



CURSO INTOUCH BASICO V7.1

MODULO 1. INTRODUCCION	4
Sección 1. Introducción al Curso	4
Sección 2. Introducción a InTouch	4
Requerimientos del Sistema	4
Instalación	4
La Licencia de Wonderware	4
Creación de una Aplicación	4
MODULO 2. ENTORNO DE DESARROLLO	7
Sección 1. Elementos de WindowMaker	7
Menús de WindowMaker	7
Sección 2. Usando WindowMaker	8
Tipos de Ventanas	8
Los Elementos Wizards	11
MODULO 3. DICCIONARIO DE TAGNAMES	13
Sección 1. Definición del Tagname	13
Acceso	13
Definición de los Tagnames	13
Definición de las Características	14
Campos a Rellenar del Tagname	14
MODULO 4. ANIMATION LINKS	17
Sección 1. ¿Qué son las Animation Links?	17
Sección 2. Animando Objetos	18
Sección 3. Edición de Links y de Tags	23
Sustituir Texto	23
Sustituir Tagnames	23
Importar y Exportar Ventanas	24
Convertir Placeholder Tagnames	24
Borrar Tagnames	24
Referencias Cruzadas de InTouch	25
MODULO 5. InTouch QUICKSCRIPTS	32
Sección 1. Tipos de Scripts	32
Sección 2. Funciones Scripts	32
USO DE VARIABLE INTERNAS	38
MODULO 6. ALARMAS Y EVENTOS	39
Sección 1. Alarmas	39
Tipos de Alarmas	39
Prioridades de las Alarmas	39
Grupos de Alarmas	39
Cómo Crear Grupos de Alarmas	40
Definición de una Condición de Alarma en un Tagname	40
Creación de un Objeto de Alarmas	41
Configuración de un Objeto de Alarmas	42
Configuración de Alarmas/Eventos	44
Creación de Condiciones de Reconocimiento	44
Campos de las Alarmas	44
Sección 2. Eventos	45

MODULO 7. CURVAS HISTORICAS Y REALES	50
Sección 1. Curvas en Tiempo Real	50
Creación de una Curva en Tiempo Real	50
Configuración de la Curva en Tiempo Real	51
Sección 2. Curvas Históricas	53
Configurar HistoricalLogging	53
Utilización del Wizard de Curva Histórica	53
Creación de la Curva Histórica (sin utilizar el Wizard)	56
Campos (.Fields) de la Tendencias Históricas	58
La Utilidad HistData	59
MODULO 8. COMUNICACIONES I/O	63
Sección 1. La Comunicación DDE	63
InTouch Access Name	63
MODULO 9. SEGURIDAD	66
Sección 1. Características de la Seguridad	66
Sección 2. Configuración de la Aplicación	66
Configurar WindowMaker	66
Configurar WindowViewer	67
Configurar Alarms	68
Configurar HistoricalLogging	68
Distributed Name Manager	69
Install Wizards	69

MODULO 1. INTRODUCCION

Sección 1. Introducción al Curso

La duración de este curso es de 15 horas.

Este curso está dividido en módulos y secciones

Se llevarán a cabo ejercicios prácticos sobre cada módulo o sección explicada

A la conclusión del curso, usted será capaz de desarrollar una aplicación MMI utilizando elementos InTouch.

Sección 2. Introducción a InTouch

InTouch es un paquete de software utilizado para crear aplicaciones de interface hombre-máquina bajo entorno *PC*. **InTouch** utiliza como sistema operativo el entorno *WINDOWS 95/98/NT/2000*. El paquete consta básicamente de dos elementos: *WINDOWMAKER* y *WINDOWVIEWER*. *WINDOWMAKER* es el sistema de desarrollo. Permite todas las funciones necesarias para crear ventanas animadas interactivas conectadas a sistemas de e/s externos o a otras aplicaciones *WINDOWS*. *WINDOWVIEWER* es el sistema runtime utilizado para rodar las aplicaciones creadas con *WINDOWMAKER*.

En cualquier pantalla de *WINDOWMAKER* disponemos de una ayuda sensitiva pulsando la tecla F1.

Requerimientos del Sistema

- Cualquier *PC* compatible IBM con procesador Pentium 200 MHz o superior
- Mínimo 500 Mb de disco duro
- Mínimo 64 Mb RAM
- Adaptador display SVGA (recomendado 2 Mb mínimo)
- Puntero (mouse, trackball, touchscreen)
- Adaptador de red
- Microsoft Windows W95/98 SE o NT

Instalación

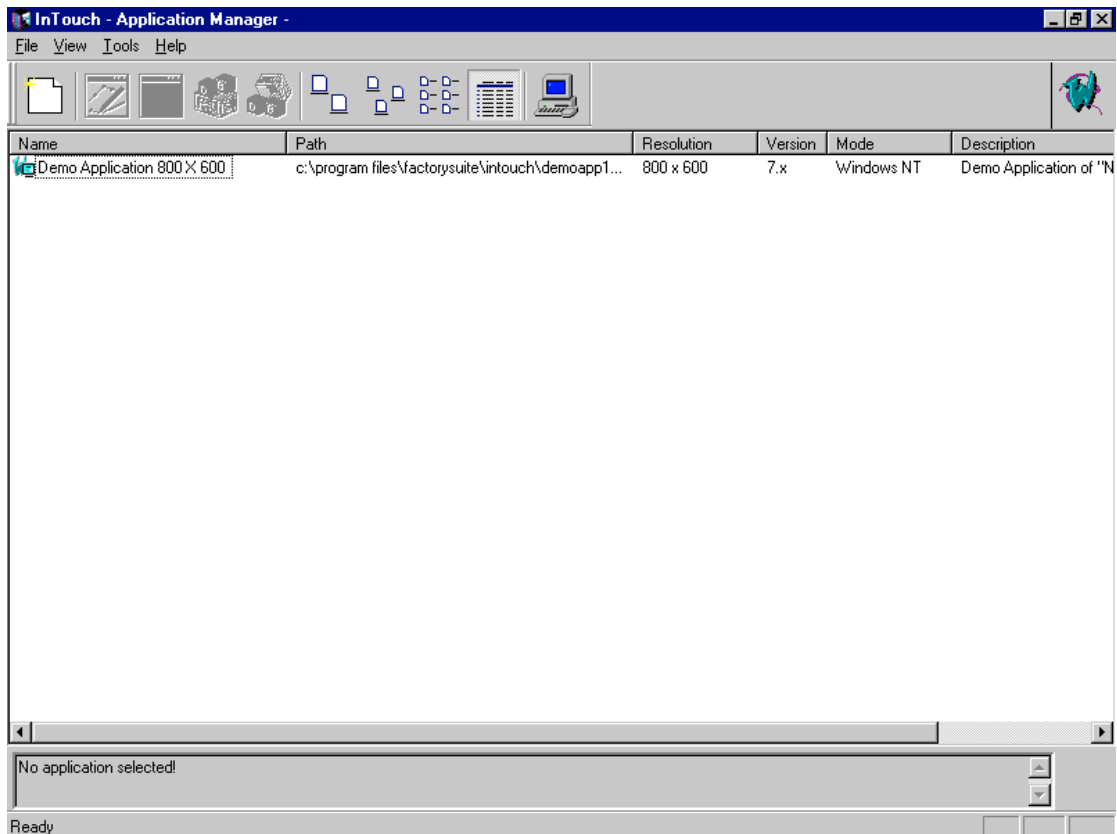
InTouch dispone de un sencillo programa de instalación que además detecta el sistema operativo sobre el que el programa se va a instalar. El CD-ROM dispone de un autoarranque.

La Licencia de Wonderware

El paquete **InTouch** viene protegido por una llave (licencia) conectable al puerto paralelo de su ordenador. Existen distintos tipos de llaves. De acuerdo a la que Ud. conecte podrá disponer de unas u otras funciones de **InTouch**.

Creación de una Aplicación

Una vez instalado el paquete **InTouch**, ya podemos crear una aplicación. Para ello, es necesario entrar en **InTouch** desde *WINDOWS* pinchando dos veces con el ratón sobre el símbolo de **InTouch**. En su monitor aparecerá la siguiente pantalla:



Esta es la pantalla principal de *Application Manager* para la entrada a **InTouch**. Desde aquí podemos seleccionar cualquiera de las aplicaciones de nuestro ordenador previamente creadas, o bien crear una aplicación nueva. Para ello, seleccione *FILE* → *NEW* para acudir al asistente de generación de aplicaciones, que le permitirá además dar un nombre y comentario a la nueva aplicación creada (muy útil tanto para documentación posterior, como para selección desde el application manager). **InTouch** volverá a la pantalla principal de *Application Manager* y mostrará en la lista el subdirectorio aplicación *CURSILLO*. Observe estos dos iconos en la barra de herramientas de *Application Manager*:



Este es el icono de *WINDOWMAKER* o creador de aplicaciones. Una vez seleccionada la aplicación que desea crear o modificar, pinche sobre este icono para llevar a cabo su trabajo



Este es el icono de *WINDOWVIEWER* o runtime. Una vez seleccionada la aplicación que desea monitorizar, pinche sobre este icono. Esta aplicación debe haber sido previamente creada, por lo que este icono no estará accesible cuando seleccione una nueva aplicación

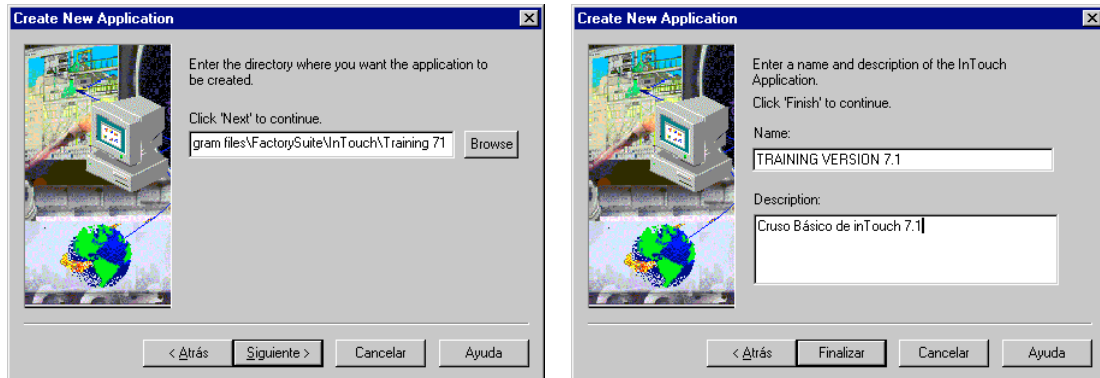
Al pinchar sobre el icono de *WindowMaker*, **InTouch** creará automáticamente un subdirectorio con este nombre, e incluirá en él los ficheros de trabajo.



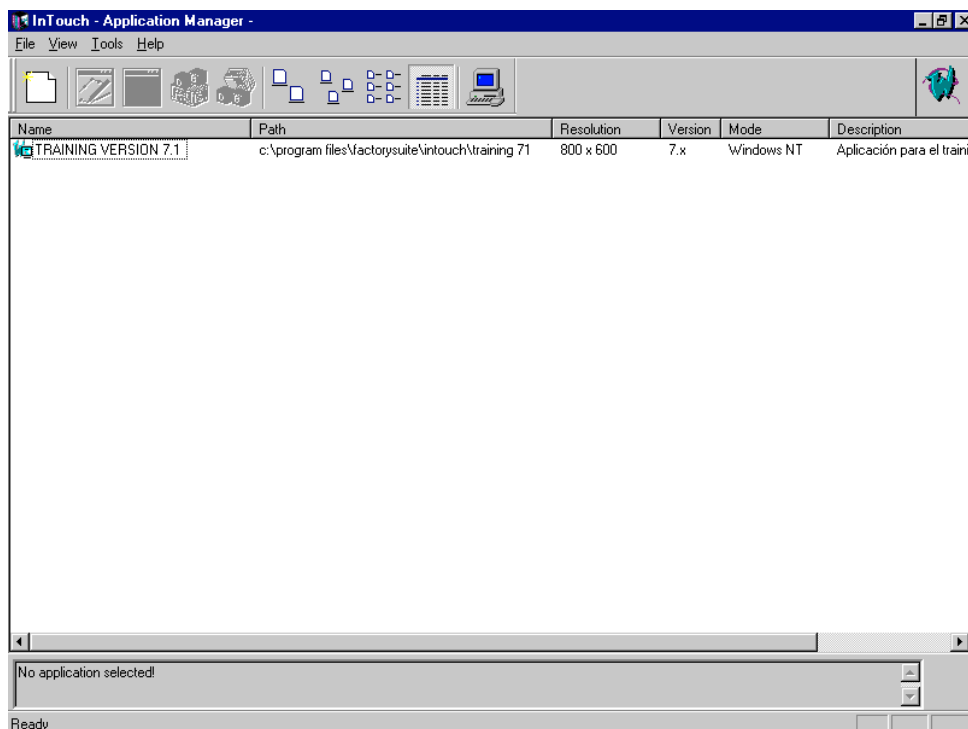
EJERCICIO 1. CREACION DE UNA APLICACION

Cree una nueva aplicación **InTouch**.

Partiendo del *Application Manager*, seleccione **FILE** → **NEW** para crear una nueva aplicación. Automáticamente, un asistente le guiará en la creación de esta aplicación.



Tras pulsar **Finalizar**, la nueva aplicación quedará en la lista de aplicaciones de *Application Manager*. Ya dispone del icono de WindowMaker activado para poder crear la aplicación. **InTouch** se encarga de crear la carpeta con el nombre de la aplicación.



MODULO 2. ENTORNO DE DESARROLLO

Sección 1. Elementos de WindowMaker

Menús de WindowMaker


File.- Manejo de ficheros y de ventanas. Contiene los siguientes elementos:

New Window	Crear una nueva ventana
Open Window	Abrir una ventana existente
Save Window	Salvar una ventana
Close Window	Cerrar una ventana. Si no está salvada, InTouch nos consultará
Delete Window	Borra una ventana
Save Window As	Salva una ventana con un nombre distinto
Save All Windows	Salva todas las ventanas abiertas
WindowViewer	Salta o ejecuta el programa WindowViewer
Print	Permite imprimir: Información sobre los tags, ventanas y también scripts
Export Window	Exporta ventana a otra aplicación InTouch
Import	Importa ventana de otra aplicación InTouch
Exit	Sale de WindowMaker

Edit.- Contiene una serie de comandos para editar los objetos de la ventana. Con estas funciones, podemos editar los objetos que se encuentren seleccionados

Undo	Permite deshacer la última acción de edición. Dispone de hasta 25 niveles
Nothing to redo	Rehace la última acción de edición deshecha. Dispone de hasta 25 niveles
Duplicate	Duplica el/los objeto/s seleccionado/s
Cut	Cortar al portapapeles
Copy	Copiar al portapapeles
Paste	Pegar al portapapeles
Erase	Borrar
Import Image	Importar imagen
Paste Bitmap	Pegar Bitmap
Bitmap Original Size	Devolver al bitmap su tamaño original tras haberlo pegado
Edit Bitmap	Editar bitmap tras haberlo pegado
Select All	Seleccionar todo
Links	Links
Enlarge radius	Agrandar el radio de una curva del objeto seleccionado
Reduce radius	Reducir el radio de una curva del objeto seleccionado
Reshape Object	Deformar el objeto
Add Point	Añadir punto (en un objeto polígono o polilínea)
Del point	Borrar punto (en un objeto polígono o polilínea)
Symbol Factory	Llamada al programa Symbol Factory

View.- Con el menú *VIEW*, definimos qué utilidades o elementos de *WindowMaker* queremos tener visibles mientras programamos.

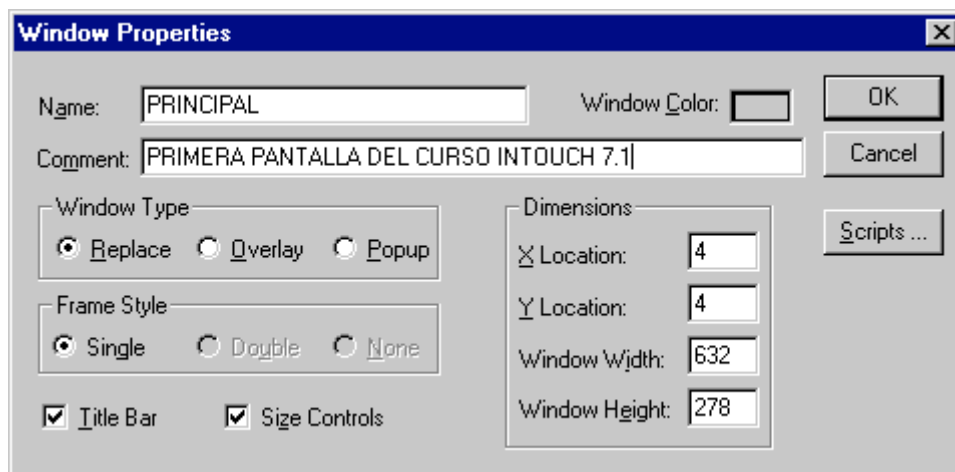
ARRANGE	Este menú contiene comandos que permiten "arreglar" los objetos seleccionados. Podemos, por ejemplo, alinear objetos, rotar o superponer.	
	MAKE SYMBOL	Este comando permite combinar varios objetos seleccionados y convertirlos en un sólo objeto llamado "símbolo". Un símbolo puede estar compuesto por varios símbolos y/o múltiples objetos.  Seleccionar todos los objetos MAKE SYMBOL Al símbolo se le pueden asignar <i>animation links</i>
	BREAK SYMBOL	Permite dividir un símbolo previamente creado en los objetos originales
	MAKE CELL	Este comando permite combinar objetos individuales y/o símbolos en una única unidad, llamada "celda". A una celda no se le pueden asignar <i>animation links</i>
	BREAK CELL	Permite dividir una celda previamente creada en los objetos originales
TEXT	Permite modificar la fuente del objeto texto seleccionado, así como ponerlo en negrita, subrayado, itálica, justificarlo y modificar su tamaño	
LINE	Permite modificar la forma de la línea del objeto seleccionado	
SPECIAL	Contiene menús muy importantes de WM, que son explicados posteriormente	
WINDOWS	Permite moverse rápidamente de una a otra ventana activa	
HELP	Permite acceder a una ayuda sensitiva en cualquier momento de la programación	
RUNTIME!	Permite un acceso directo y rápido a la emulación de la aplicación	

Sección 2. Usando WindowMaker

Tipos de Ventanas

InTouch trabaja con ventanas o pantallas. Estas ventanas disponen de:

- Elementos animados
- Tendencias gráficas y alarmas
- Lógica Asociada



Antes de empezar a dibujar, es necesario definir la pantalla sobre la que vamos a trabajar. Las pantallas pueden ser de tres tipos:

<i>Replace</i>	Cierra cualquier otra ventana que corte cuando aparece en pantalla, incluyendo ventanas tipo <i>popup</i> u otras tipo <i>replace</i>
<i>Overlay</i>	Aparece sobre la ventana displayada. Cuando cerramos una ventana tipo <i>overlay</i> , cualquier ventana que estuviera <i>escondida</i> bajo la <i>overlay</i> será restablecida. Seleccionando cualquier porción o parte visible de una ventana debajo de la <i>overlay</i> , provocará que esta ventana pase a ser considerada activa
<i>Popup</i>	Similar a la <i>overlay</i> , pero en el caso de <i>popup</i> la ventana siempre queda por encima de las demás, y no desaparece ni aunque pinchemos con el ratón sobre otra. Normalmente será necesario hacer desaparecer la ventana <i>popup</i> antes de que aparezca otra.

WINDOWMAKER (WM) de **InTouch** es una herramienta de dibujo basada en gráficos por objetos, en lugar de en gráficos por pixels. Básicamente podemos decir que creamos objetos (círculos, rectángulos, etc.) independientes unos de otros. Ello facilita la labor de edición del dibujo y, lo que es más importante, permite una enorme sencillez y potencia en la animación de cada uno de los objetos, independientemente o por grupos.

Para ello, **WM** dispone de una Barra de Herramientas de Dibujo, que permite una edición rápida de cualquier elemento.



El funcionamiento de esta barra es muy sencillo, ya que seleccionamos con el cursor aquel elemento que deseamos colocar en pantalla, disponiendo siempre de un texto inferior que indica la utilidad de cada herramienta. A continuación se explica cada uno de estos elementos:

1. SELECT MODE

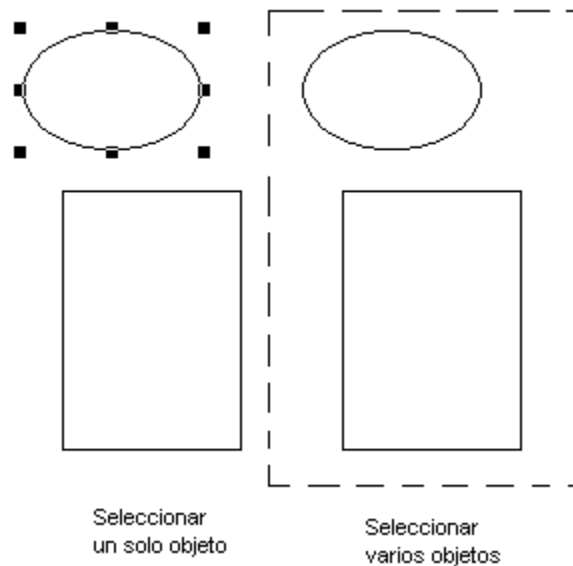


Es el primer elemento de la caja, y se utiliza para seleccionar, mover y modificar el tamaño de los objetos.

Para seleccionar o modificar el tamaño de un objeto existen dos métodos: el más sencillo consiste en colocarse encima del objeto que deseamos y pulsar el botón izquierdo del ratón. Con ello el objeto queda seleccionado, y podemos modificar su tamaño (extendiendo desde cualquier punto externo del objeto) o simplemente moverlo.

Existe un segundo método, y es utilizando el modo *Select Mode*. Escoja la herramienta *select mode* y seleccione la parte del dibujo que desee creando un rectángulo. Ello seleccionará todos los objetos incluidos en ese rectángulo. De este modo, podemos seleccionar varios objetos.

Alrededor del objeto seleccionado podrá observar varios pequeños cuadros negros. Estos cuadros son llamados *tiradores*, y son los utilizados para modificar el tamaño del objeto. Cuando un objeto se encuentra rodeado de *tiradores* significa que se encuentra seleccionado.



2. RECTÁNGULO
3. RECTÁNGULO CON ANGULOS CURVOS
4. ELIPSE
5. LINEA RECTA
6. LINEA RECTA VERTICAL/HORIZONTAL
7. POLILINEA
8. POLIGONO
9. TEXTO
10. BITMAP

Esta herramienta se utiliza para importar dibujos de ficheros de imagen (jpg, jpeg, bmp, pcx, tga) o bien del portapapeles de *WINDOWS*. Para llamar un objeto bitmap, utilice esta herramienta y forme un rectángulo. Una vez creado el rectángulo, podrá importar en su interior una imagen utilizando *EDIT -> Import image*, o bien pegando la imagen del portapapeles (*Edit -> Paste Bitmap*).

La función *Edit -> Bitmap Original Size* se utiliza para modificar el tamaño de la imagen al original con el que fue creado o a otro distinto

La función *Edit -> Edit Bitmap*, permite editar la imagen importada sin modificar el fichero de imagen original.

11. TENDENCIAS EN TIEMPO REAL
12. TENDENCIAS HISTÓRICAS
13. PAGINA DE ALARMAS
14. PULSADORES

Los Elementos Wizards

WIZARDS, en su más básico concepto, podría ser definido como "elementos inteligentes" que permiten que las aplicaciones **InTouch** puedan ser generadas de un modo más rápido y eficiente. La versión 7.1 de **InTouch** dispone de los elementos **WIZARDS** que permiten crear rápidamente un objeto en la pantalla. Haciendo doble click sobre el objeto podemos asociarle links (animación), asignarlo a tagnames o incluso incluir una lógica en ese objeto. Si agrupamos varios de estos objetos, podemos crear un elemento completo, acabado y programado, que lo podemos utilizar tantas veces como queramos. Bien, pues **WIZARDS** hace esto por Usted! Todo lo que tiene que hacer es seleccionar el **WIZARDS** que desee e **InTouch** se lo dibujará, animará y programará.

Por ejemplo, un amperímetro: **WIZARDS** le dibujará el elemento en la pantalla y cuando haga doble clic sobre él sólo necesitará rellenar los campos que se le indican. Esta configuración incluye el tagname sobre el que situar el amperímetro, valores máximo y mínimo de lectura, colores, divisiones, etc. Una vez la información ha sido introducida, el **WIZARDS** amperímetro ya puede utilizarse como tal

Además de estos **WIZARDS** "sencillos", es posible utilizar otros más "complejos" que provoquen operaciones en background, tales como crear/convertir una base de datos, importar un fichero AutoCad, configurar módulos de software (p.e. recetas, SPC), etc. Ello es posible gracias a la herramienta Wonderware Extensibility Toolkit (opcional de **InTouch**). La mayoría de **WIZARDS** son escalables y configurables en tamaño. Ello le permitirá modificar y poder ajustar los dibujos ya hechos a un tamaño necesario para su ventana.

WIZARDS son accesibles desde la caja de herramientas del **WINDOWMAKER**. Pero además, es posible incorporar un **WIZARDS** concreto (o más de uno) a la caja de herramientas, para que este aparezca en ella y sea muy sencillo seleccionarlo.



EJERCICIO 2. CREACION DE UNA VENTANA

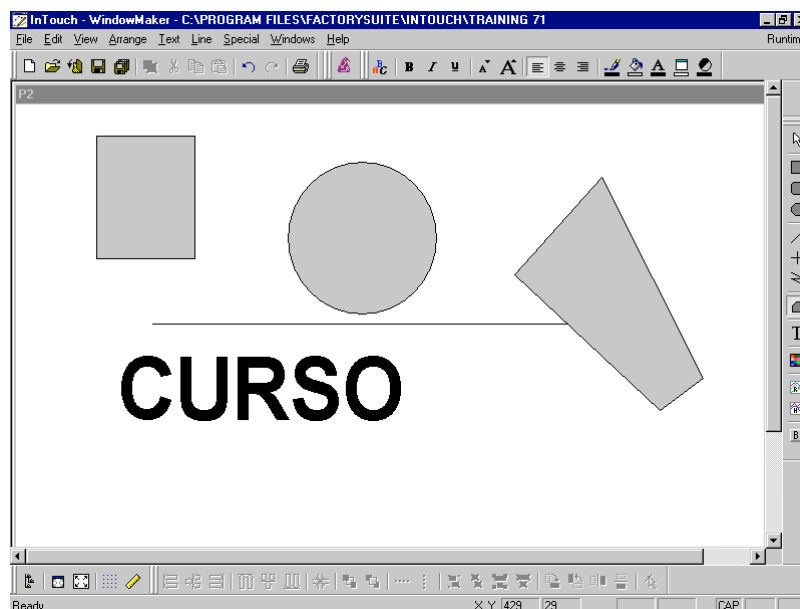
- Para la creación de una ventana, seleccione *NEW WINDOW* desde el menú *FILE* y rellene los campos necesarios en el cuadro de diálogo. El nombre de la ventana en este *TRAINING* es **PANTALLA1**



Cree más de una pantalla. Compruebe los distintos efectos en función de Title Bar, Size Controls, tamaño, color, etc.



EJERCICIO 3. MANEJO DE OBJETOS DE DIBUJO



Utilice este ejercicio para familiarizarse con los objetos de dibujo de **InTouch**. Para ello, compruebe el uso de cada uno de ellos, maneje técnicas de edición tales como Importar Imágenes, cree símbolos y celdas, ponga en línea distintos objetos, aplique color sobre ellos, maneje texto con distintas fuentes, etc.

MODULO 3. DICCIONARIO DE TAGNAMES

Sección 1. Definición del Tagname

El diccionario de tagnames es el corazón de **InTouch**. Durante el runtime, este diccionario contiene todos los valores de los elementos en la base de datos. Para crear esa base de datos, **InTouch** necesita saber qué elementos la van a componer. Debemos, por lo tanto, crear una base de datos con todos aquellos datos que necesitemos para nuestra aplicación. A cada uno de estos datos (tags) debemos asignarle un nombre. Al final, dispondremos de un diccionario con todos los tagnames o datos que nosotros mismos hemos creado.

Acceso

A este diccionario se accede desde el menú */Special/TagName Dictionary*.

Definición de los Tagnames

Desde el diccionario de tagnames definimos los tagnames y sus características. Existen diversos tipos de tagnames, según su función o características. Básicamente se dividen en:

<i>MEMORY</i>	Tags registros internos de InTouch
<i>I/O</i>	Registros de enlace con otros programas
<i>INDIRECT</i>	Tags de tipo indirecto
<i>GROUP VAR</i>	Tags de los grupos de alarmas
<i>HISTTREND</i>	Tag asociado a los gráficos históricos
<i>TagID</i>	Información acerca de los tags que están siendo visualizados en una gráfica histórica

De los 3 primeros tipos, disponemos de:

<i>Discrete</i>	Puede disponer de un valor 0 ó 1
<i>Integer</i>	Tagname de 32 bits con signo. Su valor va desde -2.147.483.648 hasta 2.147.483.647
<i>Real</i>	Tagname en coma flotante. Su valor va entre $\pm 3.4e^{38}$. Todos los cálculos son hechos en 64 bits de resolución, pero el resultado se almacena en 32 bits
<i>Message</i>	Tagname alfanumérico de hasta 131 caracteres de longitud

Nota: Existe otro tipo de tags, los System Tagnames. Se trata de tags del sistema, que nos dan información acerca de parámetros tales como fecha/hora, errores de impresora, actividad del Historical Logging, etc. Todos estos tags empiezan por el signo \$

Definición de las Características

- A) Main
- B) Details
- C) Alarms
- D) Details&Alarms
- E) Members

<i>Main</i>	Visualiza las características principales del tagname
<i>Details</i>	Visualiza las características del tag que va a crear (valor mínimo/máximo, etc.)
<i>Alarms</i>	Visualiza las condiciones de alarma del tag
<i>Details&Alarms</i>	Le permitirá de visualizar las características del tagname tanto de detalles como de alarma
<i>Members</i>	Visualiza Miembros caso de ser supertag

Una vez seleccionado el tipo de tagname y qué características debemos definir, un submenú aparecerá para que rellenemos los campos de ese tagname.

Campos a Rellenar del Tagname

<input type="radio"/> Read only <input checked="" type="radio"/> Read Write	Permite lectura/escritura o sólo lectura del registro
<input checked="" type="checkbox"/> Log Data	Graba el valor del tag al fichero de históricos cuando varía más que lo especificado en <i>Log Deadband</i>
<input type="checkbox"/> Log Events	Activa la grabación de eventos para ese tag
<input type="checkbox"/> Retentive Value	Permite que el valor current del registro sea retentivo
<input checked="" type="checkbox"/> Retentive Parameters	Permite retener los cambios del registro de cualquier campo de límites de alarmas
Initial Value: 20	Selecciona el valor inicial del registro
Min EU: -10	Introduzca el valor en unidades de ingeniería del registro equivalente al mínimo recibido

Max EU: <input type="text" value="150"/>	Introduzca el valor en unidades de ingeniería del registro equivalente al máximo recibido
Deadband: <input type="text" value="0"/>	Permite definir cuánto debe cambiar el valor de un registro para ser actualizado en pantallas
Min Raw: <input type="text" value="-32768"/>	Valor mínimo en el rango de valores enteros del valor I/O
Max Raw: <input type="text" value="32767"/>	Valor máximo en el rango de valores enteros del valor I/O
<input type="text" value="Access Name: ..."/>	Seleccione el programa de acceso
Conversion <input checked="" type="radio"/> Linear <input type="radio"/> Square Root	Seleccione si quiere una conversión lineal o de raíz cuadrada
<input checked="" type="checkbox"/> Use TagName as Item Name	Seleccione esta opción para displayar el tagname como nombre del item I/O
Log Deadband: <input type="text" value="3"/>	Permite definir cuánto debe cambiar el valor de un registro para ser grabado en el fichero



EJERCICIO 4. CREACION DE TAGNAMES

- Dentro de nuestro ejemplo en el training vamos a crear 2 tagnames :

TEMPERATURA	MEMORY INTEGER de lectura/escritura entre 0 y 100
MOTOR	MEMORY DISCRETE de lectura/escritura

Para ello, dirjase a SPECIAL → TAGNAME DICTIONARY, y pulse sobre NEW

The screenshot shows the 'Tagname Dictionary' dialog box with the 'Details' tab selected. The 'Tagname' field contains 'TEMPERATURA', and the 'Type' is set to 'Memory Integer'. The 'Group' is '\$System', and the 'Comment' is 'Temperatura Horno 1'. The 'Initial Value' is 0, 'Min Value' is 0, and 'Max Value' is 100. The 'Eng Units', 'Deadband', and 'Log Deadband' fields are empty. The 'Read/Write' radio buttons are set to 'Read/Write'.

No olvide los rangos de 0 a 100 de Temperatura.
Pulse SAVE cada vez que cree un tagname

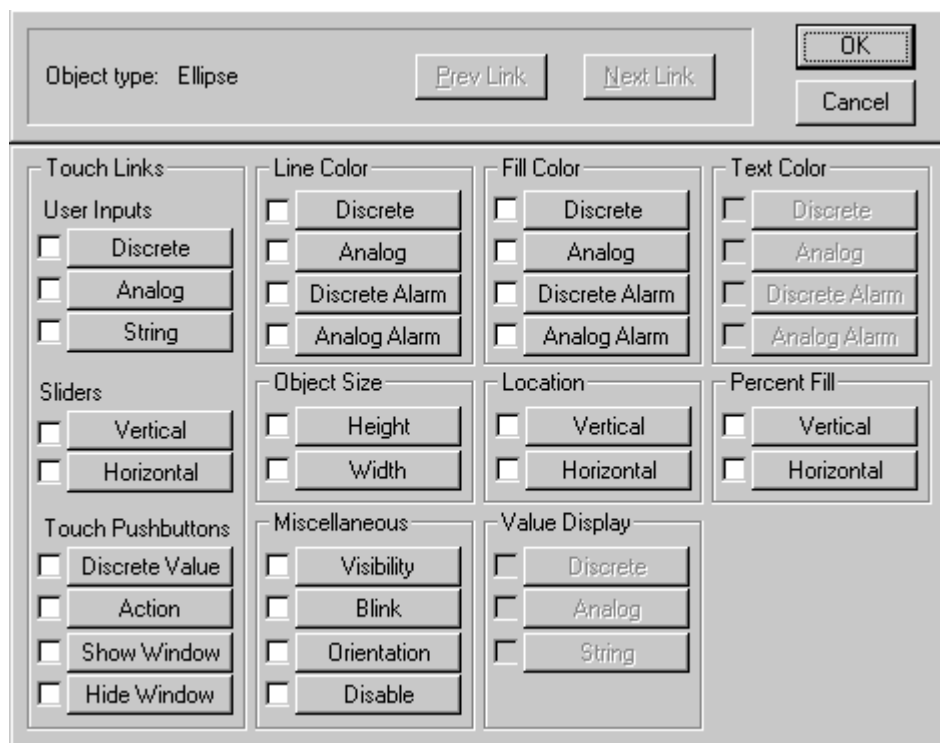
The screenshot shows the 'Tagname Dictionary' dialog box with the 'Details' tab selected. The 'Tagname' field contains 'MOTOR', and the 'Type' is set to 'Memory Discrete'. The 'Group' is '\$System', and the 'Comment' is 'MOTOR 1'. The 'Initial Value' is set to 'Off'. The 'On Msg' and 'Off Msg' fields are empty. The 'Read/Write' radio buttons are set to 'Read/Write'.

MODULO 4. ANIMATION LINKS

Tras haber creado un objeto gráfico o un símbolo, éste puede ser animado mediante las *Animation Links*. Las *Animation Links* provocan que el objeto cambie de apariencia reflejando cambios en los valores de la base de datos. Por ejemplo, una válvula puede cambiar de color de acuerdo a si está o no activa.

Sección 1. ¿Qué son las Animation Links?

Para asignar una *animation link* a un objeto, éste deberá estar seleccionado. Haciendo dos veces click sobre el objeto o símbolo deseado entramos directamente en el menú de *Animation Links*.



Una vez hecho esto, podremos seleccionar el tipo de animación que queremos asociar a ese objeto. Podemos incluso asociar varios *Animation Links* a un mismo objeto o símbolo.

Sección 2. Animando Objetos

Cada una de las funciones de *Animation Links* dispone de un submenú que deberemos rellenar. Se trata de las características propias de cada comando de animación. Los comandos de animación son los siguientes:

1. USER INPUTS

- Touch Links -
Value Input

Discrete

Analog

String

Al pulsar sobre un *USER INPUT*, InTouch nos pregunta el nuevo valor según si es:

DISCRETE: Modificación valor 0/1

ANALOG: Modificación valor analógico

STRING: Modificación cadena alfanumérica

Input -> Discrete Tagname

Tagname: MOTOR_ARRANQUE

Key equivalent

Ctrl Shift F1

Msg to User:

Set Prompt: On On Message: On

Reset Prompt: Off Off Message: Off

Input Only

OK Cancel Clear

Permite modificar un valor discreto. Al pulsar, nos aparecerá una ventana para que modifiquemos el valor a 0 ó 1. Podemos modificar en el menú los mensajes que le deben aparecer al operador.

Input -> Analog Tagname

Tagname: TEMPERATURA

Key equivalent

Ctrl Shift F2

Msg to User:

Keypad? Yes No

Min Value: 0

Max Value: 100

Input Only

OK Cancel Clear

Permite modificar un valor analógico (ya sea *Memory Type* o *I/O Type*). Podemos dar una mensaje al operador, así como limitar los valores mínimo y máximo de entrada. La función *KEYPAD?* posibilita que la entrada se lleve a cabo desde un teclado externo *PC* compatible.

Permite modificar una cadena alfanumérica. Para ello, un completo teclado *QWERTY* aparece en pantalla. Se utiliza para passwords, selecciones de datos, etc. Podemos dar un mensaje al operador. La función *ECHO CHARACTERS?* permitirá o no que los valores que se van introduciendo sean a la vez displayados en pantalla. La función *KEYPAD?* posibilita que la entrada se lleve a cabo desde un teclado externo *PC* compatible

2. VALUE SLIDER

Permite crear una barra de desplazamiento vertical u horizontal para seleccionar valores

3. TOUCH PUSHBUTTON

DISCRETE: A diferencia del *touch value*, actúa como un pulsador, directamente sobre una señal 0/1.

ACTION: Permite ejecutar una lógica o acciones (llamada a otros programas, impresión, etc).

SHOW/HIDE WINDOW: Permite llamar a otras pantallas o hacerlas desaparecer del monitor

Creamos un pulsador que colocará a 1 o a 0 el registro seleccionado en *tagname*. Podemos asignar además una tecla o combinación de teclas que hagan la misma función que este pulsador.

El pulsador creado puede ser del siguiente tipo:

<i>DIRECT</i>	Pone el valor a 1 al pulsar el botón y mantenerlo. Al soltarlo lo pone a 0
<i>REVERSE</i>	Pone el valor a 0 al pulsar el botón y mantenerlo. Al soltarlo lo pone a 1
<i>TOGGLE</i>	Invierte el estado del bit seleccionado al ser pulsado
<i>RESET</i>	Pone el valor a 0 al ser pulsado
<i>SET</i>	Pone el valor a 1 al ser pulsado

4. LINE COLOR

- Line Color**
- Discrete
 - Analog
 - Discrete Alarm
 - Analog Alarm

Permiten animar el color de línea de un objeto. Este cambio de color puede depender de un valor discreto/analógico o asociarse a una alarma de tipo discreto o analógico

Line Color -> Discrete Expression

Expression:

Colors
 0,FALSE,Off:
 1,TRUE,On:

OK
 Cancel
 Clear

Line Color -> Analog Expression

Expression:

BreakPoints:

Color:

OK
 Cancel
 Clear

5. FILL COLOR

- Fill Color**
- Discrete
 - Analog
 - Discrete Alarm
 - Analog Alarm

Permiten rellenar un objeto de un color. Este cambio de color puede depender de un valor discreto/analógico o asociarse a una alarma de tipo discreto o analógico

6. TEXT COLOR

- Text Color**
- Discrete
 - Analog
 - Discrete Alarm
 - Analog Alarm

Permiten cambiar el color de un texto. Este cambio de color puede depender de un valor discreto/analógico o asociarse a una alarma de tipo discreto o analógico

7. OBJECT SIZE

- Object Size**
- Height
 - Width

Permite asociar el tamaño vertical/horizontal de un objeto a un registro

Object Height -> Analog Value

Expression:

Properties

Value at Max Height: Max % Height:
 Value at Min Height: Min % Height:

Anchor

Top Middle Bottom

OK
Cancel
Clear

Podemos definir altura (anchura) mínima y máxima tanto real como porcentual. Definimos también cuál es el punto de partida de ese movimiento.

8. MISCELLANEOUS

Miscellaneous	
<input type="checkbox"/>	Visibility
<input type="checkbox"/>	Blink
<input type="checkbox"/>	Orientation
<input type="checkbox"/>	Disable

Visibility: Permite que un objeto aparezca/desaparezca de la pantalla
 Blink: Intermitencia del objeto
 Orientation: Modifica orientación del objeto
 Disable: Hace que un objeto "táctil" deje de serlo

Object Visibility -> Discrete Value

Expression:

Visible State

On Off

OK
Cancel
Clear

El objeto seleccionado sólo se visualizará cuando el registro *TEMPERATURA* sea mayor que 75.

Object Blinking -> Discrete Value

Expression - Blink When:

Blinked Attributes

Blink Invisible
 Blink visible with these attributes:

Text Color:

Line Color:

Fill Color:

Blink Speed

Slow Medium Fast

OK
Cancel
Clear

El objeto seleccionado se hará intermitente al activarse una alarma.

Orientation -> Analog Value

Expression:

Properties

Value at Max CCW: <input type="text" value="0"/>	CCW Rotation: <input type="text" value="0"/>	<input type="button" value="Clear"/>
Value at Max CW: <input type="text" value="100"/>	CW Rotation: <input type="text" value="360"/>	

Center of Rotation Offset from Object Centerpoint

X Position: <input type="text" value="0"/>	Y Position: <input type="text" value="0"/>
---	---

El objeto seleccionado rotará de acuerdo al valor del tagname *flecha*.

9. LOCATION

Location

<input type="checkbox"/> <input type="button" value="Vertical"/>	Permite modificar la posición del objeto
<input type="checkbox"/> <input type="button" value="Horizontal"/>	

Vertical Location

Expression:

Properties

	Value		Vertical Movement
At Top:	<input type="text" value="0"/>	Up:	<input type="text" value="0"/>
At Bottom:	<input type="text" value="100"/>	Down:	<input type="text" value="100"/>

10. VALUE DISPLAY

Value Display

<input type="checkbox"/> <input type="button" value="Discrete"/>	Se utiliza para visualizar un valor discreto, analógico o alfanumérico
<input type="checkbox"/> <input type="button" value="Analog"/>	
<input type="checkbox"/> <input type="button" value="String"/>	

11. PERCENT FILL

Percent Fill

Vertical

Horizontal

Permite asociar un registro tagname a una barra gráfica.

Vertical Fill -> Analog Value

Expression:
DEPOSITO

Properties

Value at Max Fill: 100 Max % Fill: 100

Value at Min Fill: 0 Min % Fill: 0

Direction

Up Down

Background Color:

OK
Cancel
Clear

Sección 3. Edición de Links y de Tags***Sustituir Texto***

Desde *SPECIAL* → *SUBSTITUTE STRINGS* es posible modificar el texto que tengamos seleccionado

Substitute Strings ... 1 of 1

Current String: New String:

TEMPERAURA TEMPERATURA

OK Cancel Replace

Sustituir Tagnames

Cuando duplicamos un objeto, este se convierte en una exacta réplica del duplicado. Si necesitamos usar un tagname distinto para el nuevo objeto, podemos utilizar *SPECIAL* → *SUBSTITUTE TAGS*

Substitute Tagnames... 1 of 1

Current Name: Required Type New Name:

TEMPERATURA Analog TEMPERATURA

OK Cancel Index Convert Replace

Importar y Exportar Ventanas

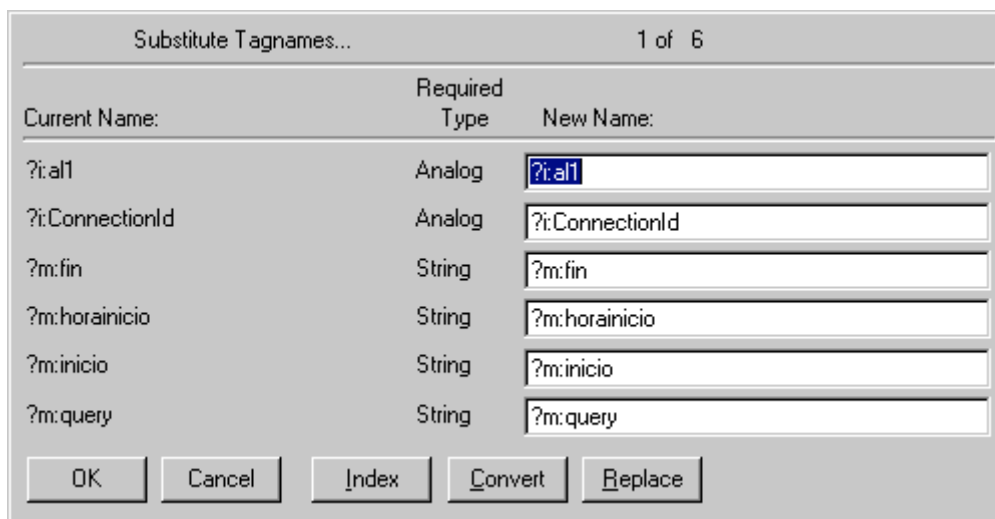
Es posible Importar/exportar ventanas de/desde otra aplicación **InTouch** mediante *FILE → IMPORT WINDOW*

Convertir Placeholder Tagnames

Al importar ventanas de otra aplicación, **InTouch** nos da la opción de mantener o no los tags de la otra aplicación. Podemos convertirlos a la nuestra mediante *SPECIAL → SUBSTITUTE TAGS*

InTouch añade al nombre del tagname un "placeholder":

- ?d: Para tags discretos
- ?i: Para tags enteros
- ?r: Para tags reales
- ?m: Para tags de Mensaje



Convert nos aceptaría los nombres de los nuevos tagnames (eliminando la porción ?m), para que pudiéramos crearlos como nuevos

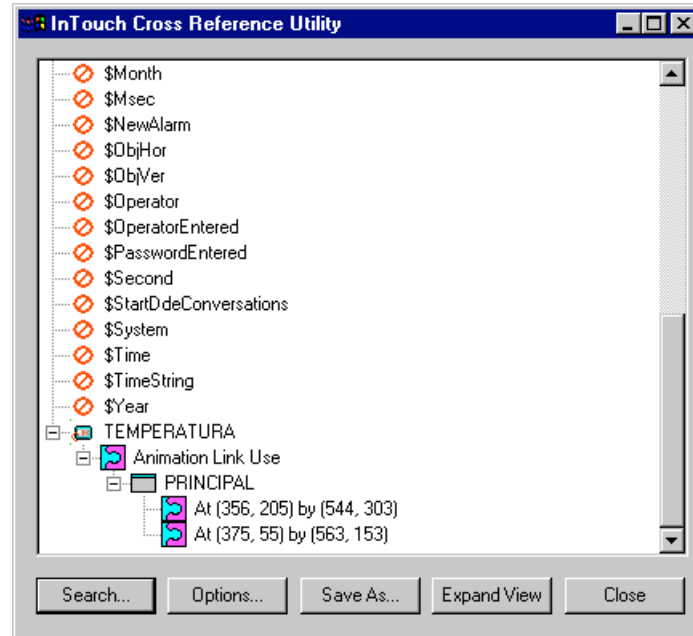
Borrar Tagnames

Es posible borrar tagnames de **InTouch** bajo las siguientes condiciones:

1. El tag a borrar no puede estar utilizado.
2. WindowViewer debe estar cerrado
3. En WindowMaker no debe tener abierta ninguna ventana
4. Hay que ejecutar *SPECIAL → UPDATE USE COUNTS*
5. Ejecutar *SPECIAL → DELETE UNUSED TAGS*
6. Volver a ejecutar *SPECIAL → UPDATE USE COUNTS*

Referencias Cruzadas de InTouch

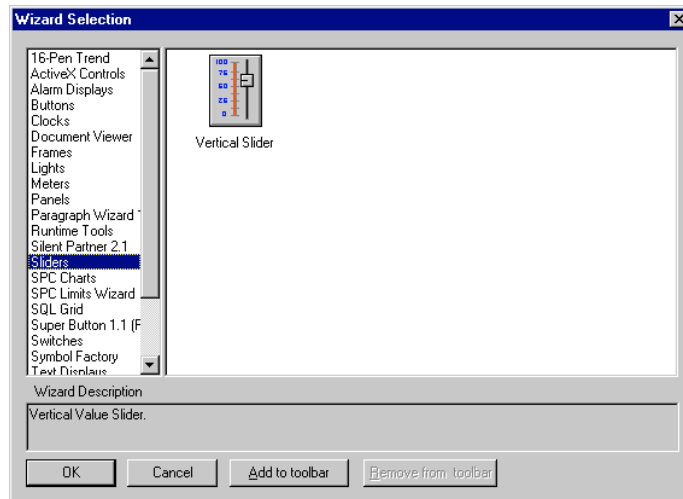
Es posible generar un listado de referencias cruzadas de tags de **InTouch** muy completo mediante la utilidad **InTouch Cross Reference** (accesible desde *SPECIAL*→*CROSS REFERENCE*)



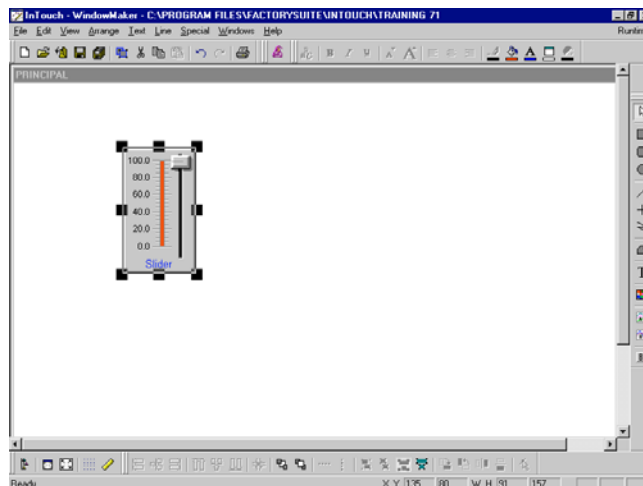


EJERCICIO 5.ANIMACION CON WIZARDS

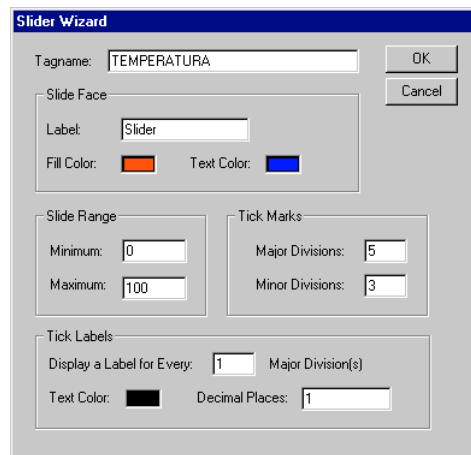
- Para nuestro proceso de animación, generaremos una pantalla con 4 WIZARDS. Primero vamos a crear una barra de deslizamiento (slider) para la introducción de datos. Utilizaremos un WIZARD ya creado para que compruebe la facilidad en su manejo (lo encontrará bajo la categoría *SLIDERS* en Wizards)



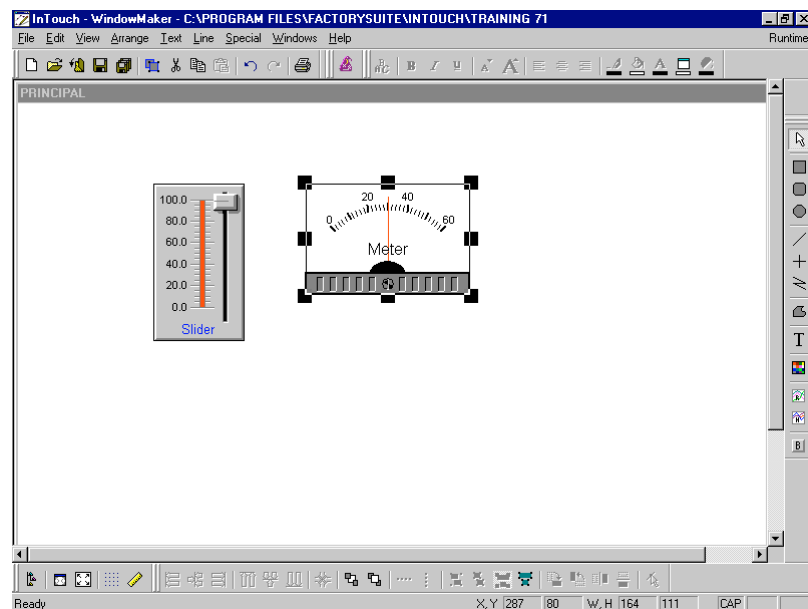
Colóquelo en pantalla donde desee:



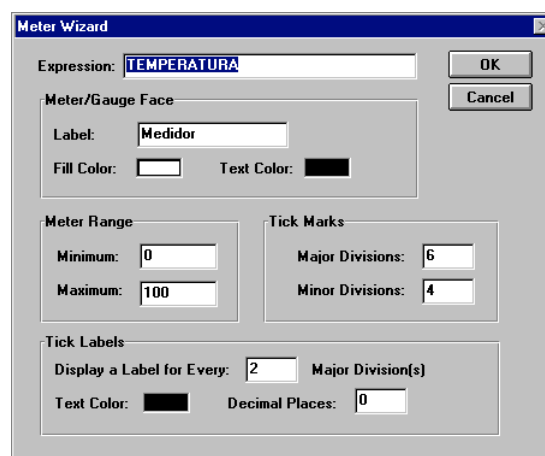
Haga doble clic sobre la barra de desplazamiento para su programación



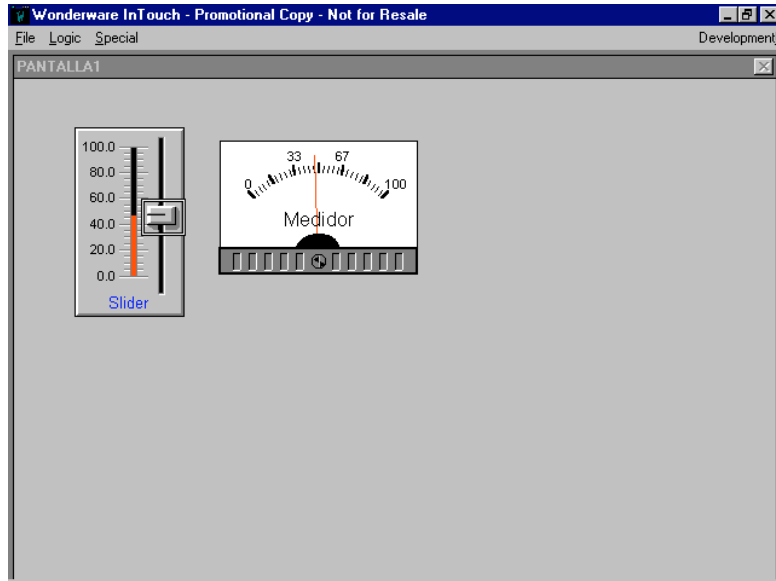
Crearemos un medidor para ver el valor que estamos actualizando. Utilizaremos para ello otro WIZARD (lo encontrará en *METERS*)



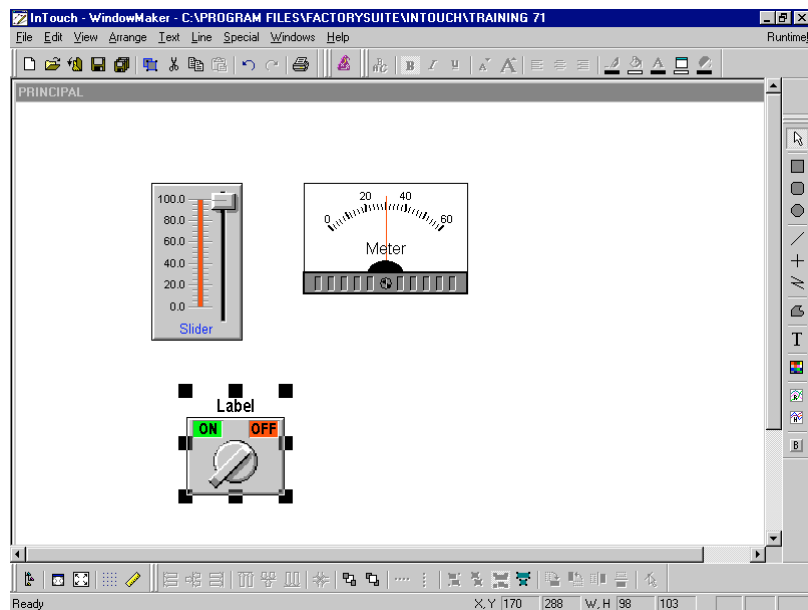
Haga doble clic sobre el WIZARD para su programación:



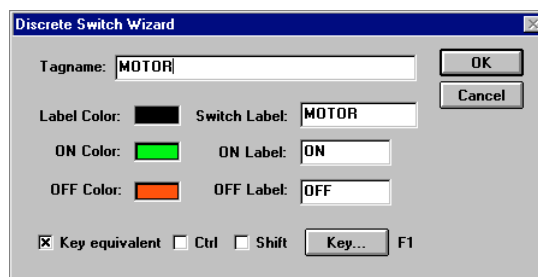
Podemos ir a *RUNTIME* para ver el efecto que tiene nuestro pequeño programa.



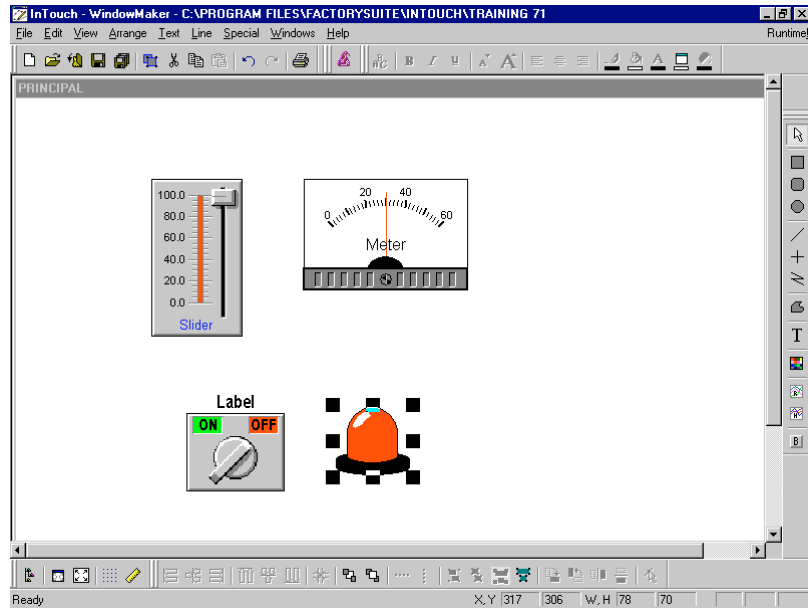
Vamos a utilizar ahora el tagname *MOTOR*. Simularemos un MARCHA/PARO de un motor utilizando un pulsador que encontraremos en la categoría *Switches* de los *WIZARDS*:



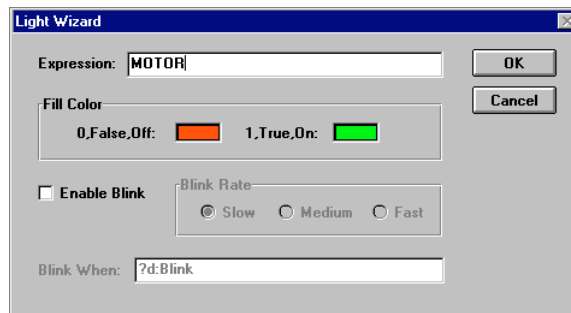
Haga doble clic sobre el WIZARD para su programación:



Vamos a crear una lámpara para ver el estado del motor. También para ello utilizaremos un *WIZARD* (bajo la categoría *Lights*).



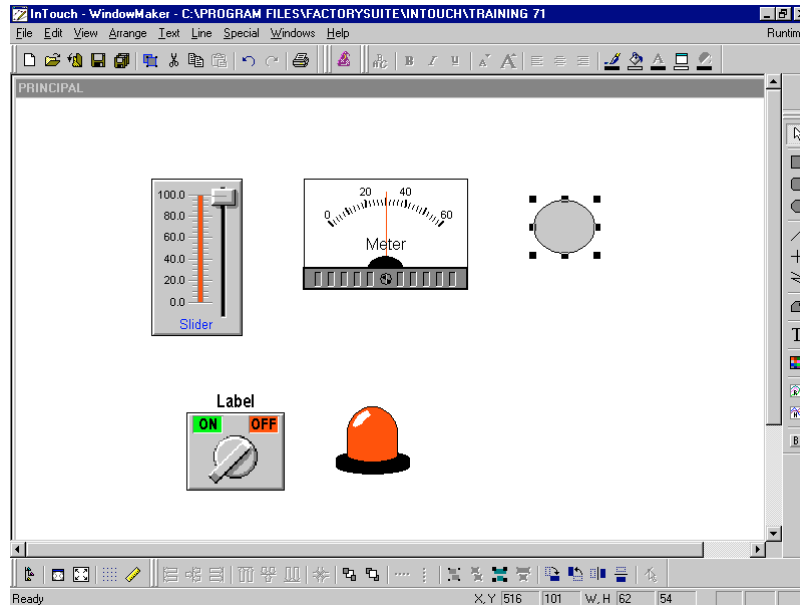
Haga doble clic sobre el WIZARD para su programación:



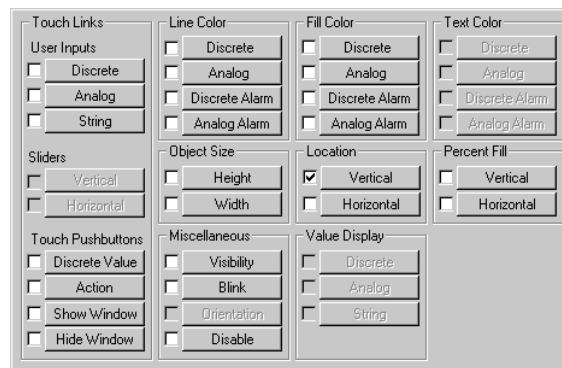


EJERCICIO 6. ANIMACION CON ANIMATION LINKS

- Vamos a hacer un ejemplo con un *ANIMATION LINK*. Para ello, cree en la pantalla un pequeño círculo o un objeto similar.



Haga doble clic sobre el círculo. Aparecerá el cuadro de *ANIMATION LINKS*:



Pulse sobre el botón sobre *VERTICAL*.

Defina en *EXPRESSION* el tagname *TEMPERATURA*. Ello nos permitirá ver cómo el círculo se mueve hacia arriba y abajo de acuerdo al valor de la temperatura.



Vertical Location

Expression:
TEMPERATURA

Properties

	Value		Vertical Movement
At Top:	0	Up:	0
At Bottom:	100	Down:	200

OK
Cancel
Clear

MODULO 5. InTouch QUICKSCRIPTS

Sección 1. Tipos de Scripts

InTouch permite crear una lógica interna con condiciones, cálculos, etc. Esta lógica puede estar asociada a:

- 1.- Toda una aplicación (*APPLICATION SCRIPTS*)
- 2.- Una sola ventana (*WINDOW SCRIPTS*)
- 3.- Una tecla (*KEY SCRIPTS*)
- 4.- Una condición (*CONDITION SCRIPTS*)
- 5.- Cambio de un dato (*DATA CHANGE SCRIPTS*)
- 6.- Asociadas a un ActiveX
- 7.- Funciones Usuario (*QUICKFUNCTIONS*)

Sección 2. Funciones Scripts

La lógica (*SCRIPT*) de InTouch es un programa que nos permitirá llevar a cabo acciones determinadas mediante una estructura *IF...THEN...ELSE*

La lógica estará activa de acuerdo al tipo de lógica elegida (por aplicación, por ventana, etc.). Básicamente, los 6 tipos distintos de lógica actúan del mismo modo.

1. FUNCION

Se trata de funciones internas específicas del sistema que pueden ser utilizadas en la lógica. Estas funciones son las siguientes:

A) FUNCIONES DE TEXTO

<i>Dtext()</i>	Permite cambiar dinámicamente un tagname de mensaje según un tag discreto
<i>StringASCII()</i>	Devuelve el valor ASCII del primer carácter de un mensaje
<i>StringChar()</i>	Devuelve el carácter de un código ASCII específico
<i>StringFromIntg()</i>	Convierte un valor entero en su representación ASCII
<i>StringFromReal()</i>	Convierte un valor real en su representación ASCII
<i>StringFromTime()</i>	Convierte la hora (en segundos desde el 1/1/70) en un mensaje
<i>StringInString()</i>	Devuelve la posición en la que se encuentra parte de un texto
<i>StringLeft()</i>	Devuelve el primer (o el más a la izquierda) carácter de un mensaje
<i>StringLen()</i>	Devuelve la longitud de un mensaje
<i>StringLower()</i>	Convierte caracteres mayúsculas en minúsculas
<i>StringMid()</i>	Devuelve un número específico de caracteres de un mensaje, empezando por una posición determinada
<i>StringReplace()</i>	Reemplaza o cambia partes específicas de un mensaje
<i>StringRight()</i>	Devuelve el último (o el más a la derecha) carácter de un mensaje
<i>StringSpace()</i>	Genera una cadena de espacios
<i>StringTest()</i>	Testea el primer carácter de una cadena para determinar de qué tipo se trata
<i>StringToIntg()</i>	Convierte el valor numérico de un mensaje a un valor entero sobre el que podemos aplicar cálculo
<i>StringToReal()</i>	Convierte el valor real de un mensaje a un valor entero sobre el que podemos aplicar cálculo
<i>StringTrim()</i>	Elimina espacios no requeridos de un mensaje
<i>StringUpper()</i>	Convierte caracteres minúsculas en mayúsculas

<i>Text()</i>	Provoca que un tag de mensaje devuelva el valor de uno analógico
---------------	--

B) FUNCIONES MATEMATICAS

<i>Abs()</i>	Devuelve un valor absoluto
<i>ArcCos()</i>	Función de arcocoseno
<i>ArcSin()</i>	Función de arcoseno
<i>ArcTan()</i>	Función de arcotangente
<i>Cos()</i>	Coseno
<i>Exp()</i>	Devuelve el resultado de e elevado a una potencia
<i>Int()</i>	Devuelve el siguiente valor entero menor que o igual a una número específico
<i>Log()</i>	Devuelve el logaritmo en base 10 de un número
<i>LogN()</i>	Devuelve los valores de un logaritmo de x en base n
<i>Pi()</i>	Devuelve el valor de pi
<i>Round()</i>	Redondea un número real
<i>Sgn()</i>	Determina el signo de un valor
<i>Sin()</i>	Devuelve el seno de un valor
<i>Sqrt()</i>	Calculo de la raíz cuadrada
<i>Tan()</i>	Devuelve la tangente de un valor
<i>Trunc()</i>	Corta un número real eliminando la porción derecha del punto decimal

C) FUNCIONES DEL SISTEMA

<i>ActivateApp()</i>	Activa otra aplicación de Windows que ya está rodando
<i>FileCopy()</i>	Copia de un fichero a otro
<i>FileDelete()</i>	Borrado de un fichero
<i>FileMove()</i>	Mueve un fichero
<i>FileReadFields()</i>	Lee datos de un fichero con formato CSV
<i>FileReadMessage()</i>	Lee un número específico de bytes de un fichero
<i>FileWriteFields()</i>	Graba datos a un fichero con formato CSV
<i>FileWriteMessage()</i>	Graba un número específico de bytes a un fichero
<i>InfoAppActive</i>	Testea si una aplicación está activa
<i>InfoAppTitle()</i>	Devuelve el título de un programa que está corriendo en Windows
<i>InfoDisk()</i>	devuelve información sobre un disco local (o de red)
<i>InfoFile()</i>	Devuelve información sobre un fichero específico o sobre un subdirectorio
<i>InfoINTOUCHAppDir()</i>	Informa sobre cuál es el subdirectorio de la aplicación actual de InTouch
<i>InfoResources()</i>	Devuelve diversos valores de recursos del sistema
<i>IsAnyAsynchFunctionBusy</i>	Informa si alguna QuickFunctions asíncrona se está ejecutando
<i>StartApp</i>	Arranca automáticamente otra aplicación de Windows

D) FUNCIONES VARIAS (MISC)

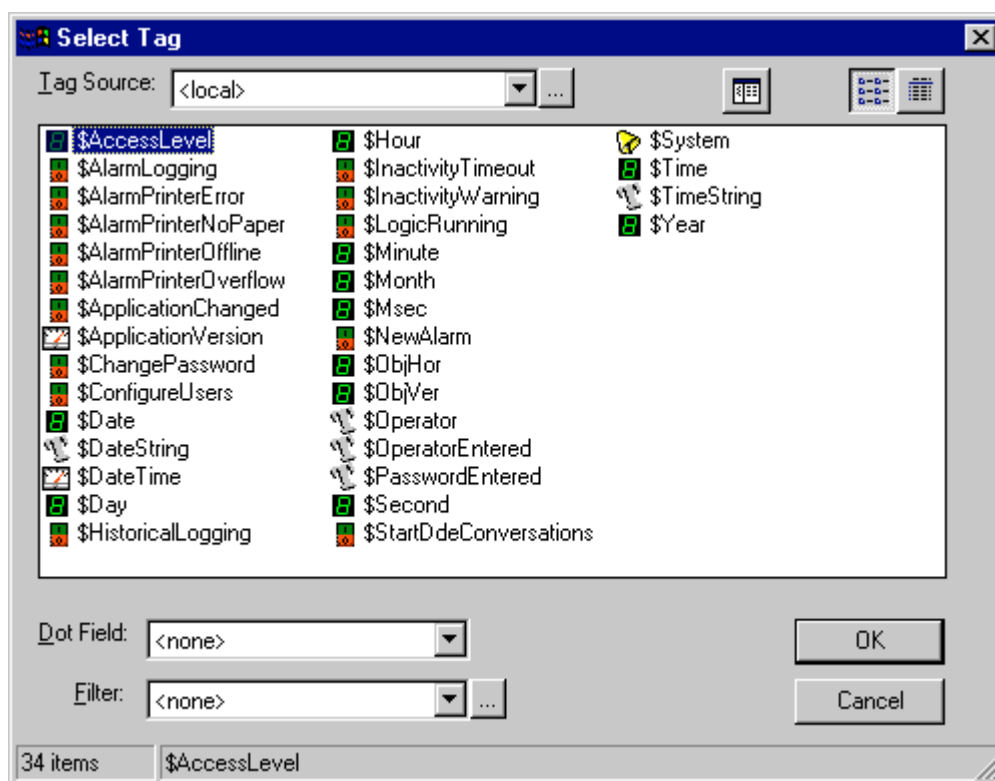
<i>Ack</i>	Reconoce alarmas locales de InTouch
<i>almAckAll()</i>	Reconoce todas las alarmas de la cola, incluso aquellas que no están en el visor
<i>almAckDisplay()</i>	Reconoce únicamente aquellas alarmas visibles en el visor
<i>almAckRecent()</i>	Reconoce las alarmas más recientes
<i>almAckSelect()</i>	Reconoce las alarmas seleccionadas en el visor de alarmas
<i>almDefQuery()</i>	Lleva a cabo una cola para actualizar una visor de alarmas con las propiedades por defecto
<i>AlmMoveWindow()</i>	Permite hacer un scroll de la ventana de alarmas
<i>almQuery()</i>	Lleva a cabo una cola para actualizar una visor de alarmas
<i>almSelectAll()</i>	Selección o no de todas las alarmas del visor
<i>almSelectItem()</i>	Selección o no la alarma iluminada del visor
<i>almShowStats()</i>	Visualiza la pantalla de estadísticas de alarmas
<i>ChangePassword()</i>	Visualiza el cuadro de diálogo para modificar un código secreto (password)
<i>DialogStringEntry()</i>	Visualiza un teclado alfanumérico, permitiendo al usuario cambiar el valor actual de una cadena de un tag de mensaje
<i>DialogValueEntry()</i>	Visualiza un teclado numérico, permitiendo al usuario cambiar el valor

	actual de un tag discreto, entero o real
<i>GetNodeName()</i>	Devuelve el nombre del nodo NetDDE a una variable de mensaje
<i>GetPropertyD()</i>	Recupera la propiedad específica de un valor discreto
<i>GetPropertyI()</i>	Recupera la propiedad específica de un valor entero
<i>GetPropertyM()</i>	Recupera la propiedad específica de un valor de mensaje
<i>Hide</i>	Permite esconder una o más ventanas
<i>HideSelf</i>	Permite esconder la ventana actual
<i>HTGetLastError()</i>	Determina si se produjo un error al recoger datos de un lápiz en un histórico
<i>HTGetPenName()</i>	Devuelve el tagname del tag actualmente utilizado para un lápiz específico para una tendencia específico
<i>HTGetTimeAtScooter()</i>	Devuelve la hora en segundos desde las 00 :00 :00 horas GMT desde el 1/1/70 para la muestra en el puntero de la posición especificada en ScootNum y ScootLoc
<i>HTGetTimeStringAtScooter()</i>	Devuelve una cadena que contiene la fecha y hora para la muestra en la posición especificada en ScootNum y ScootLoc
<i>HTGetValue()</i>	Devuelve un valor del tipo requerido para un lápiz de la curva
<i>HTGetValueAtScooter()</i>	Devuelve un valor del tipo requerido para un lápiz de la curva en una posición determinada
<i>HTGetValueAtZone()</i>	Devuelve un valor del tipo requerido para un lápiz de la curva en una posición entre dos punteros
<i>HTScrollLeft()</i>	Retrasa la hora de inicio de la tendencia
<i>HTScrollRight()</i>	Adelanta la hora de inicio de la tendencia
<i>HTSelectTag</i>	Muestra el cuadro de diálogo de seleccionar Tag para un gráfico histórico
<i>HTSetPenName()</i>	Asigna un tagname diferente a un lápiz
<i>HTUpdateToCurrentTime()</i>	Recoge y displaya el valor con un tiempo final igual al actual
<i>HTZoomIn()</i>	Calcula un nuevo ancho y hora de inicio (menor)
<i>HTZoomOut()</i>	Calcula un nuevo ancho y hora de inicio (mayor)
<i>IOSetAccessName</i>	Modifica dinámicamente la aplicación/ tópico de un AccessName
<i>IOSetItem</i>	Modifica dinámicamente el item al que está asociado un tagname
<i>LogMessage()</i>	Escribe un texto predefinido al WWLogger
<i>PlaySound()</i>	Ejecuta un fichero de sonido .WAV
<i>PrintHT()</i>	Permite imprimir una tendencia gráfica de históricos. La tendencia debe encontrarse visible en la pantalla
<i>PrintWindow()</i>	Imprime la ventana especificada
<i>RestartWindowViewer</i>	Permite cerrar y reanunciar el WindowViewer
<i>SendKeys()</i>	Envía una tecla o secuencia de teclas a otra aplicación
<i>SetPropertyD()</i>	Especifica la propiedad de un valor discreto que debe ser escrito durante el runtime
<i>SetPropertyI()</i>	Especifica la propiedad de un valor entero que debe ser escrito durante el runtime
<i>SetPropertyM()</i>	Especifica la propiedad de un valor de mensaje que debe ser escrito durante el runtime
<i>Show</i>	Muestra una ventana específica
<i>ShowAt()</i>	Especifica los pixels horizontal y vertical de una ventana que debe mostrarse
<i>ShowHome</i>	Visualiza las "Home Windows". Estas son configuradas desde el menú <i>Special\Configure\</i> de InTouch
<i>ShowTopLeftAt()</i>	Especifica los pixels horizontal y vertical de la esquina superior izquierda de una ventana que debe mostrarse
<i>wcAddItem()</i>	Añade la cadena de mensaje especificada la List Box o a la Combo Box
<i>wcClear()</i>	Elimina todos los elementos de la List Box o Combo Box
<i>wcDeleteItem()</i>	Borra un elemento asociado de la List Box o Combo Box
<i>WcDeleteSelection()</i>	Borra el elemento actualmente seleccionado de la lista
<i>WcErrorMessage()</i>	Devuelve un texto que informa sobre el error
<i>wcFindItem()</i>	Determina el índice correspondiente del primer elemento de la List Box o Combo Box que equivale a un mensaje determinado
<i>wcGetItem()</i>	Devuelve el valor de la cadena asociada a un ItemIndex
<i>wcGetItemData()</i>	Determina el valor entero asociado
<i>wcInsertItem()</i>	Inserta un a cadena de caracteres en una lista
<i>wcLoadList()</i>	Cambia el contenido de la list o combo box con los elementos contenidos en el fichero <i>mensaje</i>
<i>wcLoadText()</i>	Cambia el contenido de la list o combo box con los elementos contenidos

	en el fichero <i>mensaje</i>
<i>wcSaveList()</i>	Cambia el contenido del fichero <i>mensaje</i> por los elementos especificados en una list o combo box
<i>wcSaveText()</i>	Cambia el contenido del fichero <i>mensaje</i> por los elementos especificados en una list o combo box
<i>wcSetItemData()</i>	Asigna un valor entero al elemento especificado de la lista especificado en <i>ItemIndex</i>
<i>WWControl()</i>	Permite al usuario controlar otra aplicación desde InTouch permitiéndole Restaurar, Minimizar, Maximizar o cerrar esa aplicación
<i>WWExecute()</i>	Envía un comando a una aplicación y tópico determinado
<i>WWPoke()</i>	Envía un valor a una aplicación, tópico y elemento determinado
<i>WWRequest</i>	Hace una petición de un valor para una aplicación, tópico y elemento

2. TAGNAME

Mediante este botón se accede a la lista completa de tagnames disponibles en nuestra aplicación, tanto los especiales internos como los creados durante la aplicación.



3. FIELD

Los tagnames de InTouch nos dan información por defecto del valor del tagname. Pero en realidad incluyen mucha más información. Accedemos a esta información a través de lo que denominamos Campos del tagname. Entre estos se incluyen, límites de alarmas, reconocimiento de alarmas, valores máx. y mín., etc.

<i>.Ack</i>	Reconocimiento de alarmas locales
<i>.Alarm</i>	Se activa cuando se activa la alarma de un tag específico
<i>.AlarmDevDeadband</i>	Monitoriza y controla la desviación de la banda muerta de las alarmas mayor y menor
<i>.AlarmEnabled</i>	Habilita/inhabilita alarmas y eventos locales
<i>.AlarmGroup</i>	Contiene la lista de un visor de alarmas distribuidas
<i>.AlarmValdeadband</i>	Monitoriza y controla el valor de una banda muerta de una alarma
<i>.Caption</i>	Determina qué mensaje debe visualizarse en una check box
<i>.ChartLength</i>	Controla la longitud del tiempo a visualizarse en las curvas históricas
<i>.ChartStart</i>	Controla la fecha/hora de inicio de una curva histórica
<i>.Comment</i>	Contiene el comentario del tagname
<i>.DevTarget</i>	Monitoriza y controla el valor base para las desviaciones mayor y menor de una alarma
<i>.DisplayMode</i>	Determina el método para visualizar curvas en una tendencia
<i>.Enabled</i>	Determina si el objeto de control puede responder a eventos generados por el usuario
<i>.HiHiLimit</i>	Monitoriza y controla el valor HiHi de una alarma
<i>.HiHiStatus</i>	Determina si existe una alarma tipo HiHi para el tag especificado
<i>.HiLimit</i>	Monitoriza y controla el valor Hi de una alarma
<i>.HiStatus</i>	Determina si existe una alarma tipo Hi para el tag especificado
<i>.ListCount</i>	Determina el número de elementos en una List Box o en una Combo Box

<i>.ListIndex</i>	Determina el índice (tagname o número) de un elemento seleccionado en la lista
<i>.LoLimit</i>	Monitoriza y controla el valor Lo de una alarma
<i>.LoLoLimit</i>	Monitoriza y controla el valor LoLo de una alarma
<i>.LoLoStatus</i>	Determina si existe una alarma tipo LoLo para el tag especificado
<i>.LoStatus</i>	Determina si existe una alarma tipo Lo para el tag especificado
<i>.MajorDevPct</i>	Monitoriza y controla el porcentaje de desviación de una alarma
<i>.MajorDevStatus</i>	Informa si existe una desviación mayor de una alarma determinada
<i>.MaxEU</i>	Valores máximos (en unidades de ingeniería) de un tagname específico
<i>.MaxRange</i>	Representa el porcentaje de un rango de una curva histórica que debe ser visualizado para cada tag de la curva
<i>.MinEU</i>	Valores mínimos (en unidades de ingeniería) de un tagname específico
<i>.MinorDevPct</i>	Monitoriza y controla el porcentaje de desviación de una alarma
<i>.MinorDevStatus</i>	Informa si existe una desviación menor de una alarma determinada
<i>.MinRange</i>	Representa el porcentaje de un rango de una curva histórica que debe ser visualizado para cada tag de la curva
<i>.Name</i>	Contiene el nombre del tag
<i>.NewIndex</i>	Devuelve el index del entero correspondiente del último elemento añadido a la List box o a la Combo Box mediante <i>wwAddItem()</i> o <i>wcInsertItem()</i>
<i>.NextPage</i>	Avanza hacia abajo una página el visor de alarmas
<i>.Normal</i>	Es igual a 1 cuando no hay alarma para el tagname específico
<i>.NumAlarms</i>	Contiene el número de alarmas de un objeto de alarmas
<i>.PageNum</i>	Contiene el número de página que se visualiza en el visor de alarmas
<i>.Pen1-.Pen8</i>	Controla el tagname que visualiza cada lápiz de las curvas históricas
<i>.PrevPage</i>	Avanza hacia arriba una página el visor de alarmas
<i>.PriForm</i>	Contiene el valor de prioridad más bajo usado en un visor de alarmas distribuidas para la cola de alarmas
<i>.PriTo</i>	Contiene el valor de prioridad más alto usado en un visor de alarmas distribuidas para la cola de alarmas
<i>.ProviderReq</i>	Contiene el número de servidores de alarmas requeridos por la cola actual
<i>.ProviderRet</i>	Contiene el número de servidores de alarmas que han devuelto con éxito su cola de alarmas
<i>.QueryState</i>	Contiene el filtro de la cola actual utilizado por un visor de alarmas distribuidas
<i>.QueryType</i>	Representa el tipo de colas de alarmas en un visor de alarmas
<i>.ReadOnly</i>	Determina si el contenido de un Text Box es sólo de lectura o de lectura/escritura
<i>.Reference</i>	permite al operador cambiar dinámicamente el <i>Access Name</i> y/o el <i>Item</i> durante el runtime
<i>.ReferenceComplete</i>	Devuelve una confirmación que el <i>Item</i> requerido es el mismo que el reflejado en el campo <i>.Value</i> .
<i>.ROCPct</i>	Monitoriza y/o controla el <i>Rate-of-Change</i> de un determinado tag
<i>.ROCStatus</i>	Monitoriza y/o controla si el <i>Rate-of-Change</i> existe para un determinado tag
<i>.ScooterLockLeft</i>	Si lo forzamos a 1, impedimos que el scooter de la derecha se mueva más a la izquierda que la posición del scooter de la izquierda
<i>.ScooterLockRight</i>	Si lo forzamos a 1, impedimos que el scooter de la izquierda se mueva más a la derecha que la posición del scooter de la derecha
<i>.ScooterPosLeft</i>	Monitoriza y/o controla la posición del scooter de la izquierda
<i>.ScooterPosRight</i>	Monitoriza y/o controla la posición del scooter de la derecha
<i>.Successful</i>	Contiene el estado de la última petición cola de alarmas distribuidas
<i>.TagID</i>	Se usa en conjunto con <i>.Pen1-.Pen8</i> , y sirve para monitorizar y/o controlar el tagname asignado a un lápiz de la curva histórica
<i>.TopIndex</i>	Determina el index del entero correspondiente del elemento superior de una List Box
<i>.TotalPages</i>	Contiene el número total de páginas de un objeto de alarmas
<i>.UpdateCount</i>	Se incrementa cada vez que se produce una actualización en la curva histórica
<i>.UpdateInProgress</i>	Igual a 1 mientras se están recogiendo datos para las curvas históricas
<i>.UpdateTrend</i>	Provoca que la curva histórica actualice sus valores
<i>.Value</i>	Contiene el valor de un tagname
<i>.Visible</i>	Determina si un control de ventanas se encuentra visible

USO DE VARIABLE INTERNAS

Es posible declarar variables internas mediante la instrucción *DIM*

DIM variablelocal [As tipo-dato]

Ej. DIM VarLoc1 As Integer;

VarLoc1 podrá utilizarse en todas las QuickScripts como variable de cálculo o de apoyo) no podrá ser visualizada en pantalla.

Esta variable no ocupa tag

Si no especificamos [As tipo-dato], por defecto será declarada como integer (entero)

MODULO 6. ALARMAS Y EVENTOS

Sección 1. Alarmas

Nota : **InTouch** permite la visualización de alarmas distribuidas (gestión de las alarmas bajo una estructura cliente/servidor en una red de ordenadores). Para este cursillo se da únicamente tratamiento a las alarmas locales.

InTouch soporta la visualización, archivo (en disco duro o en base de datos relacional) e impresión de alarmas tanto digitales como analógicas, y permite la notificación al operador de condiciones del sistema de dos modos distintos: Alarmas y Eventos. Una alarma es un proceso anormal que puede ser perjudicial para el proceso y que normalmente requiere de algún tipo de actuación por parte del operador. Un evento es un mensaje de estado normal del sistema que no requiere ningún tipo de respuesta por parte del operador.

Tipos de Alarmas

Las alarmas pueden dividirse en los siguientes tipos:

Condición de Alarma	Tipo
Discrete	DISC
Deviation - Major	LDEV
Deviation - Minor	SDEV
Rate-Of-Change (ROC)	ROC
SPC	SPC
Value - LoLo	LOLO
Value - Lo	LO
Value - Hi	HI
Value - HiHi	HIHI

Cada alarma se asocia a un tag. Dependiendo del tipo de tag podremos crear uno u otro tipo de alarma.

Prioridades de las Alarmas

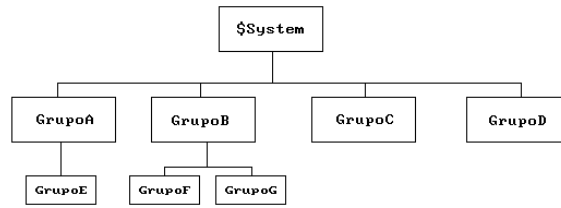
A cada alarma de cada tag puede asociarse un nivel de prioridad (importancia) de 1 a 999 (Prioridad 1 es más crítica). Ello permite filtrar alarmas en displays, en impresora o en disco duro.

Grupos de Alarmas

InTouch dispone de un cómodo sistema para prioridades de alarmas.

Cuando creamos un tagname de alarma, le asignamos un grupo de alarmas. Estos grupos o "jerarquía" de alarmas permiten significar qué alarmas son más importantes, a la vez que permiten reconocer un grupo de alarmas en lugar de todas a la vez.

Al crear un tag, lo asociamos a un grupo (si no lo hacemos, el tag de alarma queda automáticamente asociado al grupo principal, llamada \$SYSTEM).



Estos grupos y subgrupos funcionan de un modo análogo al sistema de jerarquía utilizado por el MsDOS.

Cómo Crear Grupos de Alarmas

Los grupos de alarmas se crean desde */Special/Alarm Groups*. Desde aquí también "emparentamos" unos grupos con otros, hasta crear toda la "jerarquía"

Definición de una Condición de Alarma en un Tagname

Para definir un tagname debemos seleccionar *ALARMS* en el momento de definir el tagname.

Details Alarms Both None

Para las alarmas discretas disponemos de las siguientes posibilidades:

Alarm State:
 Off On None

Off: Define alarma cuando el registro está desactivado

On: Define alarma cuando el registro está activado

Priority: Aparece al definir on/off. Entre 1 y 999. Permite definir la prioridad de la alarma

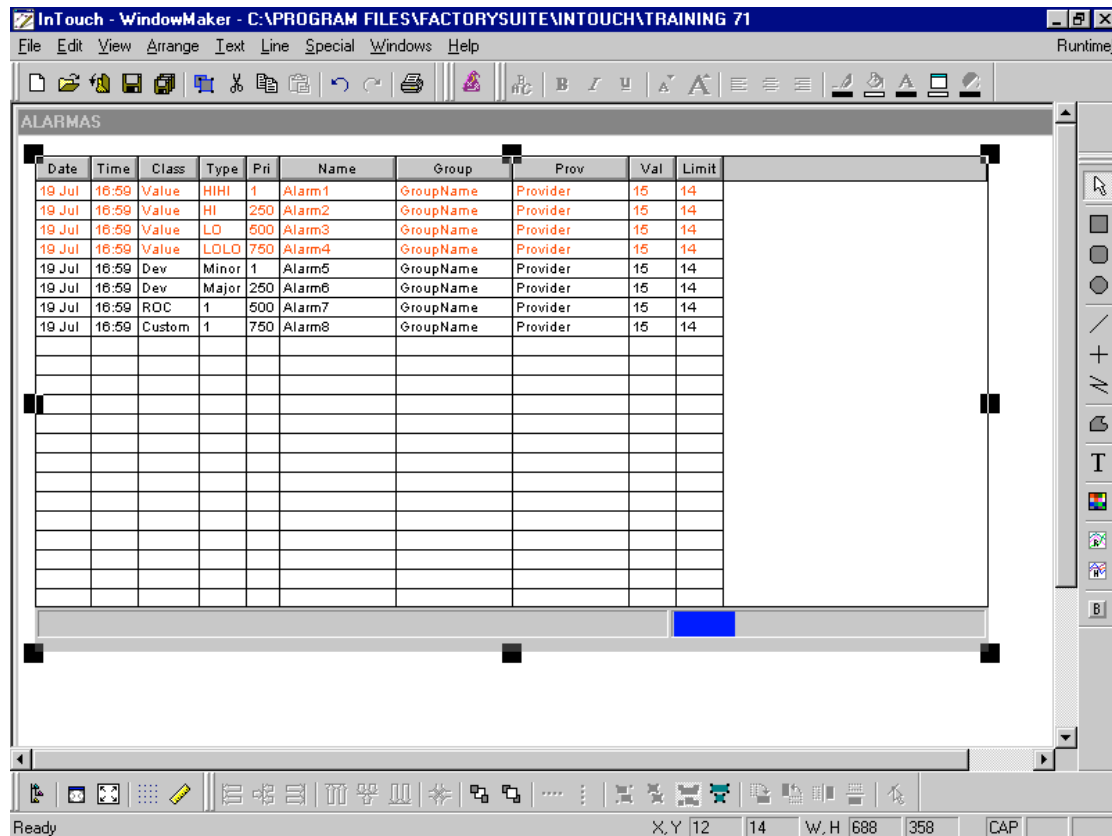
Para las alarmas analógicas, disponemos de las siguientes posibilidades:

<i>Alarm Value</i>	Valores límite de la alarma. 4 niveles
<i>Pri</i>	Prioridad de la alarma
<i>Minor/Major Deviation</i>	Se utiliza para detectar cuándo el valor analógico es una desviación mayor o menor del valor especificado en <i>Target</i>

<i>% Deviation</i>	Porcentaje de desviación permitido al tagname con respecto al <i>Target</i> para que se produzca una alarma por desviación mayor o menor
<i>Target</i>	Valor de referencia para los porcentajes mayor/menor de desviación
<i>Rate of Change</i>	Este tipo de alarma detecta cuándo el valor de la alarma varía en exceso de acuerdo a una cantidad de tiempo

Creación de un Objeto de Alarmas

Dentro del icono de *WIZARDS* de la *toolbox* (Caja de Herramientas) encontraremos el objeto *ALARMAS*. Para crear una ventana de alarmas, basta con seleccionar el objeto como si se tratara de un rectángulo o un círculo. Definimos su tamaño, y la ventana de alarmas quedará creada. Esta ventana puede ser tan grande como toda la pantalla.



Configuración de un Objeto de Alarmas

<i>Display Name</i>	Nombre unívoco del display que estamos parametrizando
<i>New Alarms Appear At</i>	Alarmas nuevas al principio del display (Top), o al final (Bottom)
<i>Show Titles</i>	Muestra o no títulos
<i>Show Vert Scroll Bar</i>	Muestra o no la barra de scroll vertical
<i>Show Horz Scroll Bar</i>	Muestra o no la barra de scroll horizontal
<i>Show Status Bar</i>	Muestra o no la barra de estado
<i>Allow Runtime Grid Changes</i>	Permite o no cambios en la grilla de visualización durante el runtime
<i>Perform Query on Startup</i>	Ejecuta consulta sobre alarmas al arrancar
<i>Auto Scroll to New Alarms</i>	Se posiciona automáticamente en la nueva alarma
<i>Allow Runtime Alarm Selection</i>	Permite selección de alarmas en runtime
<i>Use Extended Alarm Selection</i>	Usa selección extendida de alarmas
<i>From ... To Priority</i>	Niveles de prioridad de alarma mayor y menor que se displayarán en el objeto de alarmas
<i>Alarm State</i>	Define si queremos ver todas las alarmas (all) o sólo las no reconocidas (unack)
<i>Query Type</i>	Seleccione <i>Alarm Summary</i> para displayar las alarmas activas actualmente. Seleccione <i>Alarm History</i> para displayar las alarmas activas y también aquellas que han dejado de serlo.
<i>Alarm Query</i>	Tipo de query. Por defecto es \intouch!\$system (alarmas del propio nodo, del grupo \$system).- Es posible cambiar el nodo del que llegan las alarmas (\nodo1\intouch!\$system), o el grupo (\intouch!grupo1)

Utilice las pestañas superiores de Mensaje y Color para seleccionar los campos que desea visualizar en cada display de alarma, así como los colores de la alarma en estado de Set, Reconocimiento, Reset y Evento

<i>Date</i>	Definimos si queremos que aparezca la fecha, y en caso positivo su formato
<i>Time</i>	Definimos si queremos que aparezca la hora, y en caso positivo su formato
<i>Alarm State</i>	Estado de la Alarma
<i>Alarm Class</i>	Clase de la Alarma
<i>Alarm Type</i>	Definimos si queremos ver el tipo de alarma: HIHI, LO, etc Tipo de condición de alarma LDEV Para condiciones de desviación menor de alarma SDEV Para condiciones de desviación mayor de alarma OPR Cuando el operador provoca con un cambio una alarma
<i>Priority</i>	Definimos si queremos ver la prioridad de la alarma
<i>Alarm Name</i>	Nombre de la Alarma
<i>Group Name</i>	Definimos si queremos ver el grupo de alarma asociado
<i>Alarm Provider</i>	Nodo Proveedor de la Alarma
<i>Value at Alarm</i>	Definimos si queremos ver el valor current del registro dentro de la pantalla de alarmas
<i>Limit</i>	Definimos si queremos ver el valor límite de la alarma del registro
<i>Operator</i>	Seleccionar si queremos asociar el número de identificación del operador con la alarma
<i>Comment</i>	Definimos si queremos ver el comentario asociado

Configuración de Alarmas/Eventos

Disponemos de tres menús para la configuración de las alarmas. A ellos se accede desde el menú *Special*→*Configure*

- | | |
|--------------------------|--|
| 1. <i>GENERAL</i> | Configuración de parámetros de las alarmas en pantalla |
| 2. <i>ALARM LOGGING</i> | Configuración del fichero de alarmas |
| 3. <i>ALARM PRINTING</i> | Configuración de la impresión de alarmas |

Creación de Condiciones de Reconocimiento

Podemos crear pulsadores de reconocimiento utilizando el campo **.Ack** en un pulsador. Es también posible reconocimiento por grupo, por tag seleccionado, por display de alarmas, etc.

Campos de las Alarmas

<i>.Ack</i>	Campo discreto de lectura/escritura para reconocimiento de alarmas. P TagName.Ack=1; .AckNombreGrupo; .AckVariableGrupo;
<i>.Alarm</i>	Campo discreto de lectura que se activa cuando se activa el tagname o grupo GrupoAlarma.Alarm;
<i>.AlarmClass</i>	Devuelve la clase de la alarma
<i>.AlarmComment</i>	Devuelve el comentario de la alarma
<i>.AlarmDate</i>	Devuelve la fecha de la alarma
<i>.AlarmDevDeadband</i>	Campo analógico de lectura/escritura que controla el porcentaje de desviación mayor y menor de las alarmas. Por ejemplo, para cambiar el porcentaje de desviación al 25%, utilizaremos: TagName.AlarmDevDeadband=25;
<i>.AlarmEnable</i>	Discreto de lectura/escritura, que activa/desactiva eventos y alarmas de un tagname o grupo. P.e. GroupVariable.AlarmEnable=1;
<i>.AlarmGroupSel</i>	Devuelve el grupo al que pertenece a alarma
<i>.AlarmLimit</i>	Devuelve el límite de la alarma
<i>.AlarmName</i>	Devuelve el nombre de la alarma
<i>.AlarmOprName</i>	Devuelve el nombre del operador de la alarma
<i>.AlarmOprNode</i>	Devuelve el nombre del operador del nodo de la alarma
<i>.AlarmPri</i>	Devuelve la prioridad de la alarma
<i>.AlarmProv</i>	Devuelve el proveedor de la alarma
<i>.AlarmState</i>	Devuelve el estado de la alarma
<i>.AlarmTime</i>	Devuelve la hora de la alarma
<i>.AlarmType</i>	Devuelve el tipo de la alarma
<i>.AlarmValDeadband</i>	Análogo de lectura/escritura que controla el deadband de una alarma o grupo. P.e. TagName.AlarmValDeadband=10;
<i>.AlarmValue</i>	Devuelve el valor de la alarma
<i>.DevTarget</i>	Campo analógico de lectura/escritura que controla la desviación mayor/menor. P.e. TagName.DevTarget=500;
<i>.HiHiLimit</i> <i>.HiLimit</i> <i>.LoLimit</i> <i>.LoLoLimit</i>	Discreto de lectura y escritura que indica los límites de una alarma
<i>.HiHiStatus</i> <i>.HiStatus</i> <i>.LoStatus</i> <i>.LoLoStatus</i>	Discreto sólo de lectura que indica el estado en que se encuentra una alarma
<i>.MajorDevPct</i>	Entero de lectura/escritura para controlar el porcentaje de desviación mayor de la alarma
<i>.MinorDevPct</i>	Entero de lectura/escritura para controlar el porcentaje de desviación menor de la alarma

<i>.MajorDevStatus</i>	Discreto que indica si la alarma se encuentra en estado de desviación mayor por porcentaje
<i>.MinorDevStatus</i>	Discreto que indica si la alarma se encuentra en estado de desviación menor por porcentaje
<i>.Name</i>	Devuelve el nombre actual de la alarma
<i>.Normal</i>	Discreto de lectura que cuando vale 1 indica que no hay alarmas para el tag especificado
<i>.ROCPct</i>	Lectura/escritura. Monitoriza y controla el valor de Ratio de cambio de una alarma
<i>.ROCStatus</i>	Indica si una alarma se encuentra en estado ROC

Sección 2. Eventos

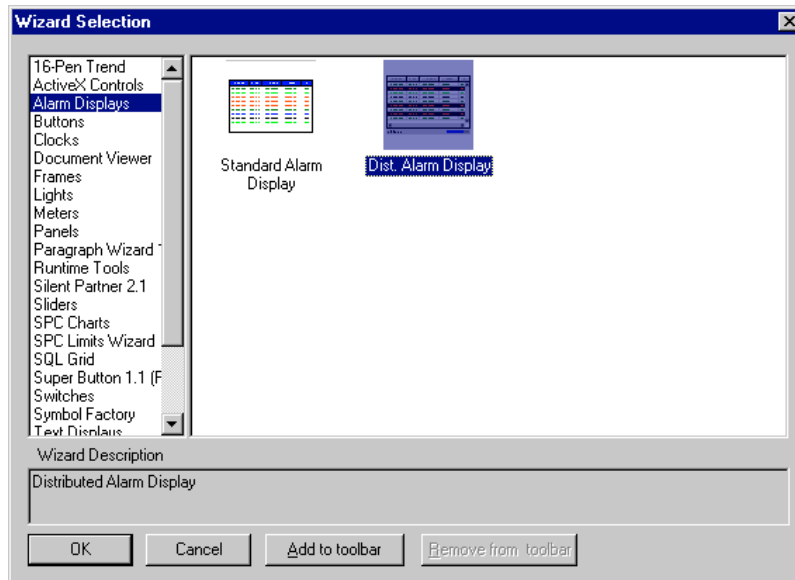
Los eventos representan mensajes de estado normal del sistema y no requieren respuesta por parte del operador. Un evento se produce cuando se produce alguna condición del sistema, por ejemplo cuando un operador entra en el sistema.

EVENTO	CONDICION
<i>ACK</i>	Se ha reconocido una alarma
<i>ALM</i>	Se ha producido una alarma
<i>EVT</i>	Se ha producido un evento
<i>RTN</i>	El tagname ha vuelto a su estado normal desde el estado de alarma
<i>SYS</i>	Evento de sistema
<i>USER</i>	Ha cambiado \$Operador
<i>DDE</i>	Un cliente DDE ha hecho un POKE sobre un tagname
<i>LGC</i>	Una Quicksript ha modificado el valor de un tagname
<i>OPR</i>	Un operador ha modificado el valor de una tagname usando un Value Input (entrada de teclado)

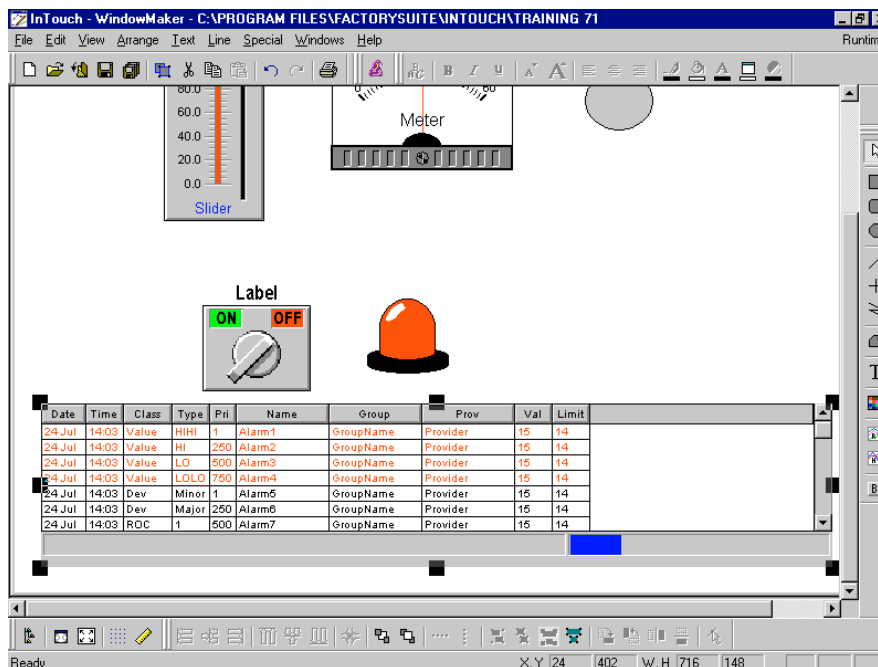


EJERCICIO 8. CONFIGURACION DE UN VISOR DE ALARMAS

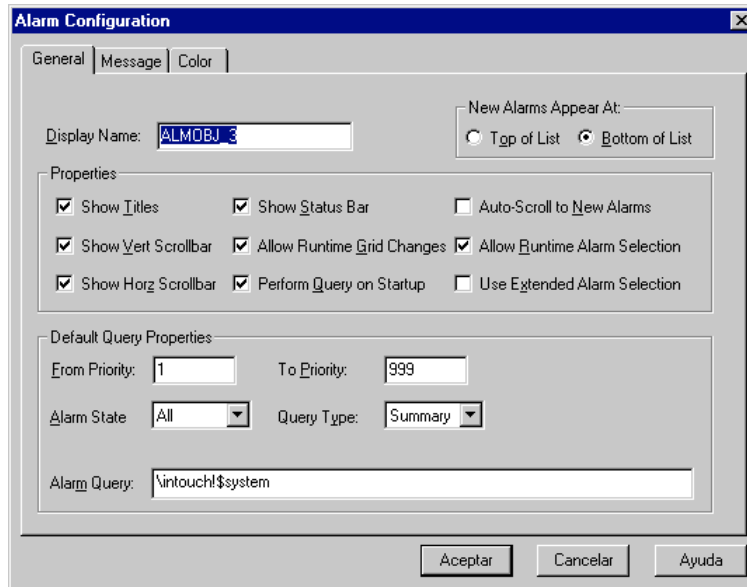
- Seleccionaremos el símbolo de *Dist. Alarm Display* de entre los WIZARDS (categoría *ALARM DISPLAYS*)



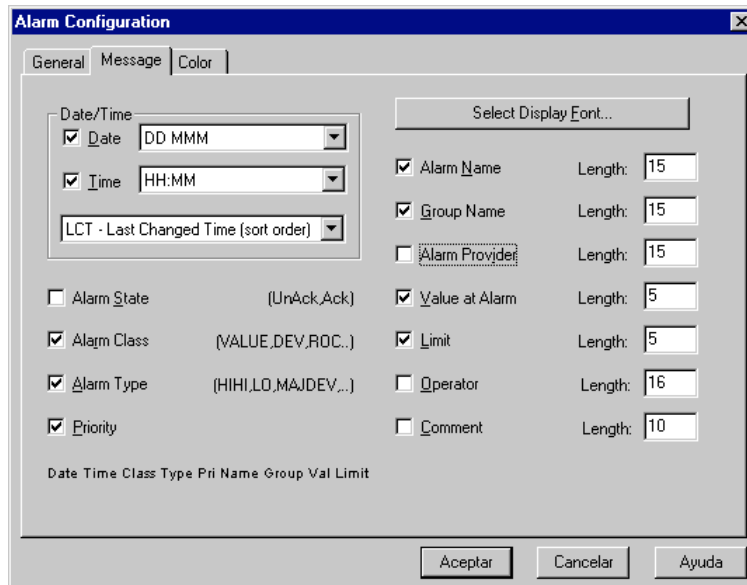
Lo colocamos en la pantalla en una posición libre de la misma. Recuerde que puede mover o cambiar el tamaño del objeto



Hacemos doble clic sobre el objeto de alarmas para configurarlo:



Pinchamos sobre *MESSAGE* para la configuración de la ventana de alarmas



Haremos una cruz pinchando con el ratón sobre aquellos campos que nos interesen visualizar en la ventana de alarmas



EJERCICIO 10.CONFIGURACION DE UNA ALARMA DIGITAL

Seleccionamos el tagname *MOTOR*. Pinchar sobre *ALARMS* para definir la alarma

Main Details Alarms Details & Alarms Members

Aparecerá inmediatamente el cuadro de *ALARM STATE*. Seleccione *On* para significar que se produce una alarma cuando el motor está en marcha :

Alarm State: On Off None Priority:

Dirjase al runtime para comprobar el efecto que ha producido la creación de la alarma.



EJERCICIO 10. CONFIGURACION DE UNA ALARMA ANALOGICA

Para crear una alarma analógica, seleccionamos el tagname *TEMPERATURA*. Pinchar sobre *ALARMS* para definir la alarma

Definiremos 4 valores según la importancia de la alarma

Alarm Value	Pri	
<input checked="" type="checkbox"/> LoLo	10	1
<input checked="" type="checkbox"/> Low	20	1
<input checked="" type="checkbox"/> High	80	1
<input checked="" type="checkbox"/> HiHi	90	1
Value Deadband:		0

<input type="checkbox"/> Minor Deviation
<input type="checkbox"/> Major Deviation
Deviation Deadband %: 0
<input type="checkbox"/> Rate of Change

Dirjase al runtime para comprobar el efecto que ha producido la creación de la alarma
Podemos también definir la alarma analógica por desviaciones porcentuales con respecto a una valor de muestra o a un tiempo determinado

Alarm Value	Pri	% Deviation	Target	Priority
<input checked="" type="checkbox"/> LoLo	10	1		1
<input checked="" type="checkbox"/> Low	20	1		1
<input checked="" type="checkbox"/> High	80	1		1
<input checked="" type="checkbox"/> HiHi	90	1		1
Value Deadband:		0		

<input checked="" type="checkbox"/> Minor Deviation	10	50	1
<input checked="" type="checkbox"/> Major Deviation	15		1
Deviation Deadband %: 0			
<input checked="" type="checkbox"/> Rate of Change	20	% per	1
		<input type="radio"/> Sec <input checked="" type="radio"/> Min <input type="radio"/> Hr	

Dirjase al runtime para comprobar el efecto que ha producido la creación de la alarma
Cree un nuevo objeto de alarmas pero en este caso configúrelo como de *ALARM HISTORY*.

Alarm Configuration

General | Message | Color

Display Name: ALMOBJ_3

New Alarms Appear At:
 Top of List Bottom of List

Properties

Show Titles Show Status Bar Auto-Scroll to New Alarms
 Show Vert Scrollbar Allow Runtime Grid Changes Allow Runtime Alarm Selection
 Show Horz Scrollbar Perform Query on Startup Use Extended Alarm Selection

Default Query Properties

From Priority: 1 To Priority: 999

Alarm State: All Query Type: **Historical**

Alarm Query: \intouch!\\$system

Aceptar Cancelar Ayuda

Defina el visor como Histórico

Dirjase al runtime para comprobar el efecto que ha producido este cambio

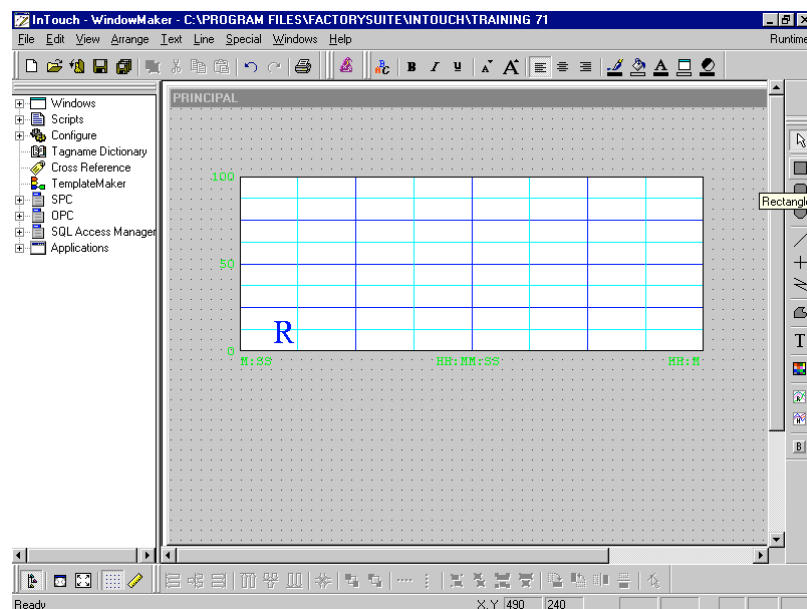
MODULO 7. CURVAS HISTORICAS Y REALES

El paquete de software **InTouch** permite displayar curvas y tendencias en pantalla tanto en tiempo real como de valores históricos.

Sección 1. Curvas en Tiempo Real

Creación de una Curva en Tiempo Real

Disponemos de una herramienta en la *toolbox* que permite crear una curva en tiempo real. Para ello, la seleccionaremos y crearemos un rectángulo del tamaño que deseemos el gráfico. Cada gráfico puede visualizar hasta 4 lápices.



Configuración de la Curva en Tiempo Real

Dentro de una curva en tiempo real, podemos definir una serie de parámetros, tales como tiempo de actualización de la curva, tamaño de la muestra, colores de los lápices, color del fondo y otros.

Real Time Trend Configuration

Comment:

Time
Time Span:
 Sec Min Hr

Sample
Interval:
 Msec Sec Min Hr

Color
Chart Color:
Border Color:

Time Divisions
Number of Major Div:
Minor Div/Major Div:
 Top Labels Bottom Labels
Major Div/Time Label:
HH:MM:SS Display: HH MM SS

Value Divisions
Number of Major Div:
Minor Div/Major Div:
 Left Labels Right Labels
Major Div/Value Label:
Min Value: Max:

Pen:	Expression:	Color:	Width:
1	<input type="text"/>	<input type="color" value="green"/>	<input type="text" value="1"/>
2	<input type="text"/>	<input type="color" value="orange"/>	<input type="text" value="1"/>
3	<input type="text"/>	<input type="color" value="yellow"/>	<input type="text" value="1"/>
4	<input type="text"/>	<input type="color" value="black"/>	<input type="text" value="1"/>

OK Cancel Clear Select Display Font ... Only update when in memory

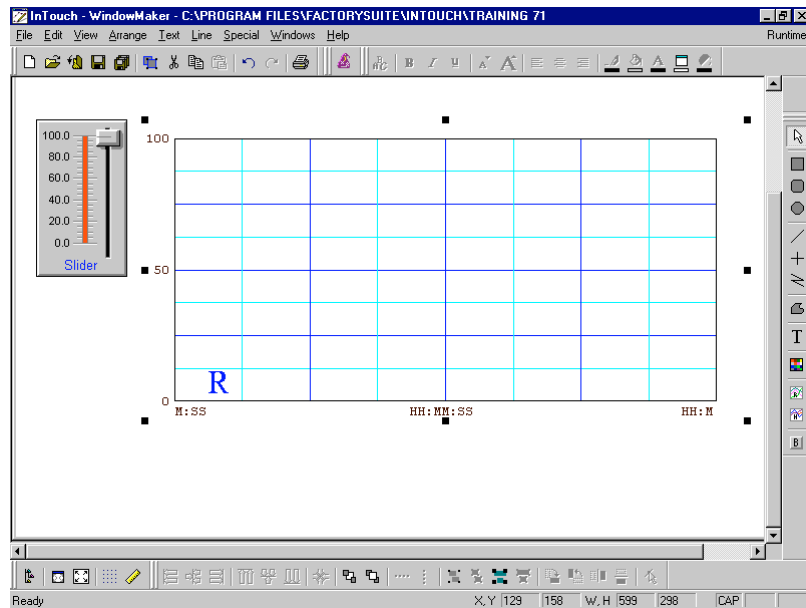
Las curvas en tiempo real no almacenan el valor displayado, sino que simplemente representan un valor o registro real del PLC (o interno de **InTouch**) y lo representan en pantalla en forma de curva.



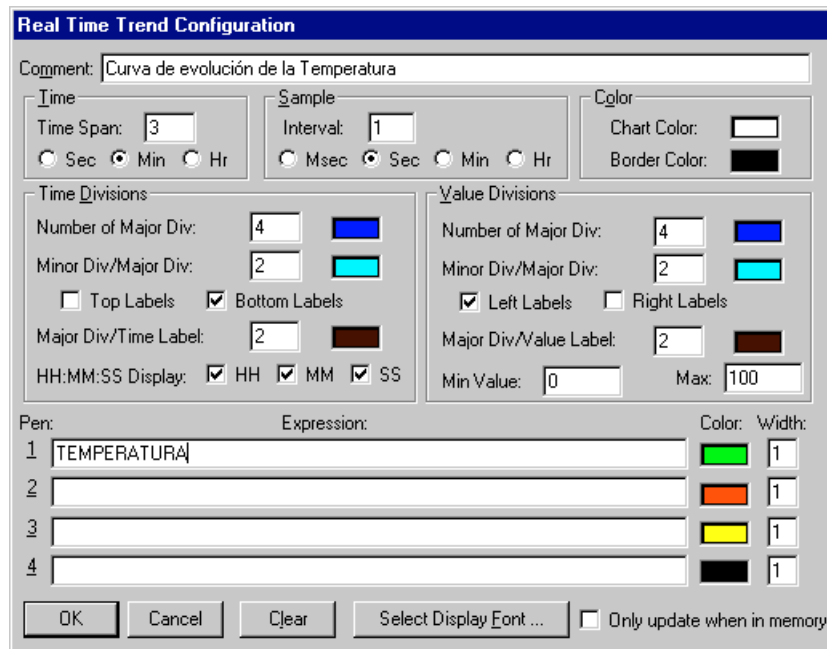
EJERCICIO 11. CURVAS EN TIEMPO REAL

- Utilice otra pantalla para ver los datos en tiempo real. Sobre esta pantalla, cree otro slider y asócielo el tagname *TEMPERATURA*.

Seleccione el icono de Curva Real desde la *toolbox* y coloque una curva R sobre la pantalla:



Haga doble clic sobre la curva para configurarla:



Diríjase al runtime para comprobar el efecto que ha producido la nueva variación en el programa.

Sección 2. Curvas Históricas

Nota : **InTouch** permite la visualización de históricos distribuidos (gestión de los históricos bajo una estructura cliente/servidor en una red de ordenadores). Para este cursillo se da únicamente tratamiento a las curvas históricas locales.

Las curvas históricas permiten visualizar la evolución con respecto al tiempo de un dato en forma de curva o tendencia. Este dato debe haber sido almacenado previamente, por lo que el tagname visualizado en este tipo de curvas debe haber sido previamente definido como del tipo *logged*

Configurar HistoricalLogging

Para activar el gestor de históricos de **InTouch**, es necesario acudir a SPECIAL→CONFIGURE→HISTORICAL LOGGING

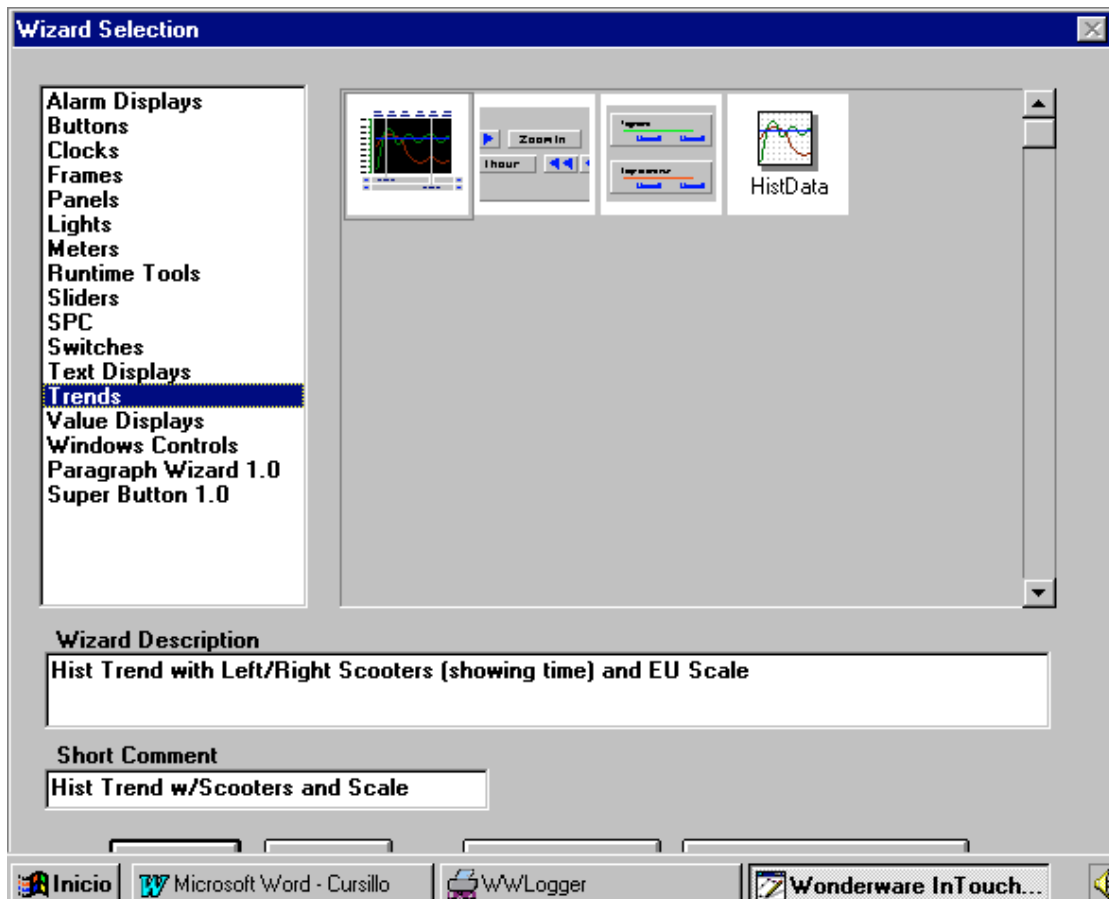
<i>Enable Historical Logging</i>	Habilítelo para activar el gestor de históricos
<i>Keep Log Files For</i>	Indique e número de días que quiere mantener en disco los ficheros históricos. "0" significa que jamas se borran automáticamente.
<i>Store Log Files ...</i>	Indique la carpeta donde se guardarán los ficheros históricos
<i>Printing Control</i>	Control de impresión. Recomendamos dejar por defecto

Tras cambiar estos parámetros, es necesario reiniciar WindowViewer

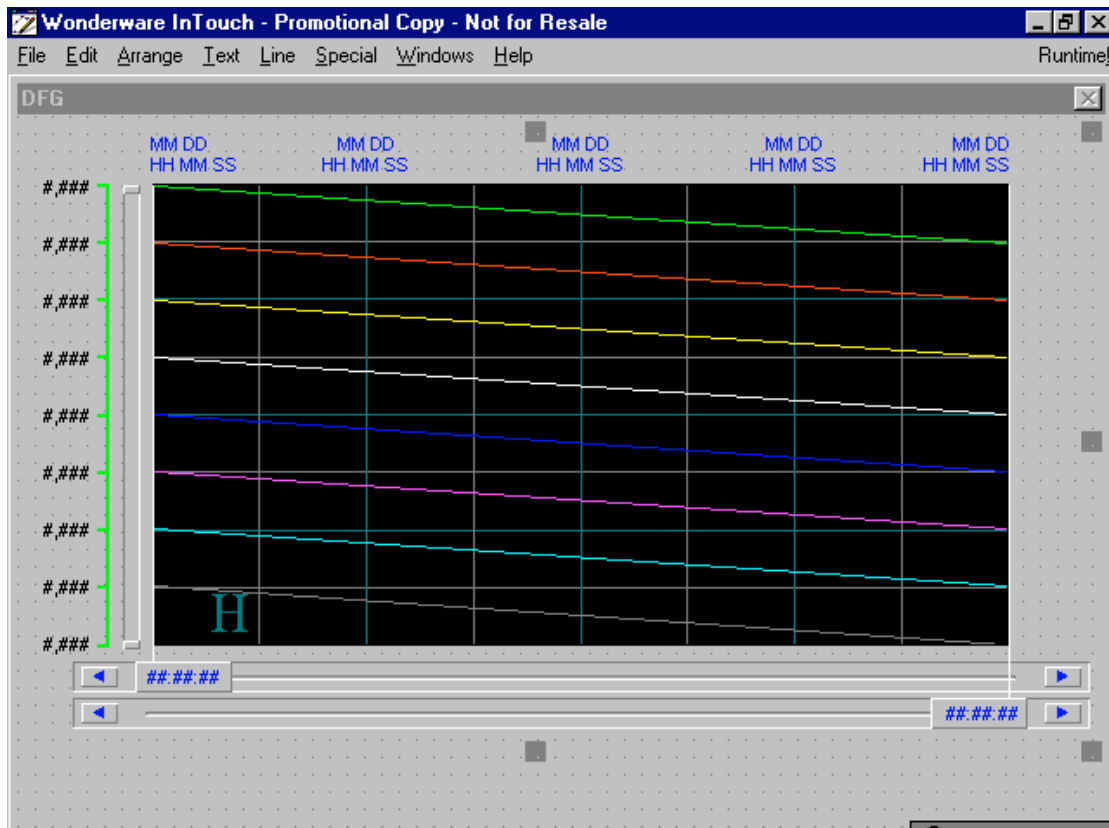
Utilización del Wizard de Curva Histórica

WIZARD DE CURVA

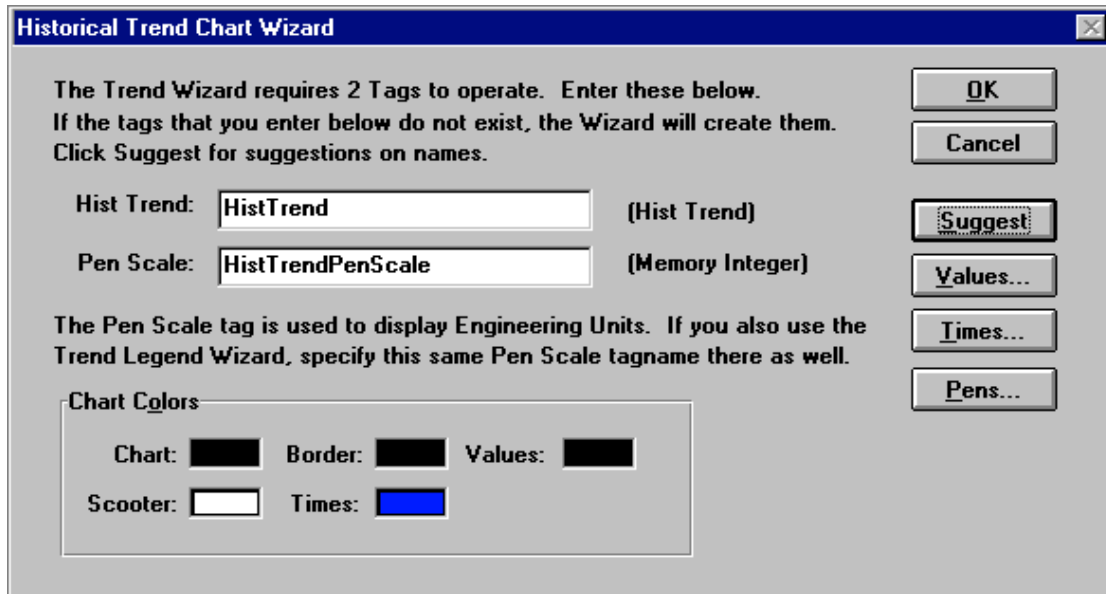
Dentro de los WIZARDS disponemos de la herramienta *CURVA HISTORICA*. Esta herramienta es muy completa, ya que nos permite disponer de la mayoría de elementos de una curva sin necesidad de programación.



Hacemos doble clic con el ratón sobre el objeto de curva histórica (situado el primero de los 4 iconos). Situamos la curva histórica del WIZARD sobre la pantalla.



La curva quedará situada sobre la pantalla. Podremos escalarla o moverla según nuestra necesidad. Haciendo doble click con el ratón en su anterior, podemos definir sus características, valores de los lápices y tiempos de visualización



En el cuadro anterior de programación, definimos lo siguiente:

<i>HistTrend</i>	Tag asociado a la curva histórica. Oprimiendo el pulsador <i>Suggest</i> , InTouch nos sugerirá un nombre de tag para evitarnos programación.
<i>Pen Scale</i>	Tag asociado a la curva histórica. Oprimiendo el pulsador <i>Suggest</i> , InTouch nos sugerirá un nombre de tag para evitarnos programación
<i>Values</i>	Permite definir el eje vertical (valores)
<i>Times</i>	Permite definir el eje horizontal (tiempos)
<i>Pens</i>	Permite definir los valores asociados a cada uno de los lápices. Disponemos de hasta 8 lápices por curva

WIZARD DE PULSADORES DE LA CURVA

Dentro de los WIZARDS disponemos de la herramienta de *PULSADORES DE LA CURVA HISTORICA*. Esta herramienta tiene ya programados unos pulsadores para hacer zooms de la curva, tiene también programados los punteros de selección de fecha/hora, etc. Nos permite definir diversas funciones sin necesidad de programación.



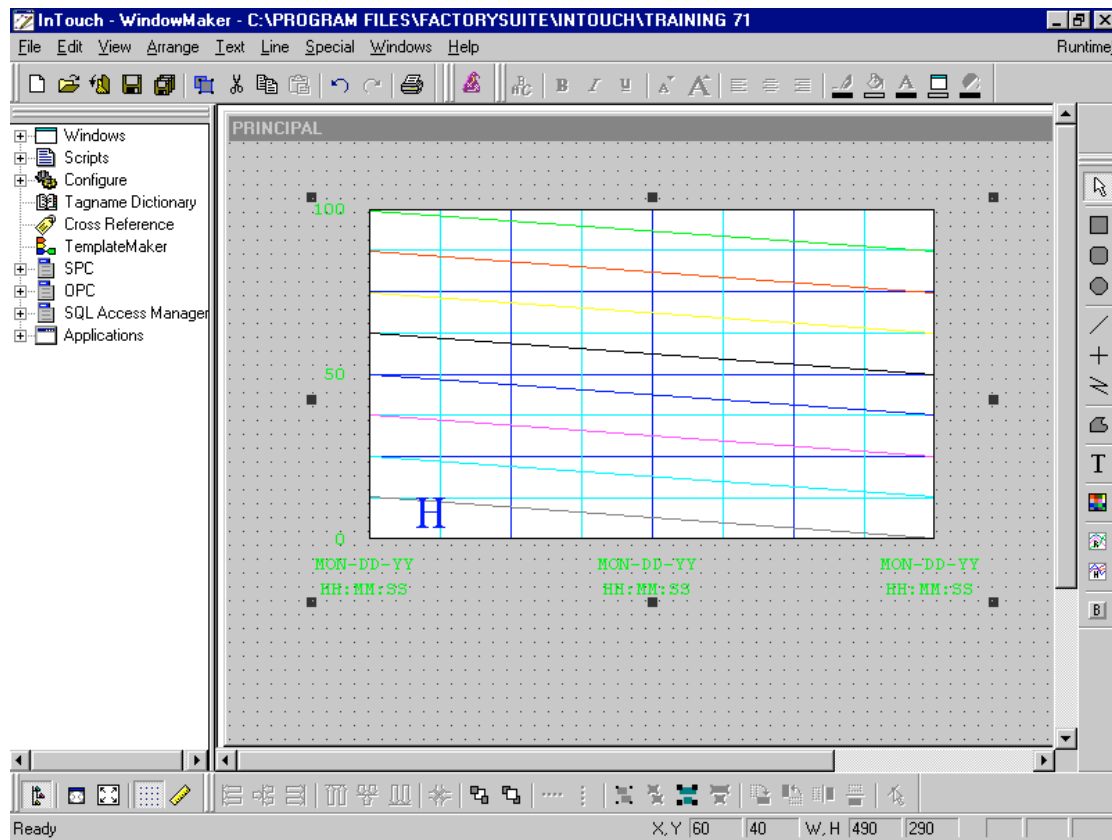
Este elemento lo referenciamos a la curva que deseemos asociándole el mismo tagname

WIZARD DE VISUALIZACION DE VALORES DE LA CURVA

Este elemento permite visualizar los valores históricos que tenemos en cada uno de los punteros.

Creación de la Curva Histórica (sin utilizar el Wizard)

La curva o tendencia histórica es creada de un modo análogo a la curva en tiempo real, desde la caja de herramientas o *toolbox*.



CONFIGURACION DE LA CURVA

Dentro de una curva histórica, podemos definir una serie de parámetros, tales como tiempo de actualización de la curva, tamaño de la muestra, colores de los lápices, color del fondo y otros.

Historical Trend Configuration

Historical Tag:

Chart time
Initial Time Span:
 Secs Mins Hrs Days

Initial Display Mode
 Min/Max Average

Color
 Chart Color:
 Border Color:

Time Divisions
 Number of Major Div:
 Minor Div/Major Div:
 Top Labels Bottom Labels
 Major Div/Time Label:
 Time: MM DD YY
 HH MM SS

Value Divisions
 Number of Major Div:
 Minor Div/Major Div:
 Left Labels Right Labels
 Major Div/Value Label:
 Min Value: Max:

Pen	Tagname	Color	Width	Pen	Tagname	Color	Width
1	<input type="text"/>	<input type="text" value="Green"/>	<input type="text" value="1"/>	5	<input type="text"/>	<input type="text" value="Blue"/>	<input type="text" value="1"/>
2	<input type="text"/>	<input type="text" value="Orange"/>	<input type="text" value="1"/>	6	<input type="text"/>	<input type="text" value="Pink"/>	<input type="text" value="1"/>
3	<input type="text"/>	<input type="text" value="Yellow"/>	<input type="text" value="1"/>	7	<input type="text"/>	<input type="text" value="Cyan"/>	<input type="text" value="1"/>
4	<input type="text"/>	<input type="text" value="Black"/>	<input type="text" value="1"/>	8	<input type="text"/>	<input type="text" value="Grey"/>	<input type="text" value="1"/>

Allow runtime changes

Podemos definir del mismo modo si deseamos que el operador pueda, durante el runtime, efectuar cualquier cambio acerca de esta configuración primera. Ello se hace activando el siguiente piloto:

Allow runtime changes

Ello provocará que la curva histórica sea, durante el runtime, un elemento interactivo, es decir, pulsando sobre la tendencia o pulsando la tecla funcional definida, el operador podrá modificar cualquier parámetro de esa tendencia de un modo sencillo, gracias a la siguiente ventana:

Historical Trend Setup

Chart Start
 Month Day Year Hour Min Sec
 / / : :

Display Mode
 Min/Max
 Avg/Scatter
 Avg/BarChart

Chart Length
 Days Hrs Mins Secs

Chart Range
 Min: % Max: %

Tags

<input type="checkbox" value="Green"/>	Pen #1 ...	TEMPERATURA
<input type="checkbox" value="Orange"/>	Pen #2 unassigned ...
<input type="checkbox" value="Yellow"/>	Pen #3 unassigned ...
<input type="checkbox" value="Black"/>	Pen #4 unassigned ...

Esta ventana es propia de **InTouch**, por lo que no es necesario crearla, sino que aparece al seleccionar la curva histórica durante el runtime (es posible crear una ventana distinta a esta para permitir al operador modificar datos de la curva).

Campos (.Fields) de la Tendencias Históricas

<i>.ChartLength</i>	Campo analógico de lectura/escritura que permite controlar la longitud del tiempo displayado en una tendencia histórica. .ChartStart displaya la longitud en segundos P.e: <u>Fuerza el span de la tendencia a 1 hora:</u> ht.ChartLength=3600{60 minutos * 60 seconds/minute}
<i>.ChartStart</i>	Campo analógico de lectura/escritura que permite controlar el tiempo de inicio de la tendencia y/o el scroll. .ChartStart displaya el número de segundos transcurridos desde las 12:00 a.m. 1/1/70. P.e: <u>Scroll a la derecha 1 minuto</u> ht.ChartStart=ht.ChartStart + 60; <u>Scroll a la izquierda 1 hora</u> ht.ChartStart=ht.ChartStart - 3600;
<i>.DisplayMode</i>	Campo analógico de lectura/escritura que determina el método para displayar valores en la tendencia: 1. Displaya mín/máx para cada muestra 2. Displaya una media en diagrama "scatter" 3. Displaya la media en diagrama de barras
<i>.MaxRange</i> <i>.MinRange</i>	Campo analógico de lectura/escritura que representa el porcentaje del rango del tagname en unidades de ingeniería. Los límites van de 0 a 100
<i>.Pen1 - .Pen8</i>	Campo analógico de lectura/escritura que determina el tagname visualizado en la tendencia
<i>.ScooterLockLeft</i>	Discreto de lectura/escritura. Si es verdadero, bloquea la posición del puntero izquierdo.
<i>.ScooterLockRight</i>	Discreto de lectura/escritura. Si es verdadero, bloquea la posición del puntero derecho
<i>.ScooterPosLeft</i>	Real de lectura/escritura, que representa la posición del puntero izquierdo.
<i>.ScooterPosRight</i>	Real de lectura/escritura, que representa la posición del puntero derecho.
<i>.TagID</i>	Campo analógico de lectura/escritura que determina el tagname visualizado en la tendencia (en conjunción con el campo <i>.pen1</i>)
<i>.UpdateCount</i>	Entero de sólo lectura que se incrementa cuando se ha completado una actualización de datos históricos
<i>.UpdateInProgress</i>	Discreto de sólo lectura que indica la actualización de los datos históricos
<i>.UpdateTrend</i>	Tagname discreto de lectura/escritura, que colocado a 1 actualiza los valores de la tendencia. P.e. HistTrendTag.UpdateTrend=1;

<i>HTGetLastError</i>	Determina si se ha producido algún error durante la actualización del lápiz
<i>HTGetPenName</i>	Devuelve nombre del lápiz asociado al tagname
<i>HTGetTimeAtScooter</i>	Devuelve el tiempo en segundos desde las 00.00.00 horas GMT del 1 de Enero de 1970 en la posición del puntero o scooter
<i>HTGetTimeStringAtScooter</i>	Devuelve la cadena conteniendo fecha/hora para la muestra en la posición del scooter especificado por ScootNum y ScootLoc
<i>HTGetValue</i>	Devuelve el valor especificado
<i>HTGetValueAtScooter</i>	Devuelve el valor especificado para la muestra según

	posición del scooter
<i>HTGetValueAtZone</i>	Devuelve el valor especificado para la muestra según posición entre ambos scooters
<i>HTScrollLeft</i>	Selecciona la hora de inicio de una curva a un valor anterior que el actual de arranque según un porcentaje del ancho del gráfico. Permite hacer un scroll según porcentaje de fecha/hora del gráfico hacia la izquierda
<i>HTScrollRight</i>	Selecciona la hora de inicio de una curva a un valor posterior que el actual de arranque según un porcentaje del ancho del gráfico. Permite hacer un scroll según porcentaje de fecha/hora del gráfico hacia la derecha
<i>HTSetPenName</i>	Asigna un tagname a un lápiz
<i>HTUpdateToCurrentTime</i>	Actualiza el fin de la gráfica a la fecha/hora actual
<i>HTZoomIn</i>	Zoom hacia adentro
<i>HTZoomOut</i>	Zoom hacia fuera

La Utilidad HistData

HistData permite convertir datos de ficheros históricos generados por **InTouch** a ficheros de tipo csv (Comma Separated Value). Estos ficheros son fácilmente manejables por programa tales como Excel, y permiten de este modo generar todo tipo de informes.

Para llevar a cabo la selección de datos a convertir y la conversión, HistData enlaza con tagnames de **InTouch** mediante DDE. Las variables que incluye HistData son las siguientes:

<i>DATADIR</i>	Message	Path del directorio que contiene los ficheros históricos
<i>DBDIR</i>	Message	Path del directorio que tiene los tags de InTouch
<i>STARTDATE</i>	Message	Fecha inicio que queremos convertir los datos (MM/DD/YY)
<i>STARTTIME</i>	Message	Hora inicio que queremos convertir los datos (MM/DD/YY)
<i>DURATION</i>	Message	Longitud de tiempo que queremos recuperar. w(semana); d (día); h (hora); m (minutos); s (segundos) p.e. 5s son 5 segundos.
<i>INTERVAL</i>	Message	Intervalo de tiempo entre muestras (máximo 6 semanas) w(semana); d (día); h (hora); m (minutos); s (segundos)
<i>TAGS</i>	Message	Lista de los tags de los que queremos convertir datos Ej. "\$Date, TagA, TagB"
<i>PRINTTAGNAMES</i>	Discrete	Está a 1 por defecto. Hace que HistData imprima los tagnames en la primera línea del fichero sobre la columna de valores asociada
<i>DATA</i>	Message	Mantiene los datos requeridos en el programa HistData en formato csv permite que otros programas puedan acceder a ellos mediante Advise y Request
<i>SENDDATA</i>	Integer	Al ponerlo a 1, HistData actualiza los datos de DATA con los datos requeridos. Una vez actualizados, SENDATA se pone automáticamente a 0
<i>FILENAME</i>	Message	Pathname completo del fichero al que vamos a escribir los datos.
<i>WRITEFILE</i>	Integer	Al ponerlo a 1, HistData escribirá los datos solicitados al fichero especificado en FILENAME. Cuando se ha escrito el fichero, WRITEFILE se resetea automáticamente a 0
<i>STATUS</i>	Discrete	Visualiza el estado de la última operación. Si es 1 significa que se ha realizado con éxito
<i>ERROR</i>	Message	Cadena que contiene una descripción del último error.



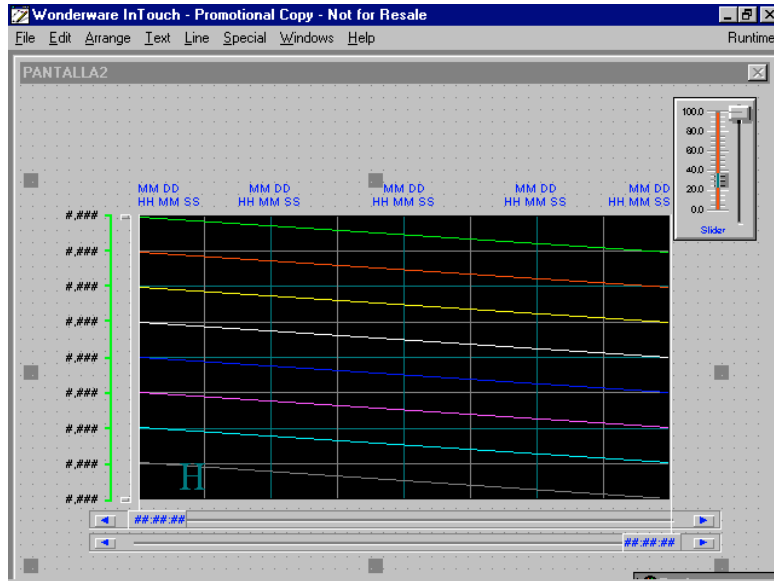
EJERCICIO 12. CURVAS HISTÓRICAS

- Para crear curvas históricas, debemos previamente tener almacenados en disco los datos históricos correspondientes a los tagnames que deseemos visualizar en las curvas.

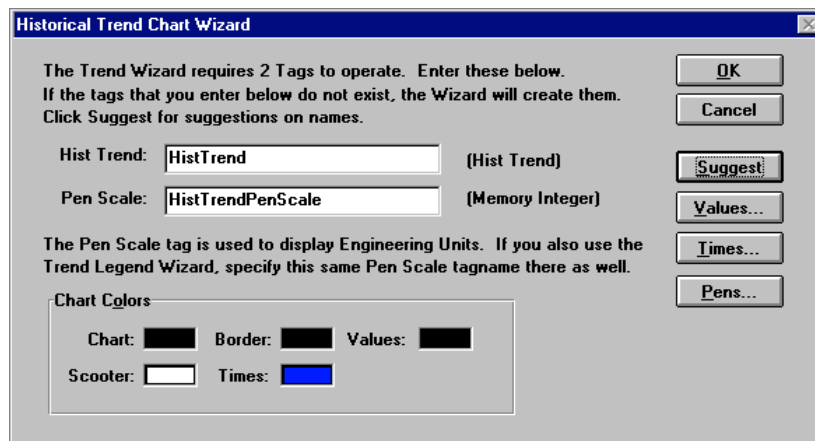
Para ello, es necesario definir el tagname como histórico. Diríjase de nuevo al menú de creación de tagnames y seleccione el tagname TEMPERATURA. Defínalo como LOG tal y como se indica a continuación :

También es necesario activar el gestor de históricos de **InTouch**, tal y como se ha explicado durante el curso. Hágalo accediendo desde *SPECIAL* → *CONFIGURE* → *HISTORICAL LOGGING*

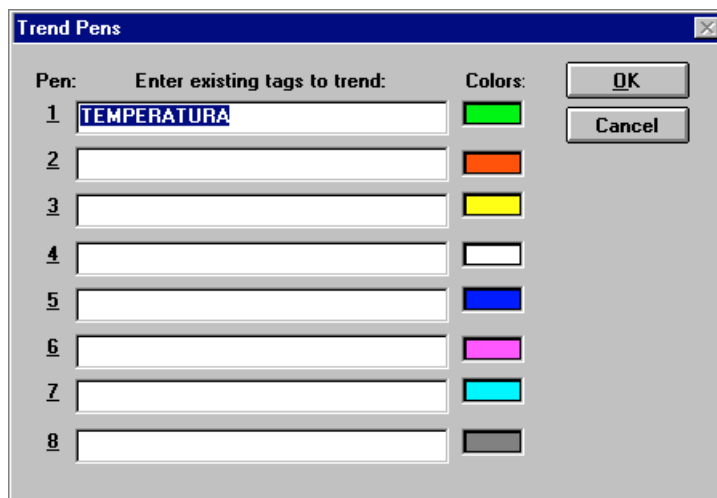
Ya de nuevo en la ventana, seleccione el WIZARD correspondiente a la creación de curvas históricas y sitúelo sobre la pantalla



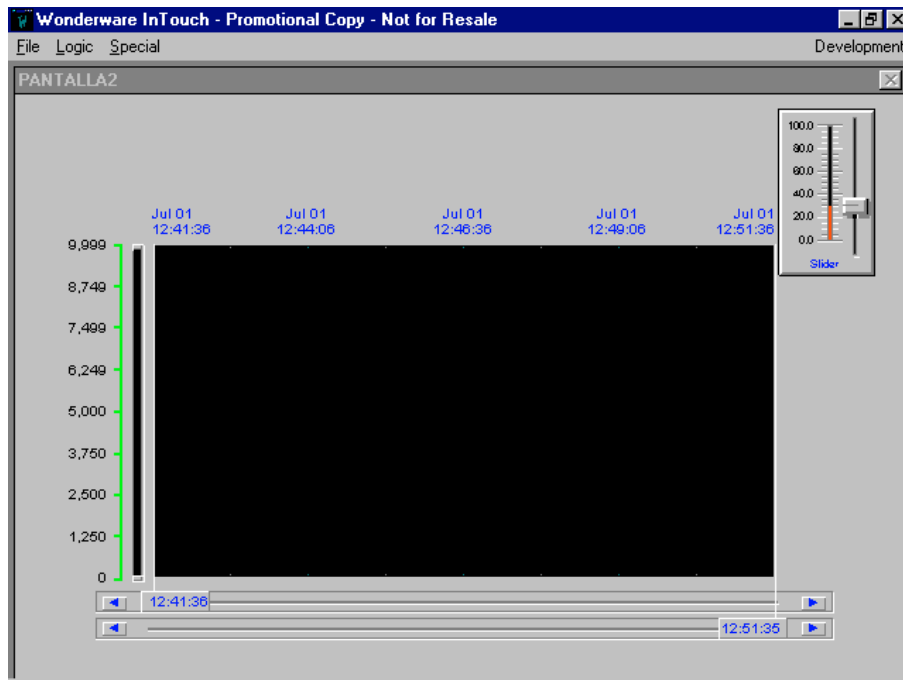
Haga doble clic sobre el WIZARD para configurar la curva. Oprime el botón *SUGGEST* para que sea el propio **INTOUCH** el que le sugiera los tagnames a utilizar.



Oprima el valor *Pens* para seleccionar el tagname *TEMPERATURA*. Puede si lo desea modificar el color de visualización



Diríjase al runtime para comprobar el efecto que ha producido la nueva variación en el programa



MODULO 8. COMUNICACIONES I/O

Sección 1. La Comunicación DDE

DDE es un protocolo de comunicaciones desarrollado por Microsoft para intercambio de datos entre aplicaciones Windows.

DDE es un sistema estándar en *WINDOWS* de muy sencillo uso, en el que la comunicación se establece automáticamente entre programas que contemplan la estructura *DDE* (cliente-servidor). Un programa que puede mandar datos al bus *DDE* es un programa servidor. Un programa cliente puede recibir datos *DDE*. Ello permite que podamos muy fácilmente creamos programas con gestiones especiales en *VBASIC*, *EXCEL*, etc., y pasar los datos a **InTouch** sin necesidad de crear un programa de comunicaciones.

Así ocurre con los servidores de autómatas que dispone **WONDERWARE**, entre los que se incluyen la práctica totalidad de los PLCs más conocidos del mundo con comunicación tanto punto a punto como en red. Por supuesto, podemos direccionar un servidor *DDE* a un puerto de comunicaciones y otro servidor a otro puerto, con lo que podemos compartir información que venga de distintos PLCs o sistemas de campo.

La comunicación DDE se basa en una convención con estos tres parámetros:

- Aplicación
- Tópico
- Elemento

Cuando queremos desde otra aplicación (p.e. Excel), enlazar vía DDE un dato **InTouch**, los parámetros son:

APLICACIÓN	VIEW
TOPICO	TAGNAME
ELEMENTO	{nombre del tag que queremos enlazar}

InTouch Access Name

Para poder enlazar datos vía DDE de otras aplicaciones Windows a **InTouch**, debemos crear un *Access Name*. A cada *Access Name* podemos asociar una Aplicación y un Tópico.

<i>Access Name</i>	Nombre de enlace. Puede ser un nombre aleatorio (recomendamos utilizar el mismo que el t3pico)
<i>Node Name</i>	Nombre del Nodo. S3lo hay que rellenarlo si vamos a leer datos de otro PC. En caso que los datos sean del mismo PC, dejarlo en blanco
<i>Application Name</i>	Nombre de la aplicaci3n de la que queremos leer (pe Excel, Siemens, etc.)
<i>Topic Name</i>	Nombre del t3pico del que queremos leer (pe. Libro1.xls, PLC1, etc.)
<i>Which protocol To Use</i>	DDE para enlace DDE SuiteLink para enlace SuiteLink (s3lo entre aplicaciones FS2000)
<i>When to Advise Server</i>	Normalmente dejar por defecto



EJERCICIO 13.ENLACE InTouch - Excel

Este ejercicio muestra cómo leer La Fila 1 Columna 1 de una hoja de cálculo *EXCEL*.

En primer lugar, cree un Access Name como el que se muestra a continuación. Accederá al menú de Access name a través de *SPECIAL* → *ACCESS NAMES*

Topic Name representa en este caso el nombre del fichero que queremos enlazar

Cree un tagname llamado DatoExcel con los siguientes parámetros:

Nótese que hemos asociado este tagname al *Access Name* de Excel, y que como item o elemento de esa hoja de Excel a leer hemos seleccionado f1c1

MODULO 9. SEGURIDAD

Sección 1. Características de la Seguridad

El concepto de seguridad en **InTouch** se refiere a la opción de proteger ciertas partes del programa para filtrar el acceso al mismo por parte de los distintos operadores. Por ejemplo, podemos disponer de un nivel de acceso que permita visualización y otro para la modificación de parámetros, etc.

Para aplicar seguridad a una aplicación, debemos acceder desde el menú *SPECIAL*→*SECURITY*. El concepto de seguridad se basa en un nombre de acceso y un código. El nombre de acceso más el código secreto permiten un nivel de acceso. De origen **InTouch** dispone del siguiente acceso general :

User Name	Password	Access Level
Administrator	WONDERWARE	9999

Significa que bajo ese Nombre de Usuario (user name) y Password podemos acceder a todos los procesos de **InTouch**. El concepto *ACCESO* es denominado en **InTouch** *LOG IN*. Podemos definir nuevos nombres de usuario con su password asociado, lo que determinará nuevos niveles de acceso. Disponemos de hasta 9999 distintos niveles de acceso. Dentro del programa **InTouch** podemos manejar varios tagnames especiales dedicados a la seguridad. Estos tagnames son los siguientes :

<i>\$AccessLevel</i>	Permite controlar el nivel de acceso de los distintos operadores P.e. Si queremos que un pulsador no sea visible si no se tiene un nivel de acceso mayor que 5000, deberemos crear un pulsador que sólo sea posible cuando <i>\$AccessLevel</i> >5000
<i>\$ConfigureUsers</i>	Tagname discreto de lectura/escritura que nos visualizará la caja de diálogo de InTouch para la configuración de usuarios
<i>\$ChangePassword</i>	Tagname discreto de lectura/escritura que nos visualizará la caja de diálogo de InTouch para el cambio de password
<i>\$InactivityTimeout</i>	Tagname discreto de lectura que se activa cuando ha transcurrido el tiempo de acceso predefinido
<i>\$InactivityWarning</i>	Tagname discreto de lectura que se activa cuando ha transcurrido el tiempo predefinido para avisar al operador que va a finalizar su tiempo de acceso
<i>\$Operator</i>	Tagname discreto de lectura que permite controlar distintas funciones del operador
<i>\$OperatorEntered</i>	Tagname de mensaje de lectura/escritura que incluye el User Name (Nombre de usuario) del operador
<i>\$PasswordEntered</i>	Tagname de mensaje de lectura/escritura que incluye el Password (Código secreto) del operador

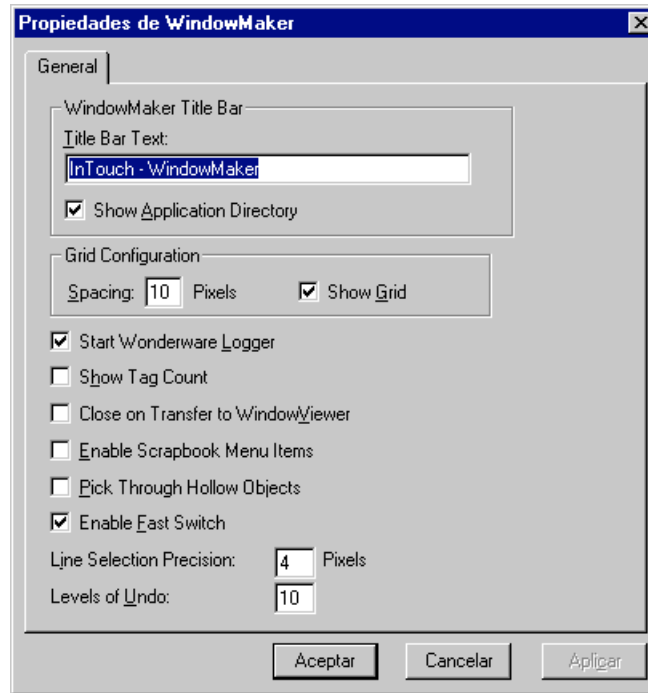
Sección 2. Configuración de la Aplicación

Desde esta función podemos configurar toda la apariencia de la aplicación, tanto del *WINDOWMAKER* como del *WINDOWVIEWER*.

A esta función se accede desde */Special/Configure* en el menú principal de *WINDOWMAKER*

Configurar WindowMaker

Permite configurar apariencia y menús del *WINDOWMAKER*.



<i>Title Bar Text</i>	Permite definir el texto cabecera de <i>WINDOWMAKER</i>
<i>Show Application Directory</i>	Permite definir si queremos o no ver el nombre de la aplicación que estamos creando
<i>Spacing</i>	Pixels entre los puntos de la grilla
<i>Show Grid</i>	Ver o no la grilla
<i>Start Wonderware Logger</i>	Permite definir si queremos activar el programa WWLogger al activar <i>WINDOWMAKER</i>
<i>Show Tag Count</i>	Mostrar el número de tags usados
<i>Close on Transfer to WindowViewer</i>	Permite cerrar <i>WINDOWMAKER</i> al transferir a <i>WINDOWVIEWER</i>
<i>Enable Scrapbook Menu Items</i>	Permite disponer de los menús de trasvase de gráficos con el programa <i>SCRAPBOOK</i> (ya no se usa)
<i>Pick Through Hollow Objects</i>	Al mover un objeto, no ver más que su silueta
<i>Enable fast Switch</i>	Permitir cambio rápido de Maker a Viewer
<i>Line Selection Precision</i>	Precisión en la selección de líneas
<i>Levels of Undo</i>	Niveles de undo

Configurar WindowViewer

Disponemos de 3 pestañas:

GENERAL

Permite configurar algunas funciones avanzadas de *WINDOWVIEWER*, tales como memoria, inicio de *WWLogger* etc.

WINDOW CONFIGURATION

Permite configurar diversos elementos relacionados con el *WINDOWVIEWER*, tales como controles de menús, título, etc.

HOME WINDOWS

Permite configurar qué ventana será la primera en aparecer al arrancar *WINDOW VIEW*.

Configurar Alarms

GENERAL

Esta función permite configurar algunos parámetros referentes a la captación de alarmas.

LOGGING

Permite configurar la grabación y archivo de las alarmas. Podemos definir el directorio de grabación, los días que vamos a mantener los ficheros en disco (0 para guardarlas indefinidamente), grupos de alarmas a grabar y prioridad para la grabación

PRINTING

Permite configurar puerto de impresión de las alarmas, formato de impresión, grupo y prioridad

Configurar HistoricalLogging

Permite configurar directorio y días que se van a guardar los archivos correspondientes a aquellos tagnames que han sido configurados como *logged*.

Distributed Name Manager

Únicamente para sistemas de alarmas e históricos distribuidos en red. Permite definir de qué nodo nos van a llegar las alarmas / históricos distribuidos en red.

Install Wizards

Permite instalar nuevos Wizards y ActiveX



EJERCICIO 14. APARIENCIA DE LAS VENTANAS EN RUNTIME

- Este ejercicio nos enseña cómo modificar la apariencia de las pantallas runtime, así como los sistemas para proteger y evitar salidas accidentales del mismo.

Desde el menú *SPECIAL* → *CONFIGURE* → *WindowViewer* → *Window Configuration*. El cuadro de diálogo que aparece a continuación le permite modificar el aspecto final de la pantalla runtime

