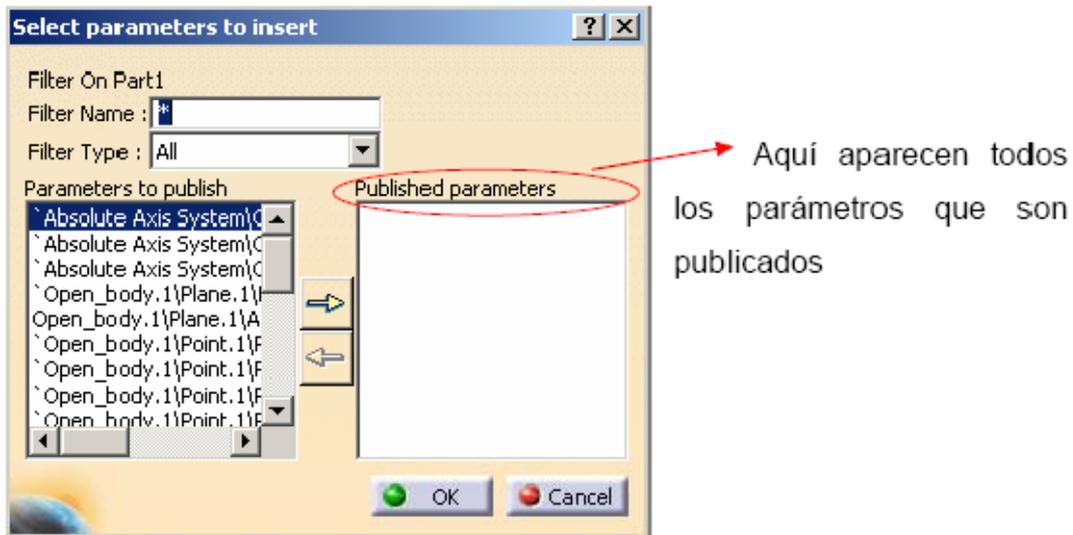


Figura 4.44. Selección de parámetros en Document Template

Icon

Elegimos un icono que haga referencia a la geometría que contenga el Document Template para hacer más fácil su ubicación.

Una vez configurado el Documento Template hay que salvar otra vez el CatPart.

4.5.2. Insertación de un Document Template

La única manera de insertar un Document Template es crear un Catálogo, porque este tipo de documentos deben ser insertados en un Conjunto ya que al guardar una copia de todo el contenido del Part de origen tiene identidad de Part y en Catia no se puede insertar un Part en otro Part ya que existe la estructura de Conjunto con un Product.

4.5.3. Creación de Catálogos

Para crear un catálogo nos vamos a un nuevo módulo de Catia que es el Infrastructure en el submódulo Catalog Editor.

Un catálogo es una estructura de organización que nos permite tener almacenado por familias las distintas aplicaciones que se van creando el Catia, así se consigue tener una librería de Conocimiento almacenado y ordenado, para su posterior reutilización.

Al organizarse por archivos que contienen familias, hay que ir creando dichas familias, que a su vez se organizan por componentes.

Para crear una familia se tienen el icono *Add families* 

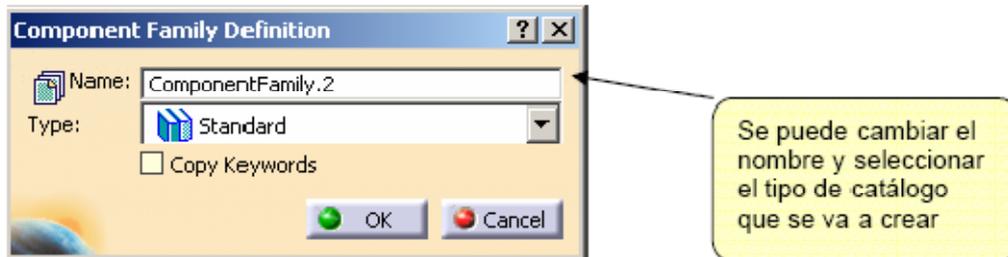


Figura 4.45. Definición de familias de catálogos

Una vez creada la familia, hay que crear un componente, para ello se selecciona en el árbol la familia que acabamos de crear, y con el icono *Add components*, , nos queda la siguiente ventana de diálogo:

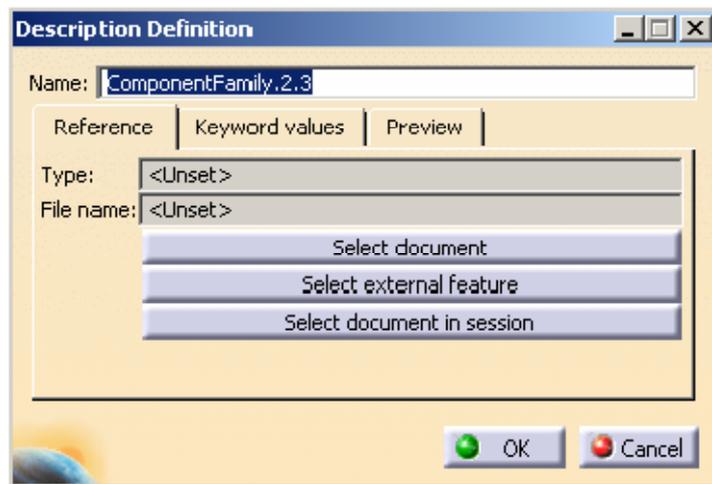


Figura 4.46. Adición de componentes.

Para crear un componente se pueden elegir tres caminos:

1. Seleccionar un documento desde un archivo
2. Seleccionar una aplicación que esté creada en otro documento y dentro de estas aplicaciones se contemplan:
 - PowerCopies
 - UserFeatures
 - Document Template
3. Seleccionar un documento que se encuentre abierto en el momento de crear el Catálogo.

En el árbol van apareciendo las distintas familias que se van creando y la referencia del documento donde están creados los componentes.

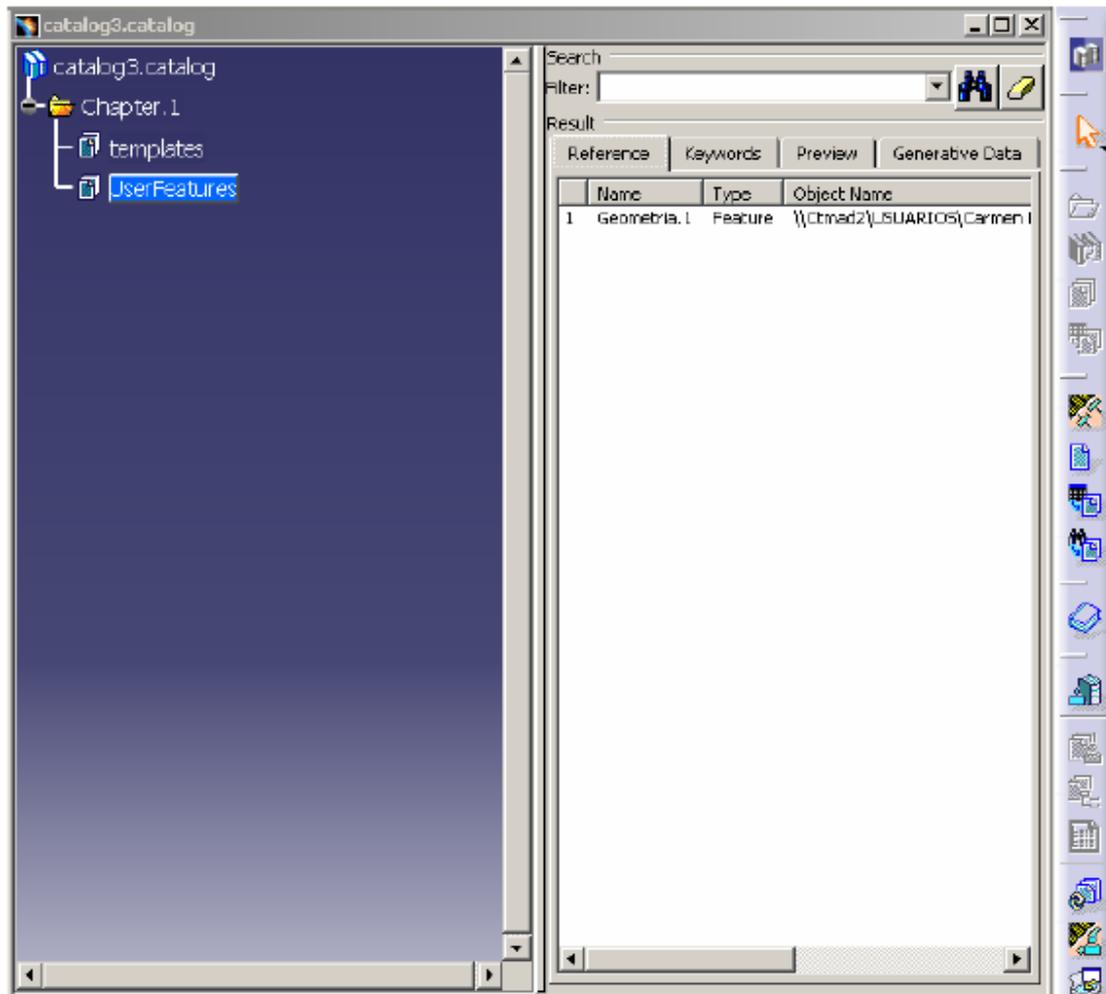


Figura 4.47. Referencias de documentos creados

Reference

Aparece una referencia de el camino donde se encuentra guardado el objeto que estamos incluyendo en el catálogo.

Keywords

Son palabras claves que sirven para reconocer alguna característica del objeto incluido en el Calálogo.

Para crear una Keywords tenemos el icono Add Keywords, .

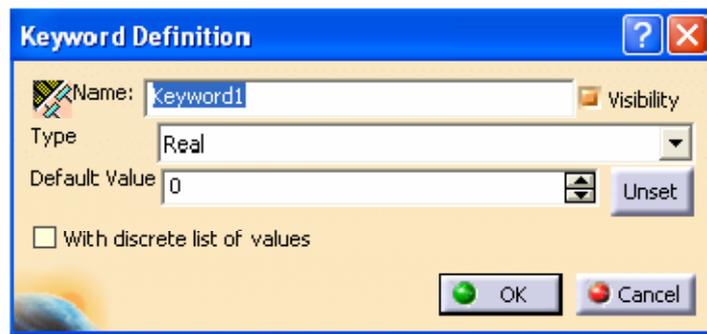


Figura 4.48. Claves para el reconocimiento de objetos de catálogos

Se puede dar el nombre que se quiera a la Keyword y también se puede elegir el tipo y asignarle un valor.

Preview

Muestra una visualización de los elementos que contiene el Catálogo.

Generative Data

Se tiene información de los documentos que contiene el catálogo, como la fecha de creación, el archivo donde se encuentran, a que familia pertenecen los componentes, y si pertenecen a otro catálogo.

También se pueden crear catálogos con documentos que tengan incluida una tabla de diseño con el icono *Add Part Family*, .

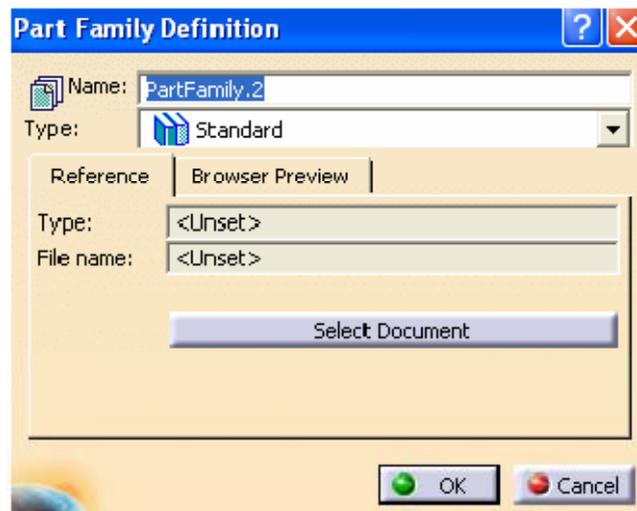


Figura 4.49. Adición de familias.

Al seleccionar este tipo de documento se generarán en el catálogo la familia entera de elementos con los distintos valores de los datos que estén incluidos en la tabla de diseño.

Por último los catálogos permiten crear ligaduras entre ellos con el icono *Add link to other catalog*, , con lo cual, se puede tener un solo catálogo raíz del que cuelguen todos los demás agrupados por familias, y tener totalmente organizadas las distintas aplicaciones que se vayan realizando en un proyecto o en una empresa.

4.6. CONCLUSIONES

Catia v5 es un programa que se encuentra modulado para las distintas fases del ciclo de vida del producto. Contiene módulos de modelado sólido junto a otros de generación de superficies que son bastante completos; por el contrario el módulo de elementos finitos es bastante rudimentario, aunque es muy rápido para un cálculo sencillo de tensiones y desplazamientos en una pieza. También posee un modulo de fabricación capaz de generar el programa de control numérico para la máquina herramienta donde se vaya a mecanizar el producto.

En cuanto al almacenamiento del conocimiento el modulo Knowledgware se divide en tres submódulos:

- Knowledge advisor: en el que el usuario puede ordenar y aplicar el conocimiento por medio de reglas, tablas, fórmulas, chequeos, macros con argumentos, ... para automatizar el diseño del producto.
- Knowledge expert: es parecido al advisor y utiliza las mismas herramientas para tratar el conocimiento pero con el matiz de que los usuarios a los que van destinado el producto no pueden modificarlo, solo añadir conocimiento al que previamente se había creado con esto se consigue que no se modifique la información.
- Product Knowledge Template: se pueden crear productos cambiantes, con diferentes matices, dependiendo de los parámetros introducidos por el usuario, pudiéndose catalogar la información para formar familias de productos.

Catia permite diferentes formas de tratar el conocimiento, analizando los pros y los contras de cada una, podemos elegir la que mejor se adapte a las exigencias de nuestro producto.

