

# Escandallos

## WhitePaper

Abril de 2016

# Contenidos

1. Propósito	3
2. Elementos Básicos	4
3. Variables	5
4. Parámetros	6
5. Instrucciones	7
5.1. Instrucciones de definición de variables	7
5.2. Instrucciones Booleanas	7
5.2.1. Instrucciones de Color	7
5.3. Instrucciones de Tipo de Cálculo	8
5.4. Instrucción de llamada a otro Escandallo	12
5.5. Instrucciones para la presentación de Mensajes	13
5.6. Instrucción de Control de Flujo	13
5.7. Instrucciones de Asignación	15
5.8. Instrucciones de Función	17
5.9. Instrucciones de Tablas	18
5.10. Instrucciones de Mano de Obra	19
5.11. Instrucciones para indicar la superficie	19
5.12. Instrucciones para la generación de secciones	19
5.13. Instrucciones para la Obtención de Datos	19
5.14. Comentarios	19
6. Historial de escandallos	20

# 1. Propósito

Un Escandallo es una lista de instrucciones fácil de programar mediante la cual es posible generar materiales, disparar opciones, modificar cotas, etc... Este documento está destinado aquellas personas que requieran programar por medio de esta herramienta un comportamiento específico.

Para ellos es necesario tener conocimientos previos de PrefGest, así como de PrefCAD.

## 2. Elementos Básicos

Los conceptos básicos para comprender el funcionamiento de un escandallo son los siguientes:

- PROGRAMA = Lista de Instrucciones;
- LISTA DE INSTRUCCIONES = Instrucción; [Instrucción];
- INSTRUCCIÓN = Instrucción(Lista de Parámetros);
- LISTA DE PARÁMETROS = Parámetro, [Parámetro].

En definitiva, un Programa está formado por una Lista de Instrucciones separadas por “;” cada una de ellas con un nombre y uno o más Parámetros., separados por comas.

El proceso a seguir a la hora de trabajar con escandallos es el siguiente:

1. En *PrefGest*, programar los escandallos.
2. En *PrefCAD*, asociar los escandallos al diseño.

## 3. Variables

Las Variables empleadas en las Instrucciones pueden ser de 3 tipos:

- **Variables de modelo:** Estas variables muestran información acerca del diseño. Especifican datos muy diversos, como las dimensiones del modelo: L, A (y el resto de cotas del modelo) o aspectos específicos de este, como por ejemplo, la variable Np específica para los escandallos asociados por vínculo de persiana al modelo (en concreto, Np detalla el número de paño en el que se encuentra).
- **Variables de opciones:** Estas variables representan los valores de las opciones.
- **Variables de usuario:** Estas variables son definidas por el usuario en el propio escandallo. Se crearán con la función “SEA” (La cual explicaremos más adelante).

## 4. Parámetros

Los Parámetros empleados en las Instrucciones pueden ser de 2 tipos:

- Cadenas de caracteres entre “” : Estas cadenas de caracteres pueden ser una variable de medidas, una opción o una referencia de material.
- Expresiones entre [ ] : Estas expresiones pueden ser un valor numérico, una variable u una fórmula matemática sencilla (una suma, una resta, una multiplicación o una división) utilizando a su vez números o variables. Por ejemplo: [2], [L], [L\*A] serían fórmulas válidas.

## 5. Instrucciones

### 5.1. Instrucciones de definición de variables

Estas Instrucciones permiten al escandallo conocer las medidas del diseño y las opciones con las que va a trabajar.

- **PREGUNTAVALOR / ASKFORVALUE** (“*Nombre\_Variable*”): Esta instrucción muestra un cuadro de diálogo en el cual se deberá introducir el valor de la variable *Nombre\_Variable*. El parámetro *Nombre\_Variable* tomará una variable de usuario.
- **OPCIONES / OPTIONS** (“*N\_Opción1*”, “*N\_Opción2*”, ..., “*N\_Opcióni*”): Esta instrucción permite especificar el conjunto de opciones que serán consideradas en el escandallo. Los parámetros *N\_Opción1*, *N\_Opción2*, ..., *N\_Opcióni* deben ser nombres de opciones dadas de alta en la base de datos.
- **PREGUNTAOPCION / ASKFOROPTION** (“*Nombre\_Opción*”): Esta instrucción muestra un cuadro de diálogo con el nombre de la opción *Nombre\_Opción* y la lista de todos sus valores posibles, de entre los cuales deberá escoger uno.
- **PREGUNTAOPCION / ASKFOROPTION** (“*Nombre\_Opción*”, “*Valor1*”, “*Valor2*”, ...): Esta instrucción muestra un cuadro de diálogo con el nombre de la opción *Nombre\_Opción* y el subconjunto *Valor1*, *Valor2*, ... de sus valores posibles, de entre los cuales deberá escoger uno. Si alguno de los valores listados no es válido, será ignorado.

Las instrucción PREGUNTAOPCION se empleará únicamente en los escandillos que vayan a ser presupuestados, siendo útil para que dependiendo del valor de una opción se muestre un subconjunto de valores de otra.

- **SEA / LET** *Nombre\_Variable* = [*Expresión*]: Esta instrucción asigna a la variable *Nombre\_Variable* el valor del Parámetro *Expresión*. El Parámetro *Expresión* puede ser un número, una variable, una función o una fórmula.

### 5.2. Instrucciones Booleanas

- **EXISTEVARIABLE / EXISTSVARIABLE** (“*Variable*”): Esta instrucción devolverá verdadero (“1”) o falso (“0”) si la variable “*Variable*” está siendo enviada al escandallo que la está ejecutando.
- **OPCION** (“*Nombre\_opción*”, “*Valor\_opción*”): Esta instrucción comprueba si la opción “*Nombre\_opción*” está tomando el valor “*Valor\_opcion*”. En caso de que sea así, devolverá un “1” (verdadero), o si la opción tiene otro valor asignado, devolverá “0” (falso).

#### 5.2.1. Instrucciones de Color

- **COLORINTERIOR / INNERCOLOUR** (“*Nombre\_color*”): Esta instrucción es de tipo booleano. Devuelve verdadero (“1”) si el color *Nombre\_color* es el color interior del color que se le ha asociado al escandallo.
- **COLOREXTERIOR / OUTERCOLOR** (“*Nombre\_color*”): Esta instrucción devuelve “1” si el color especificado en *Nombre\_color* es el color exterior del color que se le ha asociado al escandallo.

- **COLOR / COLOUR** (“Nombre\_color”): Esta instrucción devuelve verdadero si Nombre\_color es el color que se le ha asociado el escandallo. Esta instrucción es útil para ejecutar determinadas Instrucciones dependiendo del color del elemento del modelo.

El color que se le asocie al escandallo dependerá del tipo de escandallo, por ejemplo, en un vínculo común es el color del elemento al que está asociado el vínculo, en el escandallo de cajón es el color del cajón de persiana, etc.

### 5.3. Instrucciones de Tipo de Cálculo

Estas Instrucciones permiten generar materiales. Por cada referencia diferente de material será necesaria una nueva instrucción.

- **BARRA / ROD** (“Ref”, [UD],[Long],[Ang1],[Ang2]): Esta instrucción genera de la referencia Ref, una cantidad UD de tramos de longitud Long y ángulos de corte inicial y final Ang1 y Ang2. Ref deberá ser una referencia con color (referencia final) de un material de Tipo de Cálculo = Barras. De tratarse de una referencia base, se generará la primera referencia con color dada de alta en la aplicación Materiales de PrefWise.
- **BARRAC / CROD** (“Ref”, [UD],[Long],[Ang1],[Ang2]): Esta instrucción es similar a la anterior, salvo que Ref deberá ser una referencia base. El color de la referencia lo dictará el modelo.
- **BARRACINT / CINNROD** (“Ref”, [UD],[Long],[Ang1],[Ang2]): Esta instrucción es similar a la anterior. El campo Ref también deberá ser una referencia base, pero en este caso en programa intentará generar el perfil con el color interior especificado en el modelo.
- **BARRACEXT / COUTROD** (“Ref”, [UD],[Long],[Ang1],[Ang2]): Esta instrucción tiene la misma funcionalidad que la anterior, pero en este caso el programa intentará generar el perfil con el color exterior del diseño.
- **BARRA\_EX / ROD\_EX** (“Ref”, [UD],[Long],[Angulo Inicial],[Angulo Final],[Angulo],[Angulo Z Inicial],[Angulo Z Final], “Param”): Esta instrucción genera de la referencia final Ref, una cantidad UD de tramos de longitud Long y ángulos de corte inicial y final Angulo Inicial y Angulo Final. A diferencia de las instrucciones anteriores, especificaremos en esta sentencia el tramo del modelo en el que va ubicado el material de tipo barra en campo Angulo, así como los cortes en el ángulo Z, indicados en los parámetros Angulo Z Inicial y Angulo Z Final, y el parámetro “Param” el cual se utilizará para almacenar un parámetro del usuario, que posteriormente será utilizado para programar la PrefUserVB Project.
- **BARRAC\_EX / CROD\_EX** (“Ref”, [UD],[Long],[Angulo Inicial],[Angulo Final],[Angulo],[Angulo Z Inicial],[Angulo Z Final], “Param”): Esta instrucción es similar a la anterior, salvo que Ref deberá ser una referencia base. El color de la referencia lo dictará el modelo.
- **BARRACINT\_EX / CINNROD\_EX** (“Ref”, [UD], [Long], [Angulo Inicial], [Angulo Final], [Angulo], [Angulo Z Inicial], [Angulo Z Final], “Param”): Esta instrucción es similar a la anterior. El campo Ref también deberá ser una referencia base, pero en este caso en programa intentará generar el perfil con el color interior especificado en el modelo.
- **BARRACEXT\_EX / COUTROD\_EX** (“Ref”, [UD], [Long], [Angulo Inicial], [Angulo Final], [Angulo], [Angulo Z Inicial], [Angulo Z Final], “Param”): Esta instrucción tiene la misma funcionalidad que la anterior, pero en este caso el programa intentará generar el perfil con el color exterior del diseño.



- **BARRA\_EX\_U / ROD\_EX\_U** (“Ref”, [UD], [Long], [Angulo Inicial], [Angulo Final], [Angulo], [Angulo Z Inicial], [Angulo Z Final], [Unmounted]): Esta instrucción se usará con materiales de tipo barra. Genera de la referencia final *Ref*, una cantidad *UD* de tramos de longitud *Long* y ángulos de corte inicial y final *Angulo Inicial* y *Angulo Final*. También especificaremos en esta sentencia el tramo del modelo en el que va ubicado el material de tipo barra en campo *Angulo*, así como los cortes en el ángulo Z, indicados en los parámetros *Angulo Z Inicial* y *Angulo Z Final*, el parámetro “*Param*” (el cual se utilizará para almacenar un parámetro del usuario, que posteriormente será utilizado para programar la PrefUserVB Project) y en el parámetro *Unmounted* indicaremos como valor “1” cuando el material vaya desmontado en el modelo, y “0” cuando vaya montado.
- **BARRAC\_EX\_U/CROD\_EX\_U** (“Ref”, [UD], [Long], [Angulo Inicial], [Angulo Final], [Angulo], [Angulo Z Inicial], [Angulo Z Final], [Unmounted]): Esta instrucción es similar a la anterior, salvo que *Ref* deberá ser una referencia base. El color de la referencia lo dictará el modelo.
- **METRO / METER** (“Ref”, [UD],[Long]): Esta instrucción genera de la referencia *Ref*, una cantidad *UD* de tramos de longitud *Long*. *Ref* deberá ser una referencia con color (referencia) de un material de *Tipo de Cálculo = Metros*. De tratarse de una referencia base, se generará la primera referencia con color dada de alta en la aplicación *Materiales de PrefWise*.
- **METROC / CMETER** (“Ref”, [UD],[Long]): Esta instrucción genera de la referencia base *Ref*, una cantidad *UD* de tramos de longitud *Long*. Al tratarse de una referencia base, se generará la primera referencia con color dada de alta en la aplicación *Materiales de PrefWise*.
- **METROCINT / CINNMETER** (“Ref”, [UD],[Long]): Esta instrucción es similar a la anterior. Únicamente se diferencia en que el programa intentará generar el material de tipo metro con el mismo color especificado a la parte interior del modelo.
- **METROCEXT / COUTMETER** (“Ref”, [UD],[Long]): Esta instrucción, de la misma forma que la instrucción anterior, intentará generar el material de tipo metro con el color especificado a la parte exterior del diseño.
- **METRO\_EX / METER\_EX** (“Ref”, [UD],[Long], “Param”): Esta instrucción genera de la referencia *Ref*, una cantidad *UD* de tramos de longitud *Long*. *Ref* deberá ser una referencia con color (referencia) de un material de *Tipo de Cálculo = Metros*. De tratarse de una referencia base, se generará la primera referencia con color dada de alta en la aplicación *Materiales de PrefWise*. A diferencia de la instrucción “*METRO*” ó “*METER*”, disponemos del campo “*Param*”. En el podemos indicar un parámetro a rellenar por el usuario, para posteriormente utilizarlo en la programación de la PrefUserVB a conveniencia. Este parámetro es el único uso que tiene.
- **METROC\_EX / CMETER\_EX** (“Ref”, [UD],[Long], “Param”): Esta instrucción es igual que la anterior salvo por el dato de que el campo “*Ref*” contendrá una referencia base. El programa intentará generar el material de tipo superficie con el color asociado al modelo. En caso de que no este asociado, el programa generará la primera referencia con color dada de alta en la aplicación *Materiales de PrefWise*.
- **METROCINT\_EX / CINNMETER\_EX** (“Ref”, [UD],[Long], “Param”): Esta instrucción es similar a la anterior, salvo por el hecho de que intentará generar el material de tipo metro con el color interior especificado en el diseño.
- **METROCEXT\_EX / COUTMETER\_EX** (“Ref”, [UD],[Long], “Param”): De la misma forma que en la instrucción anterior, el programa intentará generar el material de tipo metro del mismo color que el especificado en la parte exterior del modelo.

- **METRO\_EX\_U / METER\_EX\_U** (“Ref”, [UD], [Long], “Param”, [Unmounted]): Esta instrucción genera de la referencia *Ref*, una cantidad *UD* de tramos de longitud *Long*. *Ref* deberá ser una referencia con color (referencia) de un material de *Tipo de Cálculo = Metros*. En el campo “Param” podemos indicar un parámetro a rellenar por el usuario, para posteriormente utilizarlo en la programación de la PrefUserVB a conveniencia. En el parámetro *Unmounted* indicaremos como valor “1” cuando el material vaya desmontado en el modelo, y “0” cuando vaya montado.
- **METROC\_EX\_U / CMETER\_EX\_U** (“Ref”, [UD], [Long], “Param”, [Unmounted]): Esta instrucción es similar a la anterior, salvo que *Ref* deberá ser una referencia base. El color de la referencia lo dictará el modelo. En caso de que no esté asociado, el programa generará la primera referencia con color dada de alta en la aplicación Materiales de PrefWise.
- **PIEZA / PIECE** (“Ref”,[UD]): Esta instrucción genera de la referencia *Ref*, una cantidad *UD* de piezas. *Ref* deberá ser una referencia con color (referencia) de un material de *Tipo de Cálculo = Piezas*. De tratarse de una referencia base, se generará la primera referencia con color dada de alta en la aplicación *Materiales* de *PrefWise*.
- **PIEZAC / CPIECE** (“Ref”,[UD]): Esta instrucción es similar a la anterior, salvo que *Ref* deberá ser una referencia base. El color de la referencia lo dictará el modelo.
- **PIEZACINT / CINNPIECE** (“Ref”,[UD]): Esta instrucción genera de la referencia *Ref*, una cantidad *UD* de piezas. *Ref* deberá ser una referencia sin color, ya que intentará generar la pieza del mismo color que el indicado en la parte interior del diseño. Si no existe referencia final con este color, el programa tomará la primera referencia final indicada en el apartado Materiales de PrefWise.
- **PIEZACEXT / COUTPIECE** (“Ref”,[UD]): Esta instrucción es similar a la anterior. Únicamente cambia en que el programa intentará generar la pieza del color exterior indicado en el modelo.
- **PIEZA\_EX / PIECE\_EX** (“Ref”,[UD], “Param” ): Esta instrucción genera de la referencia *Ref*, una cantidad *UD* de piezas. *Ref* deberá ser una referencia con color (referencia) de un material de *Tipo de Cálculo = Piezas*. De tratarse de una referencia base, se generará la primera referencia con color dada de alta en la aplicación *Materiales* de *PrefWise*. A diferencia de la instrucción “PIEZA” ó “PIECE”, disponemos del campo “Param”. En él podemos indicar un parámetro a rellenar por el usuario, para posteriormente utilizarlo en la programación de la PrefUserVB a conveniencia.
- **PIEZAC\_EX / CPIECE\_EX** (“Ref”,[UD], “Param”): Esta instrucción es similar a la anterior, salvo que *Ref* deberá ser una referencia base. El color de la referencia lo dictará el modelo.
- **PIEZACINT\_EX / CINNPIECE\_EX** (“Ref”,[UD], “Param”): Esta instrucción genera de la referencia *Ref*, una cantidad *UD* de piezas. *Ref* deberá ser una referencia sin color, ya que intentará generar la pieza del mismo color que el indicado en la parte interior del diseño. De la misma forma que en la instrucción “PIEZA\_EX” ó “PIECE\_EX”, el campo “Param” lo usaremos para establecer un parámetro a utilizar posteriormente en la PrefUserVB.
- **PIEZACEXT\_EX / COUTPIECE\_EX** (“Ref”,[UD], “Param”): Esta instrucción es similar a la anterior, salvo en que el programa intentará generar la pieza del mismo color que el especificado en la parte exterior del diseño, siempre que este asociado dicho color al material en PrefWise.
- **PIEZA\_EX\_U / PIECE\_EX\_U** (“Ref”, [UD], “Param”, [Unmounted]): Esta instrucción genera de la referencia *Ref*, una cantidad *UD* de piezas. *Ref* deberá ser una referencia con color (referencia) de un material de *Tipo de Cálculo = Piezas*. En él podemos indicar un parámetro a rellenar por el usuario, para posteriormente utilizarlo en la programación de la PrefUserVB a conveniencia. En el parámetro *Unmounted* indicaremos como valor “1” cuando el material vaya desmontado en el modelo, y “0” cuando vaya montado.

- **PIEZAC\_EX\_U / CPIECE\_EX\_U** (“Ref”, [UD], “Param”, [Unmounted]): Esta instrucción es similar a la anterior, salvo que *Ref* deberá ser una referencia base. El color de la referencia lo dictará el modelo. En caso de que no este asociado, el programa generará la primera referencia con color dada de alta en la aplicación Materiales de PrefWise.
- **SUPERFICIE / SURFACE** (“Ref”, [UD], [LH],[LW]): Esta instrucción genera de la referencia *Ref*, una cantidad *UD* de superficies de ancho *LH* y alto *LW*. *Ref* deberá ser una referencia con color (referencia) de un material de *Tipo de Cálculo = Superficies*. De tratarse de una referencia Base, se generará la primera referencia con color dada de alta en la aplicación *Materiales* de *PrefWise*.
- **SUPERFICIEC / CSURFACE** (“Ref”, [UD], [LH],[LW]): Esta instrucción es similar a la anterior, salvo que *Ref* deberá ser una referencia Base. El color de la referencia lo dictará el modelo.
- **SUPERFICIECINT / CINNSURFACE** (“Ref”, [UD], [LH],[LW]): Esta instrucción genera de la referencia base *Ref*, una cantidad *UD* de superficies de ancho *LH* y alto *LW*. El programa intentará generar esta referencia con el mismo color asociado al modelo en la parte interior. En caso de que este color no esté asociado en el apartado de Materiales en PrefWise para esta superficie, el programa tomará el color asociado en la primera línea del apartado Materiales.
- **SUPERFICIECEXT / COUTSURFACE** (“Ref”, [UD], [LH],[LW]): Esta instrucción es similar a la anterior, salvo por que el programa intentará generar esta referencia con el mismo color asociado al modelo en la parte exterior.
- **SUPERFICIE\_EX / SURFACE\_EX** (“Ref”, [UD], [LH],[LW], “Param”): Esta instrucción genera de la referencia *Ref*, una cantidad *UD* de superficies de ancho *LH* y alto *LW*. *Ref* deberá ser una referencia con color (referencia) de un material de *Tipo de Cálculo = Superficies*. De tratarse de una referencia Base, se generará la primera referencia con color dada de alta en la aplicación *Materiales* de *PrefWise*. El campo “Param”, como hemos comentado en casos anteriores, podemos indicar un parámetro a rellenar por el usuario, para posteriormente utilizarlo en la programación de la PrefUserVB a conveniencia. Este parámetro es el único uso que tiene.
- **SUPERFICIEC\_EX / CSURFACE\_EX** (“Ref”, [UD], [LH],[LW], “Param”): Esta instrucción es igual que la anterior salvo por el dato de que el campo “Ref” contendrá una referencia base. El programa intentará generar el material de tipo superficie con el color asociado al modelo. En caso de que no este asociado, el programa generará la primera referencia con color dada de alta en la aplicación *Materiales* de *PrefWise*.
- **SUPERFICIECINT\_EX / CINNSURFACE\_EX** (“Ref”, [UD], [LH],[LW], “Param”): Esta instrucción genera de la referencia base *Ref*, una cantidad *UD* de superficies de ancho *LH* y alto *LW*. El programa intentará generar esta referencia con el mismo color asociado al modelo en la parte interior. El campo “Param”, como hemos comentado anteriormente, podemos indicar un parámetro a rellenar por el usuario, para posteriormente utilizarlo en la programación de la PrefUserVB.
- **SUPERFICIECEXT\_EX / COUTSURFACE\_EX** (“Ref”, [UD], [LH],[LW], “Param”): Esta instrucción es similar a la anterior, salvo por que el programa intentará generar esta referencia con el mismo color asociado al modelo en la parte exterior.
- **SUPERFICIE\_EX\_U / SURFACE\_EX\_U** (“Ref”, [UD], [LH], [LW], “Param”, [Unmounted]): Esta instrucción genera de la referencia *Ref*, una cantidad *UD* de superficies de ancho *LH* y alto *LW*. *Ref* deberá ser una referencia con color (referencia) de un material de *Tipo de Cálculo = Superficies*. El campo “Param”, como hemos comentado en casos anteriores, podemos indicar un parámetro a rellenar por el usuario, para posteriormente utilizarlo en la programación de la PrefUserVB a conveniencia. Este parámetro es el único uso que tiene. En el parámetro *Unmounted* indicaremos como valor “1” cuando el material vaya desmontado en el modelo, y “0” cuando vaya montado.

- **SUPERFICIEC\_EX\_U / CSURFACE\_EX\_U** (“Ref”, [UD], [LH], [LW], “Param”, [Unmounted]): Esta instrucción es similar a la anterior, salvo que *Ref* deberá ser una referencia base. El color de la referencia lo dictará el modelo. En caso de que no esté asociado, el programa generará la primera referencia con color dada de alta en la aplicación *Materiales* de PrefWise.
- **VOLUMEN / VOLUME** (“Ref”,[Cant]): Esta instrucción genera de la referencia *Ref*, una unidad con volumen indicado en la cantidad *Cant*. *Ref* deberá ser una referencia con color (referencia) de un material de *Tipo de Cálculo = Volumen*. De tratarse de una referencia base, se generará la primera referencia con color dada de alta en la aplicación *Materiales* de PrefWise
- **VOLUMENC / CVOLUME** (“Ref”,[ Cant]): Esta instrucción es similar a la anterior, salvo que *Ref* deberá ser una referencia base. El color de la referencia lo dictará el modelo.
- **VOLUMENCINT / CINNVOLUME** (“Ref”,[ Cant]): Esta instrucción genera de la referencia *Ref*, una unidad con volumen indicado en la cantidad *Cant*. *Ref* deberá ser una referencia sin color, ya que intentará generar la pieza del mismo color que el indicado en la parte interior del diseño. Si no existe referencia final con este color, el programa tomará la primera referencia final indicada en el apartado *Materiales* de PrefWise.
- **VOLUMENCEXT / COUTVOLUME** (“Ref”,[ Cant]): Esta instrucción es similar a la anterior. Únicamente cambia en que el programa intentará generar la pieza del color exterior indicado en el modelo.
- **VOLUMEN\_EX / VOLUME\_EX** (“Ref”,[ Cant], “Param” ): Esta instrucción genera de la referencia *Ref*, una unidad con el volumen indicado en la cantidad *Cant*. *Ref* deberá ser una referencia con color (referencia) de un material de *Tipo de Cálculo = Volumen*. De tratarse de una referencia base, se generará la primera referencia con color dada de alta en la aplicación *Materiales* de PrefWise. A diferencia de la instrucción “*VOLUMEN*” ó “*VOLUME*”, disponemos del campo “*Param*”. En el podemos indicar un parámetro a rellenar por el usuario, para posteriormente utilizarlo en la programación de la PrefUserVB a conveniencia.
- **VOLUMENC\_EX / CVOLUME\_EX** (“Ref”,[Cant], “Param”): Esta instrucción es similar a la anterior, salvo que *Ref* deberá ser una referencia base. El color de la referencia lo dictará el modelo.
- **VOLUMENCINT\_EX / CINNVOLUME\_EX** (“Ref”,[Cant], “Param”): Esta instrucción genera de la referencia *Ref*, una unidad con el volumen indicado en la cantidad *Cant*. *Ref* deberá ser una referencia sin color, ya que intentará generar el material de tipo volumen del mismo color que el indicado en la parte interior del diseño. De la misma forma que en la instrucción “*VOLUMEN\_EX*” ó “*VOLUME\_EX*”, el campo “*Param*” lo usaremos para establecer un parámetro a utilizar posteriormente en la PrefUserVB.
- **VOLUMENCEXT\_EX / COUTVOLUME\_EX** (“Ref”,[Cant], “Param”): Esta instrucción es similar a la anterior, salvo en que el programa intentará generar el volumen del mismo color que el especificado en la parte exterior del diseño, siempre que este asociado dicho color al material en PrefWise.

## 5.4. Instrucción de llamada a otro Escandallo

- **ESCANDALLO / SCRIPT** (“N\_ESC”, “V1=Var1; V2=Var2;”): Esta instrucción llama al escandallo *N\_ESC* pasándole las variables que el usuario estime oportunas. Por ejemplo, L como ancho del modelo y A como alto del modelo.

## 5.5. Instrucciones para la presentación de Mensajes

- **MENSAJE / MESSAGE** (“*texto1*”): Esta instrucción escribe en pantalla un mensaje informativo.
- **ERROR** (“*texto2*”): Esta instrucción escribe en pantalla un mensaje de error.

## 5.6. Instrucción de Control de Flujo

```
SI [Condición] ENTONCES
    Lista de Instrucciones
FINSI
```

Esta instrucción ejecuta la *Lista de Instrucciones* sólo si se cumple la condición que aparece entre [ ]. La condición puede ser de tres tipos: de Medidas de una variable, de color o de opciones.

### Condición de Medidas de una variable

La condición será que las dimensiones de las cotas del modelo sean =, >, <... al valor especificado.

Ejemplo:

```
SI [A<1000] ENTONCES
    Lista de Instrucciones
FINSI
```

Ejemplo:

```
SI [A#1000] ENTONCES
    Lista de Instrucciones
FINSI
```

### Condición de Color

La condición será que el color asignado al escandallo sea igual al especificado en la condición.

Ejemplo:

```
SI [COLOR("Blanco")] ENTONCES
    Lista de Instrucciones
FINSI
```

En este caso, la *Lista de Instrucciones* se ejecutará si el color asignado al escandallo es Blanco. El color deberá estar dado de alta en la aplicación *Colores* de *PrefWise*.

### Condición de opciones

La condición será que el valor que tenga seleccionado una opción sea igual al especificado en la condición, o bien sea diferente igual al especificado en la condición.

Ejemplo:

```
SI [OPCION("Nombre_op", "Valor_op")] ENTONCES
    Lista de Instrucciones
FINSI
```

En este caso, la *Lista de Instrucciones* se ejecutará si el valor que tiene seleccionada la opción *Nombre\_op* es igual a *Valor\_op*.

Ejemplo:

```
SI [OPCION("Nombre_op", "Valor_op")=0] ENTONCES
    Lista de Instrucciones
FINSI
```

En este caso, la *Lista de Instrucciones* se ejecutará si el valor que tiene seleccionada la opción *Nombre\_op* es diferente a *Valor\_op*.

Las opciones pueden ser de dos tipos:

- *Opciones del Sistema* (o *Volátiles*): Estas opciones las define el programa, y las más comunes aparecen en la siguiente tabla (para más información acerca de estas, consultar el documento “WP2006.3.084 – Opciones Volátiles”):

Nombre de la Opción	Valores de la Opción	Descripción de la Opción
Nivel1	Nombre del Nivel 1 del Árbol de Materiales	Nombre del Nivel del Árbol de Materiales
Nivel2	Nombre del Nivel 2 del Árbol de Materiales	
Nivel3	Nombre del Nivel 3 del Árbol de Materiales	
Nivel4	Nombre del Nivel 4 del Árbol de Materiales	
Nivel5	Nombre del Nivel 5 del Árbol de Materiales	
Practicable	Derecha / Izquierda	Tipos de Apertura Practicable
Oscilobatiente	Inferior / Superior	Tipos de Apertura Oscilobatiente
Activa	Sí / No	Hoja Activa
Exterior	Sí / No	Apertura Exterior
Corredera	Arriba / Abajo / Izquierda / Derecha	Tipos de Apertura de Corredera
Elevable	Sí / No	Tipos de Apertura elevable
Vidrio	Sí / No	Añadir Vidrios
Cajón	Sí / No	Añadir Cajón de Persiana

- Opciones creadas por el usuario. Estas opciones las define el usuario en la aplicación Opciones Globales de PrefWise.

Es posible encadenar varias condiciones mediante los nexos Y y O:

Ejemplo:

```
SI [OPCION("Practicable", "Izquierda")=0 Y
    OPCION("Practicable", "Derecha")=0] ENTONCES
    Lista de Instrucciones
FINSI
```

La *Lista de Instrucciones* se ejecutará si el modelo no es practicable a derechas ni practicable a izquierda.

Ejemplo:

```
SI [A<1500 O A=1500] ENTONCES
    Lista de Instrucciones
FINSI
```

La *Lista de Instrucciones* se ejecutará si la altura del modelo es superior o igual a 1,5 m.



Igualmente, es posible anidar varias instrucciones de control de flujo:

Ejemplo:

```
SI [Condición1] ENTONCES
  SI [Condición2] ENTONCES
    Lista de Instrucciones
  FINSI
FINSI
```

La *Lista de Instrucciones* se ejecutará si se cumplen simultáneamente la *Condición1* y la *Condición2*.

## 5.7. Instrucciones de Asignación

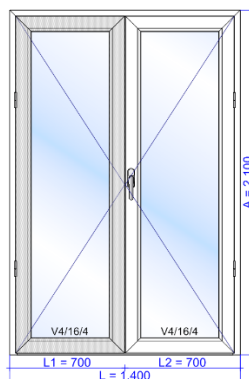
- **ESTABLECEOPCION / SETOPTION** (“Nombre\_op”, “Valor\_op”): Esta instrucción asigna a la opción *Nombre\_op* el valor *Valor\_op*. En lugar de escribir un valor entre “”, puede escribir el nombre de otra opción entre <>. En este caso, se le asignará a la opción el valor que se encuentre seleccionado en la opción entre <>.

Veamos este ejemplo de escandallo que establecerá una opción “Zapata” a un valor en concreto, según el nivel (HL) que tome una hoja de apertura corredera.

```
SI [HL=2] ENTONCES
  ESTABLECEOPCION("Zapata", "SIN PERFIL");
FINSI
SI [HL=1] ENTONCES
  ESTABLECEOPCION("Zapata", "COS78954");
FINSI
```

- **ESTABLECEOPCIONNUMERICA / SETNUMERICOPTION** (“Nombre\_op”, “Valor\_op”): Esta instrucción asigna a la opción Numérica *Nombre\_op* el valor *Valor\_op*. En lugar de escribir un valor entre “”, puede escribir una expresión numérica entre [ ]. Esta instrucción suele emplearse en los escandallos de mallorquina para indicar los valores del paso de lamas, del número de lamas....
- **ESTABLECEOPCIONVECINOS / SETOPTIONNEIGHBORS** (“Nombre\_op”, “Valor\_op”): Esta instrucción va a recorrer todos los tramos vecinos del hueco actual (del hueco donde se está ejecutando el escandallo) y para cada uno de ellos buscará si tienen la opción “Nombre\_op” requerida, para cambiarla y disparar si es preciso al valor “Valor\_op”. El programa tomará los delimitadores como límite para establecer la opción de tramos vecinos.

Es decir, teniendo este modelo, con la hoja izquierda aplicando esta instrucción por escandallo constructivo:



Como entre la hoja izquierda y derecha tiene delimitador (batiente), en la hoja derecha no revisará la opción asociada al tramo, ni en el propio delimitador como tal.

Para este ejemplo, en el marco, mirara si en el tramo lateral izquierdo, inferior y superior tiene asociada la opción indicada en ESTABLECEOPCIONVECINOS. Y si la tiene asociada, entonces la disparará.

Solo aplicara la instrucción del hueco del escandallo hacia afuera. Por tanto, si esta hoja tuviera una contraventana incluida en sus tramos, NO miraría si en esos tramos tiene asociada la opción comentada en la instrucción.

- **ESTABLECECOTA / SETDIMENSION** ("NombreCota",[Expresión]): Le asigna a la cota *NombreCota* la medida indicada por la *Expresión*. Esta medida puede ser un número o una fórmula que lo calcule.

Veamos este ejemplo de escandallo que aumentará el tamaño total del modelo, según la guía que haya sido asociada.

```
SI [OPCION ("Guías","Guía Estrecha")] ENTONCES
    ESTABLECECOTA("L", [L + 80]); %porque queremos tomar L+80 para esta guia
FINSI
SI [OPCION ("Guías","Guía Ancha")] ENTONCES
    ESTABLECECOTA("L", [L + 120]); %porque queremos tomar L+120 para esta guia
FINSI
```

Este escandallo deberemos asociarlo **por vínculo constructivo al contorno** del modelo.

- **ESTABLECESUBCOTA / SETSUBDIMENSION** ("NombreCota",[OrdenSubCota],[Expresión]): Le asigna a la subcota número *OrdenSubCota* de la cota *NombreCota* la medida indicada por *Expresión*. Un ejemplo del uso de esta función es el siguiente:

Necesitamos establecer las subcotas de L en función de una formula específica. Por esto, realizaremos un escandallo especificando esto:

```
ESTABLECESUBCOTA("L", [ 1 ], [ (L+106)/4 - 15 ] );
ESTABLECESUBCOTA("L", [ 2 ], [ (L-106)/4 + 15 ] );
ESTABLECESUBCOTA("L", [ 3 ], [ (L-106)/4 + 15 ] );
ESTABLECESUBCOTA("L", [ 4 ], [ (L+106)/4 - 15 ] );
```

Como en el caso anterior, este escandallo deberemos asociarlo **por vínculo constructivo al contorno** del modelo.

- **ESTABLECEERROR / SETERROR** ([Codigo\_Error]); Esta instrucción asigna un error al modelo. El parámetro “Codigo\_Error”, así como su severidad, etc.... será configurado por el usuario desde el apartado de *Plausibilidad* en *PrefWise*. Para más información acerca de esto, consultar el documento “WP2006.3.070 - Plausibilidades”.
- **ESTABLECEERROREX / SETERROREX** ([Codigo\_Error], “Extensión”); Esta instrucción es similar a la anterior. Se diferencia en que es posible añadir un texto adicional en el campo “Extensión” al error que queremos lanzar.

Ejemplo de un escandallo donde hace saltar una plausibilidad en función del tamaño de la hoja, este escandallo lo tendremos que asociar a la hoja mediante **vínculo común**.

```
SI [(L>1600)O(A>1600)] ENTONCES
    ESTABLECEERROREX ([1600], "La hoja excede de las dimensiones máximas");
FINSI
```



## 5.8. Instrucciones de Función

- **ENTERO / INTEGER** (*Expresión*): Esta instrucción devuelve la parte entera de *Expresión*. En *Expresión* puede escribir una fórmula numérica o un número (siendo el signo decimal un punto “.”y no una coma “,”).

Ejemplo: ENTERO (1000/3) devuelve el valor 333.
- **REDONDEA / ROUND** (*Expresión*): Esta instrucción devuelve el valor entero que resulta de redondear al entero más próximo el valor de *Expresión*. En *Expresión* puede escribir una fórmula o un número (siendo el signo decimal un punto “.”y no una coma “,”).

Ejemplo: REDONDEA (14/3) devuelve el valor 5.
- **NEG** (*Expresión*): Esta instrucción devuelve el valor en negativo especificado en *Expresión*. También se puede asociar, en vez de un valor constante, una variable descrita en el escandallo o una formula.

Ejemplo: NEG (3-1) devuelve el valor “-2”.
- **TRUNCATE** (*Expresión, Decimales*): Esta función trunca el valor real obtenido de *Expresión* a un valor con los decimales que se le dice en el parámetro Decimales.

Por ejemplo:

TRUNCATE ( 435.44568, 2 ) = 435.44

TRUNCATE ( 435.44568, 3 ) = 435.445
- **ABS** (*Expresión*): Esta instrucción devuelve el valor absoluto de la expresión. El resultado es entero.

Por ejemplo, las instrucciones ABS(-1.23) y ABS(1.23) aplican la función ABS a un número positivo y a un número negativo. También se puede asociar, en vez de un valor constante, una variable descrita en el escandallo o una formula. En esta ocasión, ambos ejemplos devuelven 1,23.
- **SQRT** (*Expresión*): Esta función numérica devuelve la raíz cuadrada de una expresión numérica. En *Expresión* se puede poner una fórmula o un número (los decimales separados por punto, por ejemplo 4.6).

Este ejemplo devuelve la raíz cuadrada de un literal numérico. El resultado devuelto es 12.

SQRT(144);
- **SIN** (*Expresión*): Esta función devuelve el seno trigonométrico del ángulo dado, en radianes, en una expresión numérica. En *Expresión* se puede poner una fórmula o un número (los decimales separados por punto, por ejemplo 4.6).

Por ejemplo, SIN (45.175643) devuelve 0.929607.
- **FACT** (*Expresión*): Esta función devuelve el factorial de un número entero positivo, es decir, se define como el producto de todos los números naturales anteriores o iguales a él. En *Expresión* se puede poner una fórmula o un número (los decimales separados por punto, por ejemplo 4.6).

Por ejemplo, FACT (12) devuelve 479001600.
- **COS** (*Expresión*): Es una función matemática que devuelve el coseno trigonométrico del ángulo especificado, expresado en radianes, en la expresión dada. En *Expresión* se puede poner una fórmula o un número (los decimales separados por punto, por ejemplo 4.6).

Por ejemplo, COS (14.78) devuelve -0.599465.
- **LN** (*Expresión*): Devuelve el logaritmo natural o logaritmo neperiano de una expresión numérica. En *Expresión* se puede poner una fórmula o un número (los decimales separados por punto, por ejemplo 4.6).

Por ejemplo, la función LN (42) devuelve el valor 3,737766961828337.

- **EXP** (*Expresión*): Devuelve el exponente de la base e de una expresión numérica. La función EXP complementa la acción de la función LN; también se suele llamar antilogaritmo.

En estos ejemplos se aplica la función EXP a cero y a valores positivos y negativos.

EXP (74)

Devuelve 1,373382979540176E+32.

EXP (-27)

Devuelve 1,879528816539083E-12.

EXP (0)

Devuelve 1.

- **POW** (*Expresión, Potencia*): Devuelve el resultado de elevar una expresión numérica a una determinada potencia. La evaluación del parámetro de potencia debe devolver un entero.

Por ejemplo, la función POW (4,3) eleva 4 a la potencia 3 y devuelve 64.

- **LOG** (*Expresión*): Devuelve el logaritmo natural de la expresión float especificada. La constante e (2.71828182845905...) es la base de los logaritmos naturales. La base de los logaritmos naturales es la constante e (2.71828182845905...).  $\text{LOG}(e) = 1.0$ . El logaritmo natural del exponencial de un número es el propio número:  $\text{LOG}(\text{EXP}(n)) = n$ . Y el exponencial del logaritmo natural de un número es el propio número:  $\text{EXP}(\text{LOG}(n)) = n$ .

La función LOG (97.34) devuelve el valor 1,988291341907488.

- **TAN** (*Expresión*): Esta función devuelve la tangente trigonométrica del ángulo dado, en radianes, en una expresión numérica. En *Expresión* se puede poner una fórmula o un número (los decimales separados por punto, por ejemplo 4.6).

Por ejemplo, TAN (15) devuelve 0.267949.

- **ATAN** (*Expresión*): Devuelve el ángulo en radianes cuya tangente es una expresión especificada. También se denomina arco tangente. En *Expresión* se puede poner una fórmula o un número (los decimales separados por punto, por ejemplo 4.6).

Por ejemplo, ATAN (-45.01) devuelve -1.54858.

- **ASIN** (*Expresión*): Devuelve el ángulo, expresado en radianes, cuyo seno es la expresión numérica especificada. También se denomina arco seno. En *Expresión* se puede poner una fórmula o un número (los decimales separados por punto, por ejemplo 4.6).

Por ejemplo ASIN (-1.0) devuelve -1.5708

- **ACOS** (*Expresión*): Función matemática que devuelve el ángulo, en radianes, cuyo coseno es la expresión especificada; también se denomina arco coseno. En *Expresión* se puede poner una fórmula o un número (los decimales separados por punto, por ejemplo 4.6).

Por ejemplo, la función ACOS (-1.0); devuelve 3.14159.

## 5.9. Instrucciones de Tablas

- **TABLA / TABLE** ("*Nombre\_tabla*",[L],[A]): Esta instrucción llama a la tabla *Nombre\_tabla* pasándole las variables L y A, de forma que se generarán los materiales listados en la tabla para los valores de L y A.

## 5.10. Instrucciones de Mano de Obra

- **MANOOBRA / WORKFORCE** (*"PuestoMO",[seg]*): Esta instrucción asigna al puesto de mano de obra *PuestoMO* una duración en segundos igual a *seg*.

## 5.11. Instrucciones para indicar la superficie

- **NOTIFICASUPERFICIE / SETSURFACE** (*[L],[A]*): Esta instrucción se utiliza para que en el escandallo se conozca la superficie que se está generando cuando el escandallo se presupuesta por metro cuadrado (Precios Netos). L y A serán las variables que en el escandallo indican el ancho y alto de la superficie generada.

## 5.12. Instrucciones para la generación de secciones

- **SECCION / SECTION** (*"Referencia",[Descuento],[Descuento Z], [Tramo], [Rotación], [Reflejo]*): Esta instrucción se utiliza para indicar la sección del material generado por escandallo. El Tramo puede ser 1, 2, 3 o 4. La Rotación se indica en grados y, en el Reflejo 1 significa sí y 0 significa no.

## 5.13. Instrucciones para la Obtención de Datos

- **OBTENEROPCIONNUMERICA / GETNUMERICOPTION** (*"Nombre\_Opción"*): Esta instrucción devuelve el valor asociado a la opción de tipo numérica especificada en la variable *"Nombre\_Opción"*. Esta instrucción se puede usar para asociarle el valor a una variable de usuario definida con SEA o que se puede usar entre "[ ]" (corchetes) como función. Veamos los posibles casos.

Para conseguir el valor de una opción numérica en una variable, existe la posibilidad de utilizar la instrucción **OBTENEROPCIONNUMERICA** o **GETNUMERICOPTION**, por ejemplo:

```
SEA MiVariable = GETNUMERICOPTION ("Numerica");  
  
PIECE("MiPieza",[MiVariable]);
```

También se puede utilizar estas funciones directamente en cualquier otra instrucción entre corchetes, por ejemplo:

```
PIECE("MiPieza",[GETNUMERICOPTION ("Numerica")]);
```


## 5.14. Comentarios

Las líneas que en el escandallo comienzan por el símbolo **%** no se ejecutan sino que son tomadas como comentarios.

## 6. Historial de escandallos

PrefSuite almacena en la base de datos un historial del código insertado y modificado en los escandallos.

De esta forma, se puede conocer si un escandallo ha sufrido cambios, que usuario ha llevado a cabo estas modificaciones, y recuperar el código de este escandallo antes del cambio aplicado.

Desde PrefGest, está disponible el acceso al historial de cada escandallo desde “Artículos > Escandallos”, en la pestaña “Código” del escandallo editado, pulsando el botón .

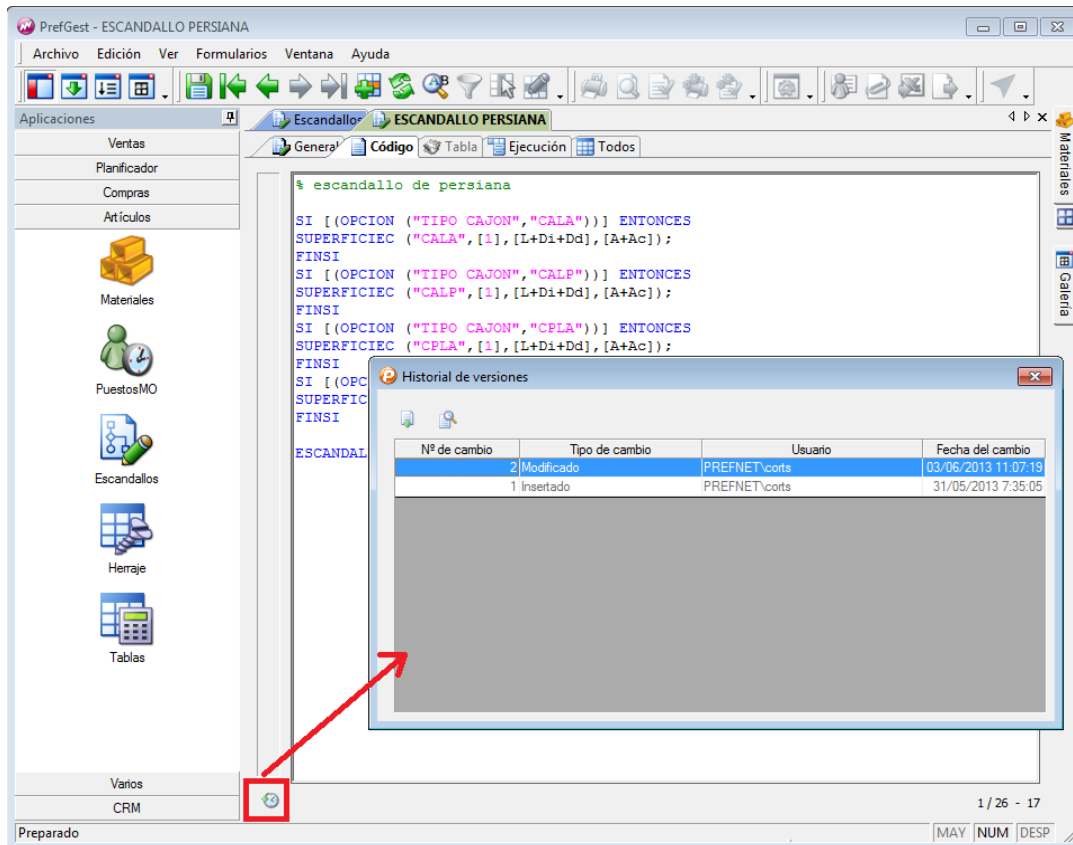


Figura 1. Historial de escandallos, accesible desde PrefGest.

Desde PrefCAD, en la ventana “Editor”, está también accesible el historial del escandallo editado al pulsar el botón marcado en la imagen siguiente.

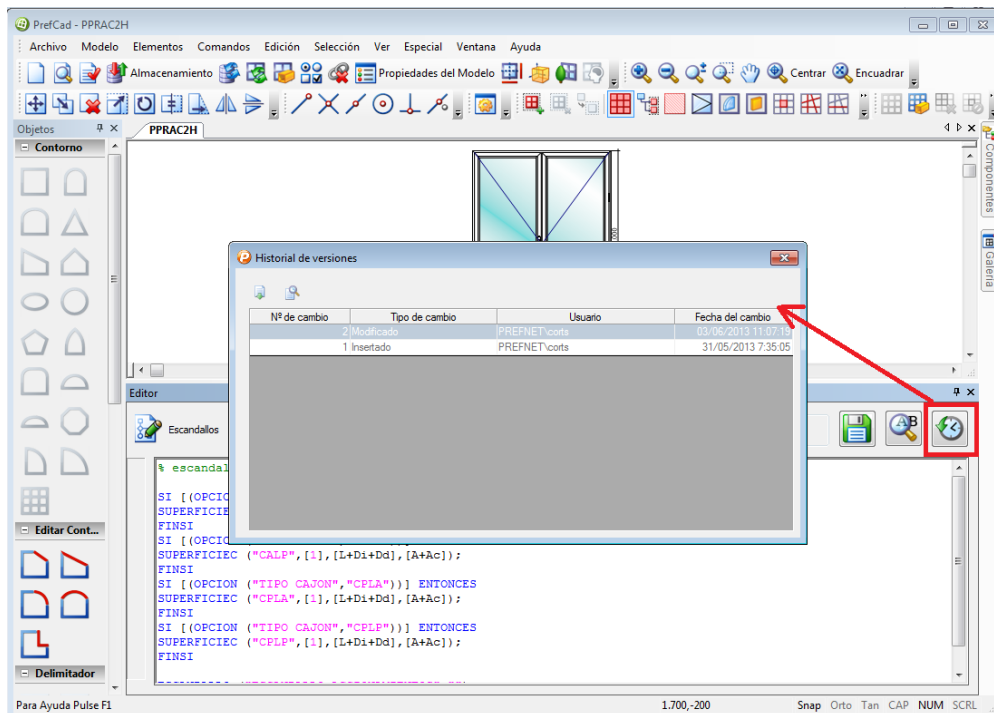



Figura 2. Historial de escandallos, accesible desde PrefCAD.

Al acceder a la ventana de historial, del escandallo editando, los campos disponibles en esta son:

- N° de Cambio: Indica el orden de cada cambio almacenado.
- Tipo de cambio: Indica el tipo de cambio aplicado en el escandallo. Admite dos posibles valores: insertado y modificado.
- Usuario: Indica el usuario SQL o de dominio que aplicó el cambio.
- Fecha de cambio: Indica fecha y hora en la que se llevó a cabo el cambio.

En esta ventana, bien sea en PrefCAD o en PrefGest, seleccionando la línea y pulsando el botón “Obtener versión” , se muestra en pantalla el código almacenado en este punto del historial.

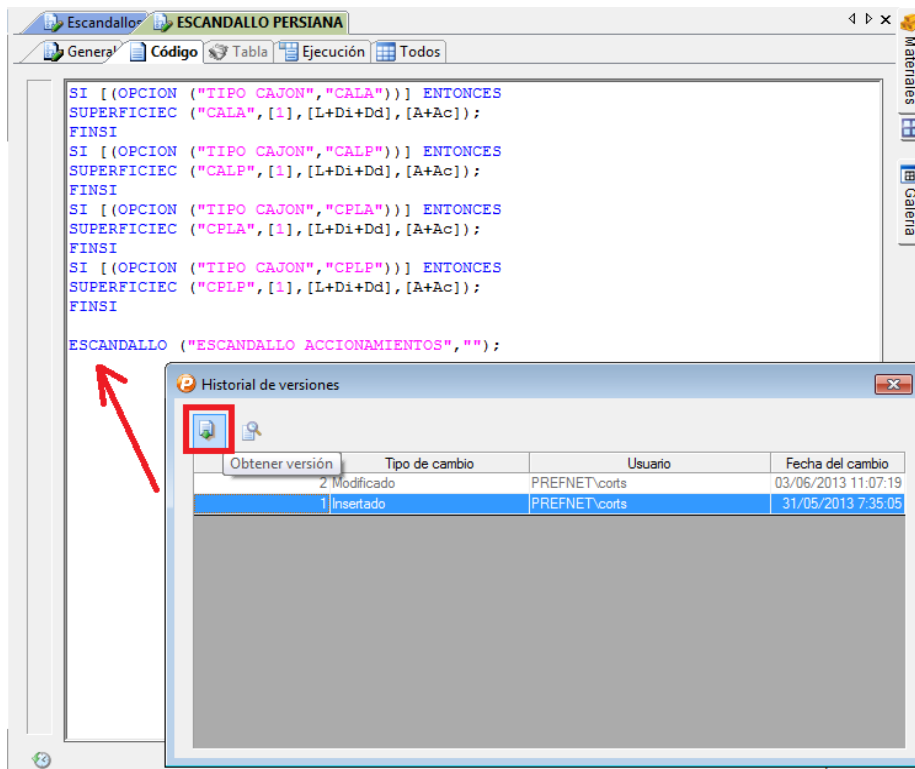



Figura 3. Historial de escandallos. Botón “Obtener versión”.

De la misma forma, seleccionando la línea y pulsando el botón “Ver versión” , se abre un archivo TXT con el código de esta versión del historial.

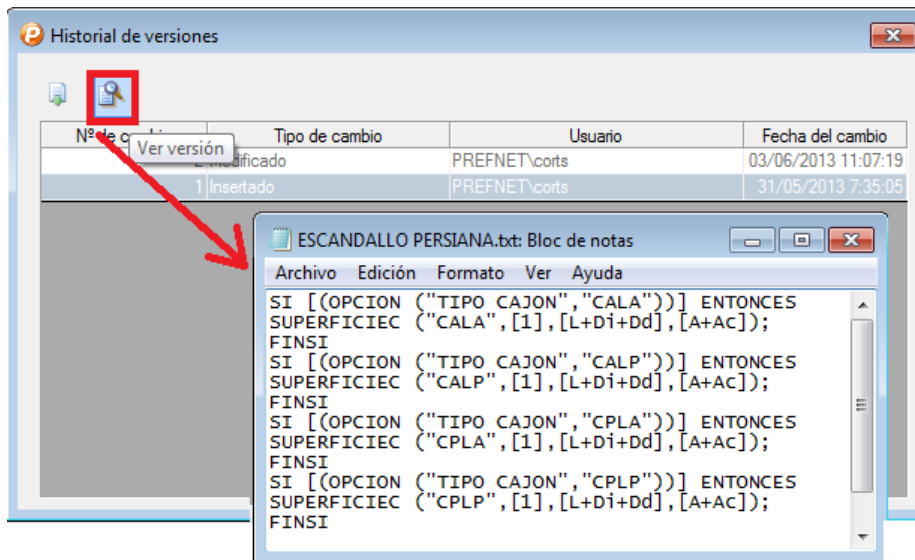
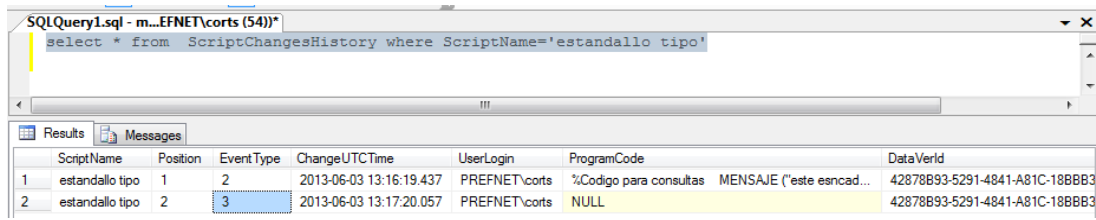


Figura 4. Historial de escandallos. Botón “Ver versión”.

Este historial se almacena internamente en la tabla “ScriptChangesHistory”. Al eliminar un escandallo, la información en esta tabla no se eliminará.

Al eliminar un escandallo, esta acción también quedará registrada. Para estos casos, el campo “EventType” tiene valor “3”.



The screenshot shows a SQL query window with the following query: `select * from ScriptChangesHistory where ScriptName='estandallo tipo'`. The results are displayed in a table with the following columns: ScriptName, Position, EventType, ChangeUTCTime, UserLogin, ProgramCode, and DataVerid. The results are as follows:

ScriptName	Position	EventType	ChangeUTCTime	UserLogin	ProgramCode	DataVerid
estandallo tipo	1	2	2013-06-03 13:16:19.437	PREFNET\corts	%Codigo para consultas MENSAJE ("este esncad...	42878B93-5291-4841-A81C-18BBB3
estandallo tipo	2	3	2013-06-03 13:17:20.057	PREFNET\corts	NULL	42878B93-5291-4841-A81C-18BBB3

Figura 5. Información de la tabla "ScriptChangesHistory".