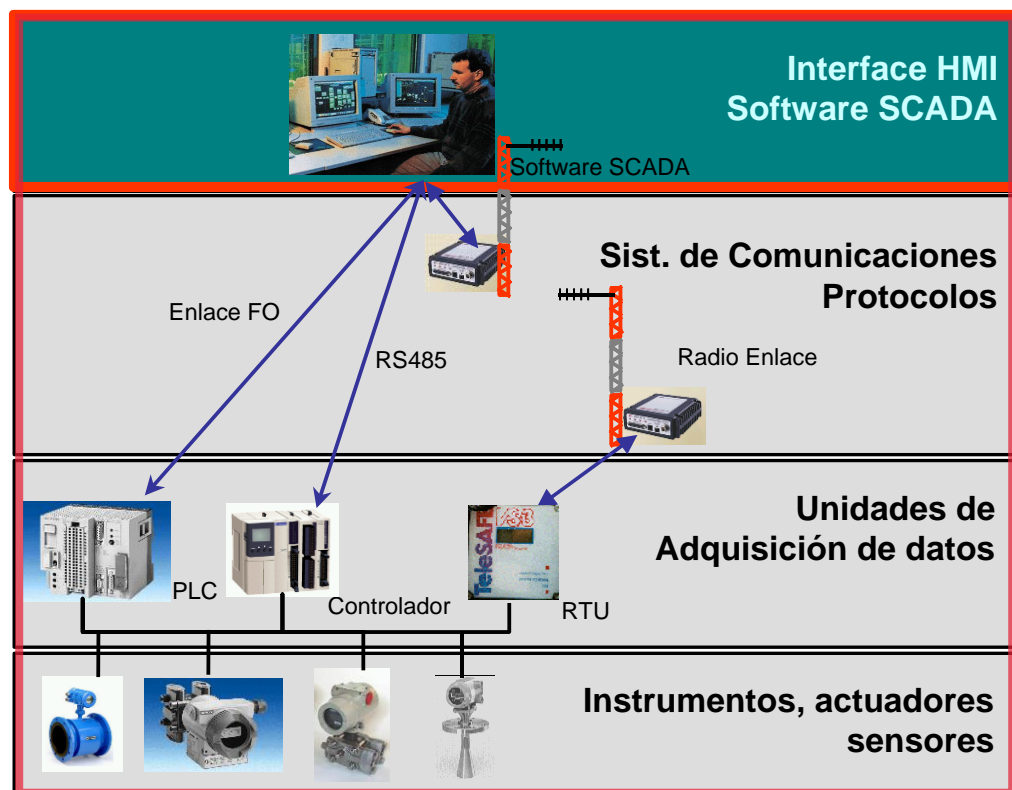


1. SOFTWARE SCADA (Supervisory Control And Data Acquisition)

Una de las partes más importantes de los sistemas de automatización, lo constituye el subsistema de **visualización de los datos**, lo que se conoce como Interfase Hombre- Máquina (MMI) o Interfase Humano – Máquina (HMI), también se denomina Software SCADA. Este software forma parte del último nivel de supervisión.

El software SCADA o HMI está formado por diferentes software's o programas que corren en una computadora y cuyo objetivo es el de visualizar todos los datos que se miden en la planta o en el campo de manera amigable y que permita el control de la misma de manera simple y efectiva.

Este software, en las arquitecturas tradicionales, utiliza un modelo Maestro – Esclavo con una arquitectura Punto – Multipunto en donde el software SCADA es el Maestro que realiza la consulta sobre varios dispositivos de campo como ser RTU's, PLC's, Controladores PID, etc. Estos a su vez tienen un esquema similar con los instrumentos de medición y actuadores.



En las aplicaciones más modernas se utiliza el modelo **Cliente – Servidor**, en general para arquitecturas con más de un nodo de supervisión. En estos casos varios sistemas supervisores pueden generar diferentes requisiciones de información a múltiples servidores de datos, la administración de esta información se realiza a través de modelos de accesos por detecciones de colisiones como las utilizadas en Ethernet.

Estos programas poseen varios componentes que permiten además de visualizar los datos, establecer alarmas, visualizar tendencias de las variables medidas, comunicarse con los dispositivos de campos, generar datos históricos y otras funciones para cumplir con el objetivo. En el siguiente esquema puede observarse algunos de los principales módulos de software SCADA.

Estos soft's pueden conformar un solo paquete o también pueden venir integrados con el sistema de control, pero en cualquiera de los casos las funcionalidades son similares.

En general estos programas realizan la comunicación con los diferentes dispositivos de adquisición de datos y control a través de comunicaciones digitales, dicha información es almacenada en una base de datos de tiempo real. Esta base de datos de tiempo real brinda información a los otros componentes para representar gráficamente los datos, para generar alarmas y gráficos. Desde la visualización también se permite enviar órdenes o comandos de manera de interactuar con el proceso



Existen diferentes productos comerciales, de los cuales los más difundidos son los basados en sistema operativo Windows.

También existen Software SCADA's basados en otros sistemas operativos como LINUX, UNIX o QNX, siendo este último un sistema operativo especialmente diseñado para los sistemas de adquisición de datos y supervisión en tiempo real.

Dentro de los softwares basados en Sistema Operativo Windows, los más comunes son el iFIX de Intellution e InTouch de Wonderware. En la actualidad ambos sistemas se comercializan como parte de un conjunto de programas o suite que incluyen otros productos auxiliares a la supervisión de procesos.

Otros software comerciales de este tipo son: Winlog (Windows), Realflex (QNX), Factory Link (UNIX y Win), Genesis (Win) y PCIM (Win).

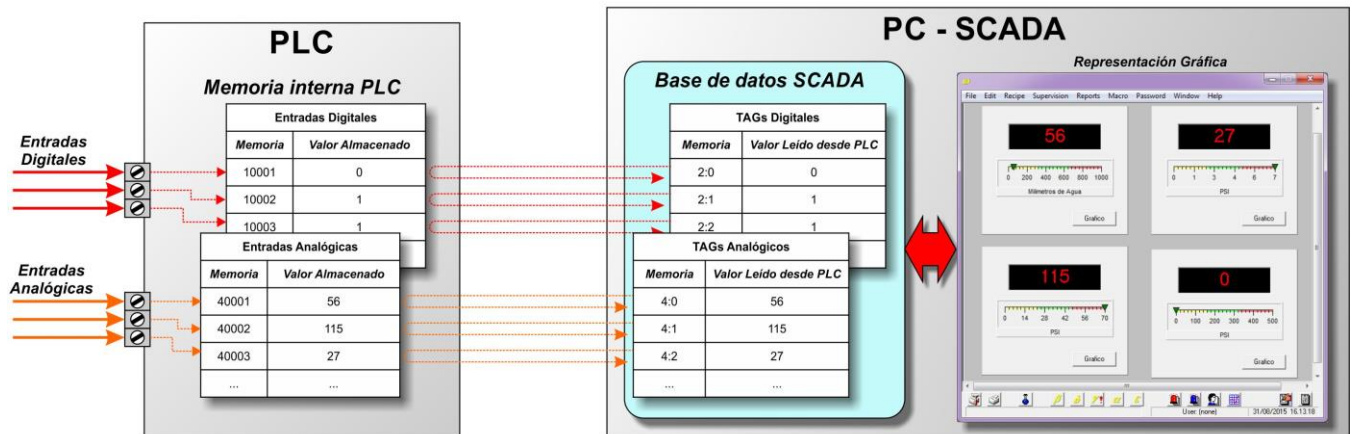
En lo referido los sistemas integrados estos generalmente se encuentran integrados con las herramientas de programación del sistema de Control, algunos ejemplos de ellos son el Delta V (Emerson), Industrial IT (ABB) y ARSView (Rockwell).

2. BASES DE DATOS SCADA

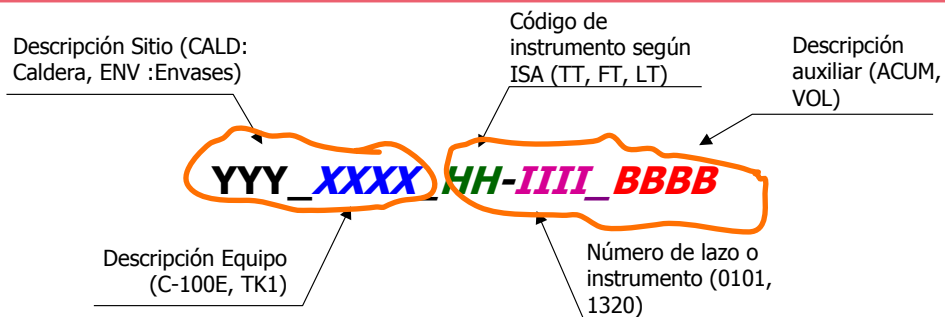
El intercambio de información se generaliza bajo el concepto de Base de Datos (DB). Se trata de un fichero de datos en el cual se realizan consultas y actualizaciones. Se puede agregar, extraer, actualizar o realizar operaciones diversas por medio de un lenguaje adecuado.

En general las bases de datos elementales se pueden considerar como una tabla formada por varias filas y columnas donde cada fila corresponde a un registro y cada columna a una característica de ese registro, un ejemplo sería una tabla con los Datos de los Estudiantes inscriptos en una materia donde se indican el Apellido, Nombres, Dirección, Edad, Teléfono y Nº de legajo

En el caso de la base de datos SCADA, la misma puede considerarse de manera similar solo que en las filas en lugar de ser Alumnos serán Variables de la planta o proceso y las columnas representan diferentes características de esa variable.



Para identificar cada variable de una manera organizada e inequívoca se utilizan lo que se denomina "TAG", cuya identificación está normalizada por la Sociedad Americana de Instrumentación ISA (Instrument Society of America). Es un estándar internacional, que se adecuan generalmente a las diferentes aplicaciones, la cual indica como se debe representar cada TAG de acuerdo a la variable que representa, asociado a este TAG se encuentra una descripción de la variable. Un ejemplo de la denominación de cada tag se podría realizar con la siguiente estructura:



Ejemplo

TKS_TK101_LT-1201_VOL

Un resumen de la normativa ISA para la denominación de instrumentos puede observarse a continuación:

	Encabezado Variable Medida Status	Modificador	Funcion Instrumento Funcion de la Variable	Modificador
A	Aire de Instrumentos	Automático	Alarma	Arranque
B	Llama o Quemador o Caldera			
C	Comunicaciones	Compresor	Control	Close
D	Densidad o Peso Específico	Diferencial	Fecha (DD)	
E	Tensión		Elemento Primario	

F	Caudal			
G		Gas		
H		Manual	Hora (HH)	Alto
I	Corriente		Indicador	
J				
K	Tiempo			
L	Nivel	Fluido	Luz, Piloto, Led Indocador	Bajo
M		Motor		Manual/Auto
N				
O		Petróleo	Orificio	Open
P	Presión o Vacío			Paro
Q	Contador de Eventos	Acumulado		
R			Registro	
S	Velocidad o Frecuencia	Bomba	Switch or Status	
T	Temperatura		Transmisor	
U	Multivariable		Multifunción	Multifunción
V			Válvula	
W	Peso o Fuerza	Agua	Vaina	
X	Comando		Contactor	
Y	A DEFINIR		Relay	
Z	Posición		Elemento Final de Control / Clasificar	

Las otras características importantes para la definición de cada registro son:

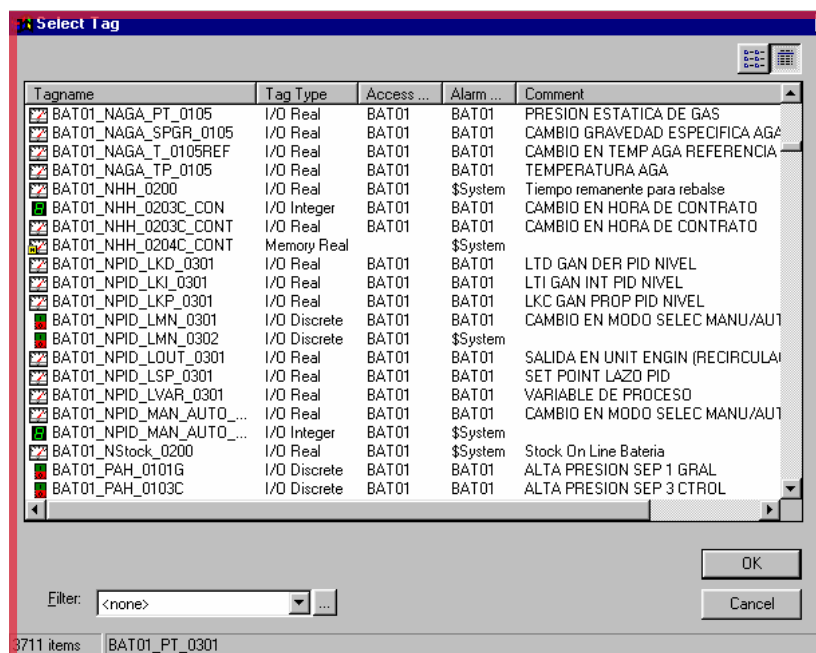
- Tipo de variable (Analógica o Discreta, de entrada o salida, interna o externa)
- Rangos de las Variables Analógicas (valores mínimos y máximos)
- Estados de las variables Discretas (cual estado corresponde a ON y cual a OFF)
- Valores de Alarmas (máximos, mínimos)
- Unidades
- Identificación del dispositivo de campo al cual pertenece la variable externa

A continuación puede observarse dos ejemplos de base de datos en formato de planilla, muy utilizado en la confección de la ingeniería de automatización.

TIPO	TAG	DESCRIPCION	Units	Min EU	Max EU	Min Raw	Max Raw	LoLo Alarm	Lo Alarm	Hi Alarm	HiHi Alarm	Esclavo	Dir
IOAnalog	LT-102_TK1	Nivel Tanque 1	mm	0	1500	0	1500	50	100	1200	1400	4	40001
IOAnalog	LT-103_TK2	Nivel Tanque 2	mm	0	1500	0	1500	50	100	1200	1300	4	40002
IOAnalog	PT_101_BBA	Presión descarga Bomba	bar	0	1.5	0	1500	-	-	1.3	1.4	4	40003
IOAnalog	FT_101_BBA	Caudal Bomba recirculación	lts/min	0	50	0	99999	0	-	-	-	40	40004

TIPO	TAG	DESCRIPCION	OffMsg	OnMsg	ESTADO ALRM	Esclavo	Dir
IODisc	XS_101_STATUS	Estado Bomba de recirculación	Parada	Marcha	None	4	31
IODisc	LSH_102_TK1	Alto nivel Tanque 1	Normal	Alto	ON	4	32
IODisc	PSL_202_AIRE	Baja presión de aire	Normal	Bajo	ON	4	33

Se muestra a continuación una pantalla de configuración de la base de datos del software SCADA comercial InTouch, de Wonderware, la misma se denomina Tagname Dictionary. Este tipo de estructura es compartida por softwares SCADA de otros fabricantes, poseen una base de TAGs, donde se definen sus características, entre ellas que tipo de variable tienen asociada, si son de lectura o escritura, etc.



Se puede seleccionar el TAG a través de una planilla donde se indica el nombre, el tipo, a que dispositivo se halla vinculado y el comentario o descripción del mismo

Al seleccionar un determinado TAG, se abre una pantalla donde se pueden definir nuevos TAG's o modificar los existentes y lo referido a las características de cada uno.

En esta instancia se definen:

Tipo: Si se trata de una variable interna (Memory) o con conexión a campo (I/O) y Se trata de una variable Discreta, Real (con punto flotante), Entera (Integer) o de Texto (Message). También se define si la variable es de lectura solamente (Read Only) o Lectura y Escritura (Read/Write)

TAG: Denominación del Tag y su descripción extensa o Comentario (Comment)

Unidad: Unidad de ingeniería

Rangos: Se definen los rangos del equipo de adquisición, los cuales podrán estar en unidades de ingeniería en cuenta o alguna otra representación, también se indican los Rangos en Unidades de Ingeniería. En caso que sea necesario realizar una conversión de unidades el software puede realizar la misma mediante el uso de los rangos. Por ejemplo si el equipo transmite directamente valores de presión estos pueden ser representados sin

necesidad de conversiones, pero si transmite 4-20 mA es necesario indicar que valores corresponden para 4mA y para 20 mA (por ejemplo 4mA → 0 PSI, 20 mA → 100 PSI).

Comunicación : Se definen los dispositivos con los que se establecerá la comunicación y la dirección de los mismos.

Alarmas : Se definen si están habilitados los diferentes niveles de alarmas y cuales son los valores y prioridades para cada caso.

Variación: Se definen cuando se notificará que una variable se ha modificado, lo que puede ser a través de superar un nivel de banda muerta (deadband) o a través de la velocidad con la que cambia la variable (ROC :Rate Of Change).

Históricos y Eventos: Se definen cuando está habilitada la generación de archivos históricos y/o eventos de la variable y ante que variación se va a registrar un valor histórico (Log Dead Band), en el caso de los eventos se registran todos los cambios, estos generalmente se usan para variables discretas.

Por ejemplo en el SCADA InTouch:

The screenshot shows the 'Tagname Dictionary' dialog box in InTouch. The 'Details & Alarms' tab is selected. The 'Tagname' field contains 'BAT06_LT-0202G'. The 'Type' is set to 'I/O Real'. The 'Group' is 'BATERIA06'. The 'Comment' is 'Nivel Tanque 2 General'. The 'Log Data' checkbox is checked. The 'Initial Value' is 7. The 'Deadband' is 0. The 'Eng Units' are 'cm'. The 'Access Name' is 'BAT06'. The 'Item' is '40282'. The 'Log Deadband' is 0.5. The 'Alarm Value' table shows four levels: LoLo (40), Low (50), High (160), and HiHi (180), all with priority 1. The 'Value Deadband' is 3. The 'Rate of Change' checkbox is checked. The 'Conversion' is set to 'Linear'. The 'Minor Deviation' and 'Major Deviation' checkboxes are unchecked. The 'Deviation Deadband %' is 10.

Annotations in the image include:

- TAG**: Points to the 'Tagname' field.
- Descripción de la variable**: Points to the 'Comment' field.
- Realización de Históricos de esta variable**: Points to the 'Log Data' checkbox.
- Unidades**: Points to the 'Eng Units' field.
- Dirección en RTU, PLC o Unidad de Adquisición de datos**: Points to the 'Access Name' field.
- Valores de Alarma**: Points to the 'Alarm Value' table. A legend below it specifies: LoLo : Muy Bajo, Low :Bajo, High : Alto, HiHi : Muy Alto.
- Tipo de la variable**: Points to the 'Type' field. A legend below it specifies: I/O : Externa, Real : Formato.
- Valores max y min**: Points to the 'Min EU' and 'Max EU' fields. A legend below it specifies: Raw : en PLC o RTU, EU : valores variable.
- Identificación de RTU, PLC o Unidad de Adquisición de datos**: Points to the 'Item' field.
- Variación para generar un valor histórico**: Points to the 'Log Deadband' field.
- Variación necesaria para que sea visualizada**: Points to the 'Deviation Deadband %' field.
- Análisis de la velocidad de variación para que sea visualizada**: Points to the 'Rate of Change' checkbox.

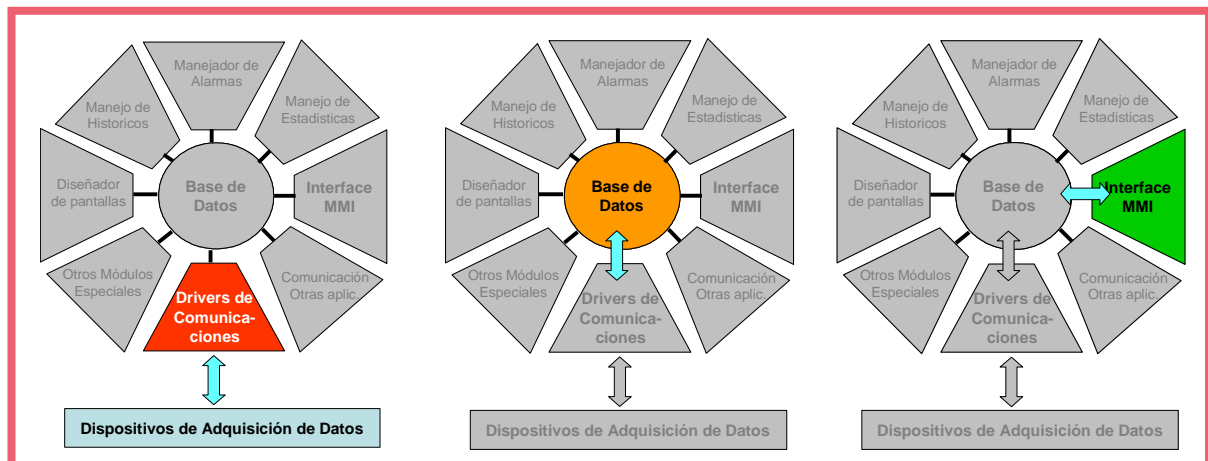
El funcionamiento de la base SCADA, generalmente es a través de un sistema cliente – servidor, donde esta base de datos funciona tanto como cliente, pidiendo datos a los módulos de comunicaciones y como servidor de datos para los módulos de visualización, alarmas e históricos.

En el siguiente esquema se ha considerado un caso básico donde solamente se realiza la visualización de una variable, en este caso se producen 3 proceso en forma paralela, por un lado los módulos de comunicación le solicitan, con una determinada frecuencia, información a los dispositivos de adquisición de datos. Dicha frecuencia se establece en la configuración de cada driver de comunicación. Una vez recepcionados los datos, los mismos son interpretados

por los drivers y puestos a disposición de algún cliente de datos como el manejador de la base de datos.

Por otro lado el manejador de la Base de Datos toma los datos disponibles de los drivers y realiza la conversión de los mismo, y los pone a disposición de algún cliente de datos como ser el módulo de visualización, alarmas o históricos.

En forma paralela el módulo de visualización lee los datos disponibles en la base de datos y los presenta gráficamente en la pantalla del operador.



COMUNICACIONES CON ELEMENTOS DE ADQUISICION DE DATOS Y AUTOMATIZACIÓN

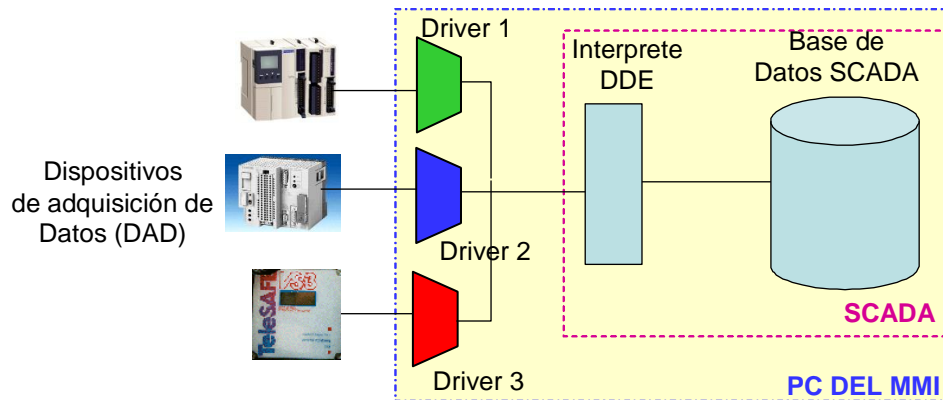
La comunicación con los dispositivos de campo, los cuales a su vez adquieren datos del proceso y realizan control del mismo, es otro de los componentes fundamentales del software de supervisión, ya que es a través de este componente donde se nutren el SCADA de información en tiempo real.

En estas comunicaciones, como en cualquier otra, dos aspectos son fundamentales: el medio físico y el lenguaje o protocolo a utilizar. Dado que los tipos de Dispositivos de Adquisición de Datos o DAD (RTU, PLC, Controlador, etc.) son muy variados y los protocolos que ellos pueden manejar son muchos, se utilizan tres modelos.

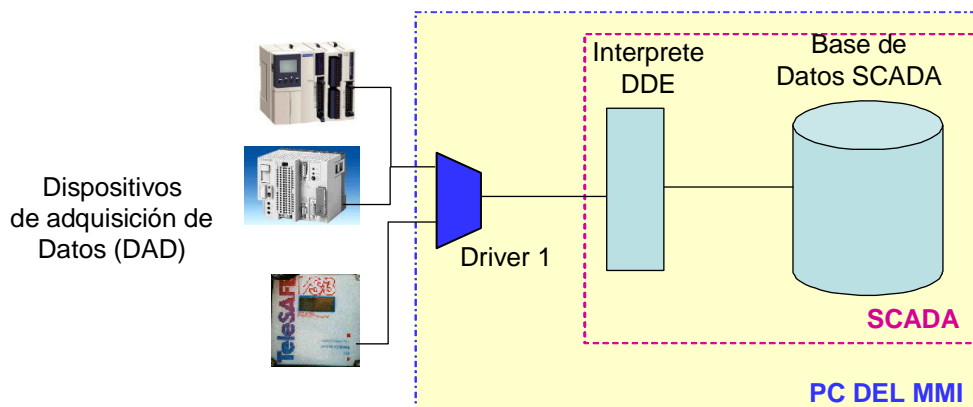
- El más antiguo de ellos se basa en utilizar un intérprete, el cual habla el lenguaje del Dispositivos de Adquisición de Datos o DAD y lo interpreta, convierte la información y la transfiere a la base de datos del SCADA, estos interpretes son denominados comúnmente DRIVERS y se requiere uno por cada tipo diferente de protocolo.
- El otro modelo es utilizar un lenguaje común y estándar, forzando a los fabricantes de dispositivos de adquisición de datos a tener un intérprete de ese lenguaje común o protocolo estándar dentro de su dispositivo, como sería el caso del protocolo MODBUS (la utilización de un lenguaje estándar es solo posible si el dispositivo al que se va a acceder lo soporta).
- Una variación a este último modelo lo constituye los Drivers OPC (OLE for Process Control) donde se utiliza la tecnología OLE (Object Linking and Embedding) de Windows.

DRIVERS

Tal como se mencionara anteriormente el concepto del DRIVER, es el de un intérprete y por lo tanto se requiere de uno por cada tipo diferente de protocolo o lenguaje, en un inicio existían tantos protocolos como dispositivos de adquisición de datos, lo cual complicaba el desarrollo de los software SCADA ya que debían desarrollar un driver por cada dispositivo que se creaba.



La dificultad de poder mantener esto en el tiempo llevó tanto a los fabricantes de equipos como a los de software SCADA a unificar el uso de algunos drivers, en donde sobresale el MODBUS por su simpleza, su amplia difusión y por tratarse de un protocolo abierto disponible a cualquier usuario. La desventaja de esto es que la potencialidad de comunicación del DAD se ve limitada por las características del protocolo.

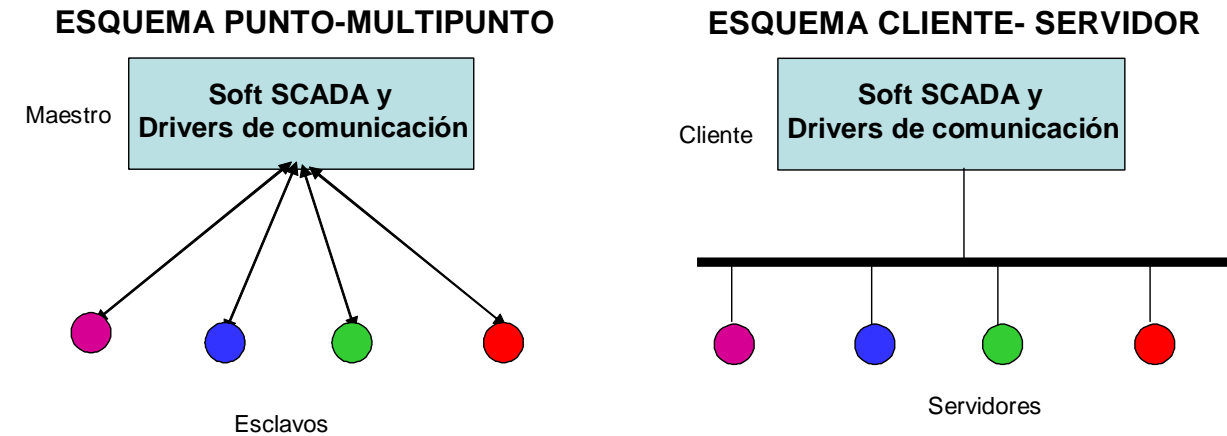


En general en la mayoría de los drivers se utiliza el método de direccionamiento a través de identificación de dispositivo mediante un N° de dispositivo y de una identificación del dato a través de un registro.

Los drivers, además de generar la conversión de la información que se envía y se recibe de los dispositivos de campo, informan sobre el estado de estas comunicaciones y en algunos casos de las estadísticas de los mismos (si la comunicación se ha interrumpido, cuantos datos se han intercambiado, etc).

Existen drivers que funcionan solo bajo el modelo de Punto – Multipunto con el sistema de encuesta o Polling basado en tiempo (por Ejemplo MODBUS) y otros que permiten su utilización tanto en modo Polling o por Excepción, es decir los dispositivos esclavos puede enviar mensaje en caso de alarmas o eventos importantes (por ejemplo ROC, MODSCAD,

BISAP). Por otro lado también hay drivers que utilizan modelo Cliente – Servidor (DATA HIGHWAY+)



OPC SERVERS

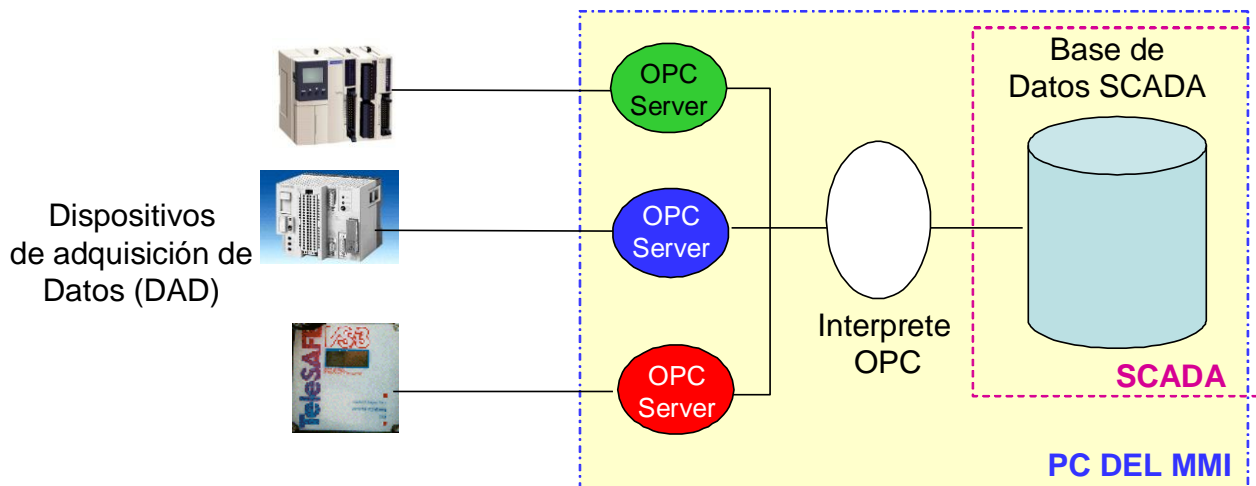
Aprovechando una propiedad definida como estándar de los objetos de los sistemas operativos Windows denominada **OLE (Object Link Embbeded)** que permite incrustar o embeber una aplicación dentro de otra, como por ejemplo cuando incluimos una tabla Excel en un documento Word, se desarrollo un nuevo estándar denominado OPE (OLE for Proccess Control) el cual permite incluir objetos de comunicaciones en aplicaciones de sistema operativo Windows. Esto permite tener un set de comandos y repuestas estándar que permite que dos aplicaciones puedan entenderse sin inconvenientes.

La diferencia entre el uso de OPC y el uso de drivers como intérpretes de protocolo, es que la comunicación se produce en un lenguaje estándar abierto y a nivel de aplicaciones, de una manera similar al DDE.

De esta manera los fabricantes de cada dispositivo desarrollan un Servidor OPC, el cual por un lado se comunica con su dispositivo y por el otro es capaz de responder o recibir información en un formato estándar OPC. Los fabricantes de software de visualización por su lado desarrollan in interpretador de OPC que se comunicará con el Servidor OPC.

La ventaja de este método es que permite, con un solo interpretador de OPC, leer cualquier dispositivo que tenga asociado un OPC Server, con lo cual toda actualización o incorporación de nuevos dispositivos será fácil y se ampliara el ciclo de vida de los paquetes.

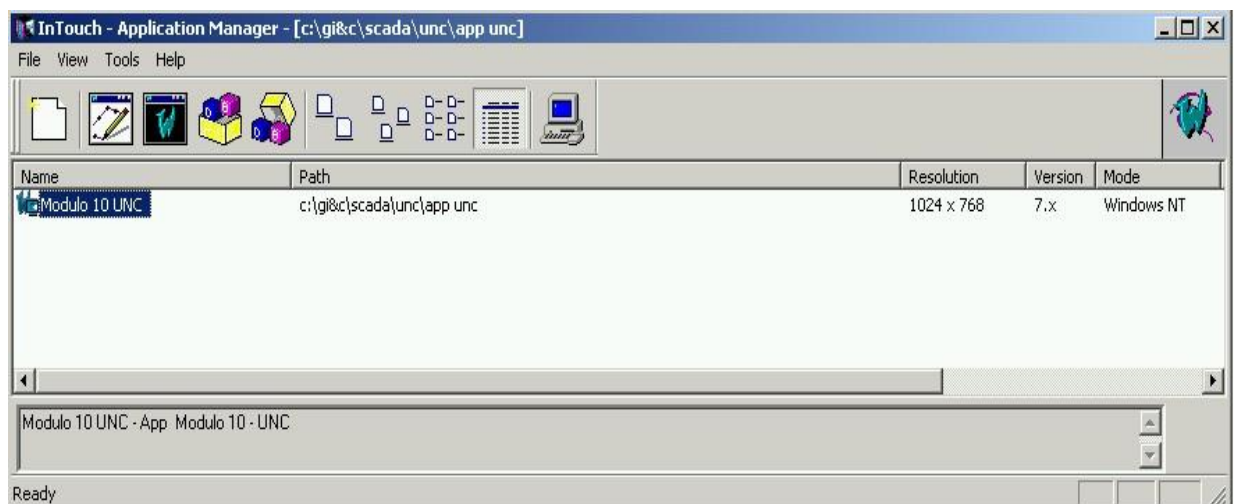
Otra ventaja importante es que al estar los OPC Servers diseñados por los fabricantes estos permiten acceder a todos los registros de los dispositivos con lo cual se gana potencialidad en la comunicación sin perder estandarización.



3. DISEÑO DE APLICACIONES

Otro componente fundamental de un software SCADA, además de los componentes de base de datos y el de comunicación es sin duda el **módulo de Visualización**, dicho componente generalmente se encuentra formado por dos aplicaciones de software, una que permite el diseño de las pantallas a visualizar y el otro que realiza la visualización tomando los datos de la base de datos, drivers y manejando las alarmas, reportes, históricos, etc.

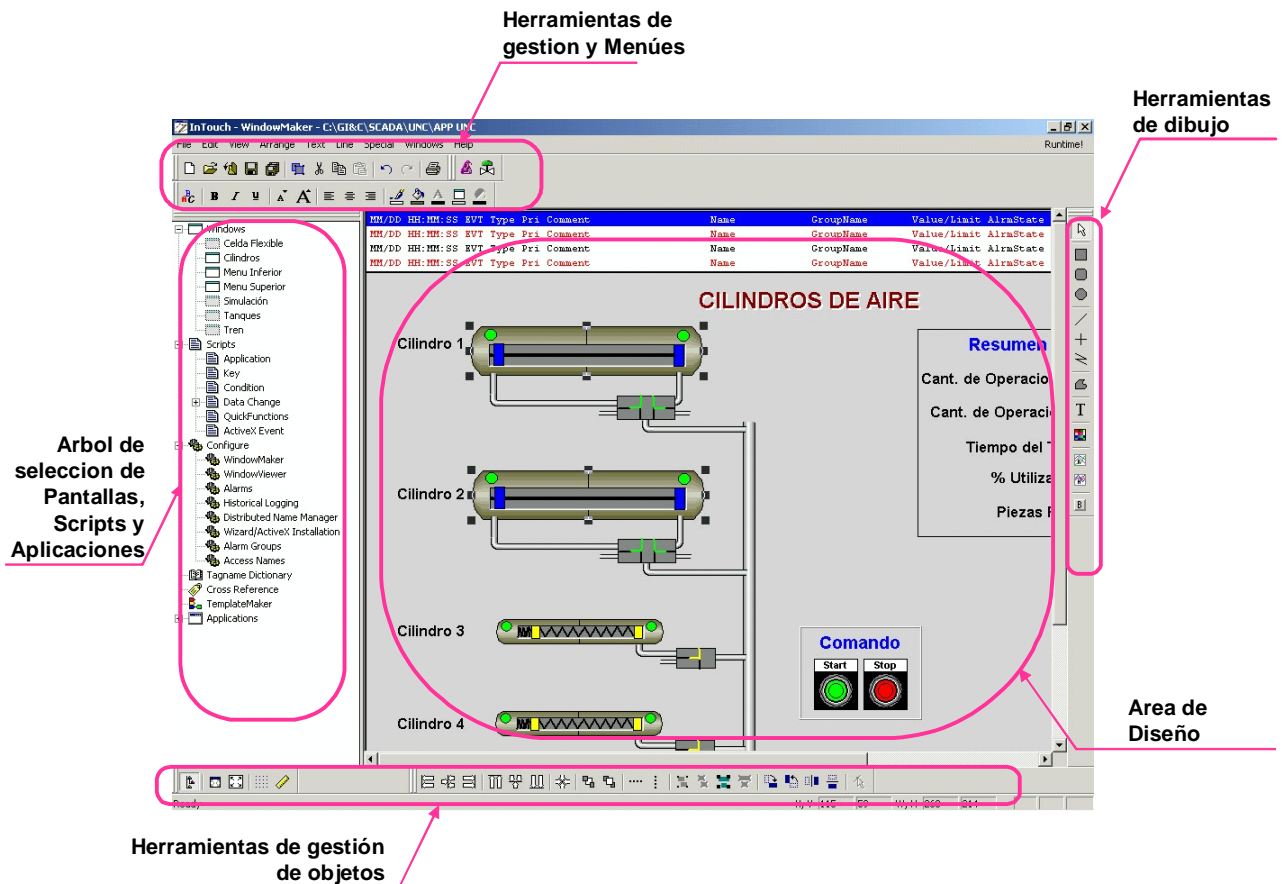
Por ejemplo en el paquete de InTouch, existe un programa denominado **INTOUCH** que permite administrar varias aplicaciones y desde allí se puede acceder al módulo de visualización que se denomina **VIEW** (es quien muestra gráficamente el estado del sistema al Usuario) y al módulo de diseño de pantallas se denomina **MAKER** (este módulo es empleado por el desarrollador para diseñar la pantalla/s que verá el usuario).



DISEÑO GRAFICO DE APLICACIONES

El proceso de diseño de pantallas se puede dividir en las siguientes etapas:

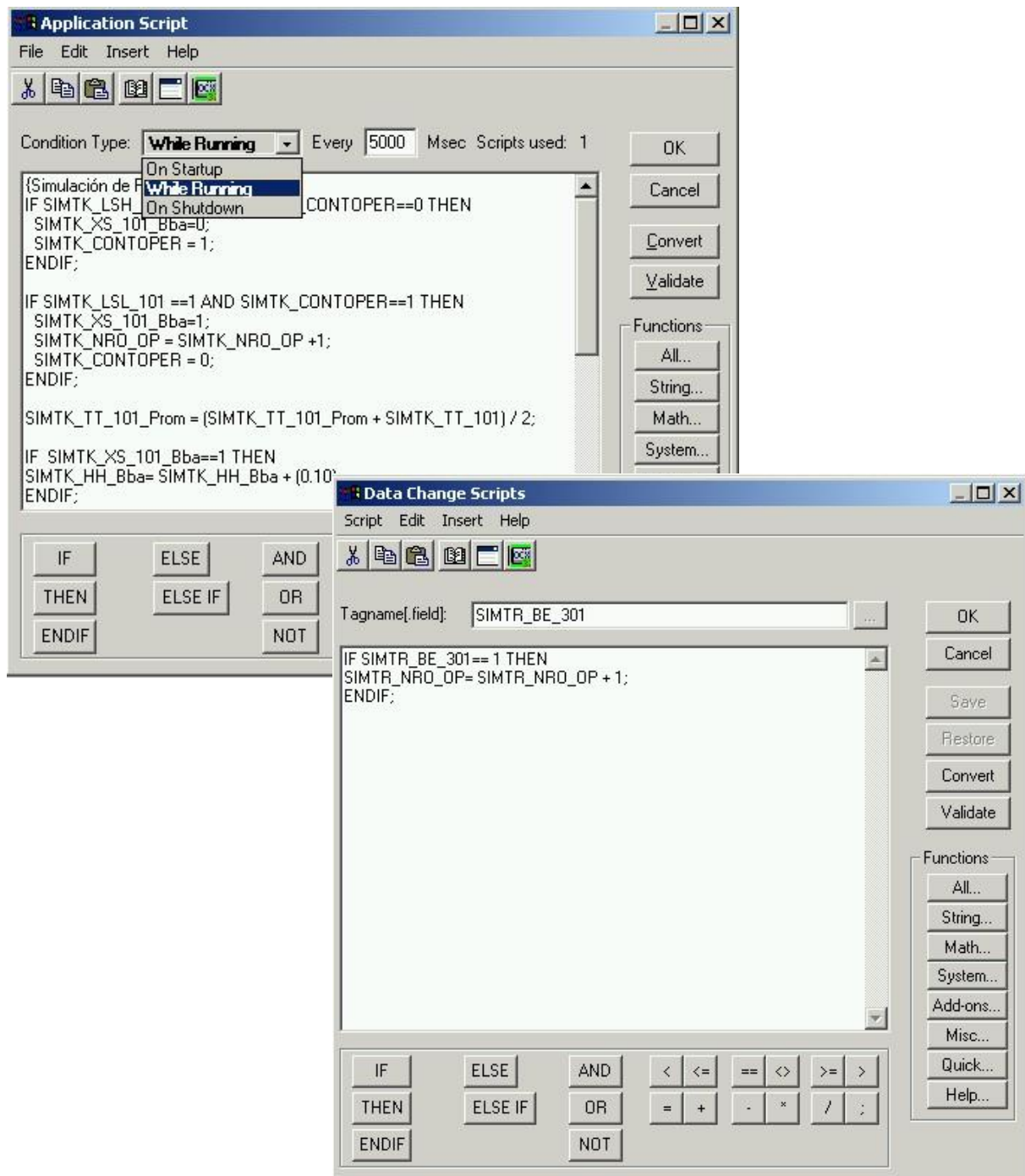
- ✓ **Diseño gráfico de las pantallas:** Utilizando objetos y formas predefinidas e importando imágenes u objetos de otras aplicaciones se puede armar un esquema que representa al proceso que se desea monitorear o visualizar.



- ✓ **Animación de pantallas:** Cada objeto o elemento incluido en la pantalla puede ser animado en función de alguna variable, el tipo de animación dependerá del tipo de objeto, por ejemplo un rectángulo tendrá como posibilidades de animarse cambiando de color de llenado o del las líneas que lo componen, o podrá cambiar de posición o de tamaño. En cambio un texto podrá cambiar de color o representar una variable. En base a que animación se le puede dar a cada objeto dicha animación se puede asociar a una variable o Tag, de manera que cuando este cambie su valor o estado el objeto realice la animación configurada, teniendo la impresión que lo que se representa en la pantalla es una representación del proceso o la planta animada.
- ✓ **Vinculación de pantallas:** La representación de un proceso puede requerir varias pantallas (por claridad no siempre es posible ilustrar todo un proceso en una sola pantalla). Una vez desarrolladas las pantallas se deben vincular unas con otras, para lo cual se puede generar un menú para el acceso rápidos de las pantallas principales o a través de botones de enlace de una pantalla a otra. Las pantallas principales generalmente se realizan en modo que se reemplazan unas a otra mientras que las pantallas secundarias suelen ser del tipo Popup, que significa que aparecen arriba de otras pantallas.

PROGRAMACION A TRAVES DE SCRIPTS

Además de realizar animaciones de las pantallas en función de las variables de campo o planta, los software SCADA permiten programar lógicas o acciones a través de un lenguaje propio de tipo estructurado, con funciones predefinidas. Dichas funciones permiten operaciones matemáticas, trigonométricas, iteraciones, bifurcaciones, comparaciones, operaciones de strings, asignaciones y otras funciones propias de cada fabricante, como ser: Manejo de pantallas, Manejo de históricos, Manejo de sonidos, gestión de alarmas, etc.

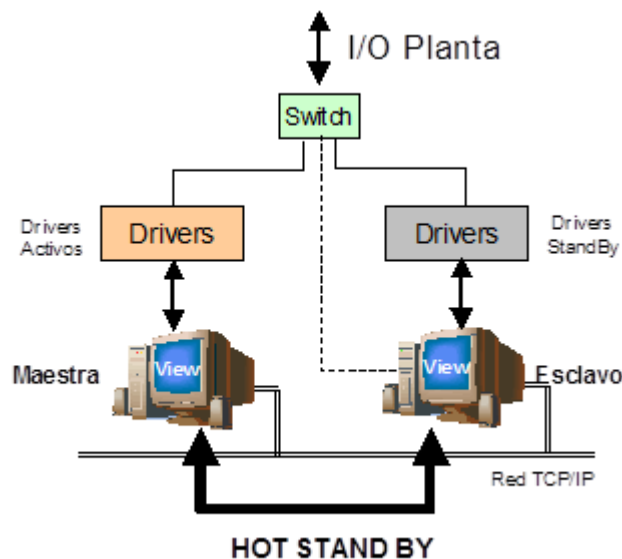


ARQUITECTURAS DE APLICACIONES

Si bien hay muchas aplicaciones que corren en una sola PC y en forma independiente (Stand Alone), existen otras configuraciones muy comunes donde la información de Planta surge de los diferentes sectores de la misma e interesa operativamente a ese sector, pero algunos datos se comparten entre los diferentes sectores o islas de automatización.

Otra situación muy común es que la información no solo es monitorearse desde donde se reciben los datos sino que desea monitorearse desde otro punto sin realizar ningún tipo de comando.

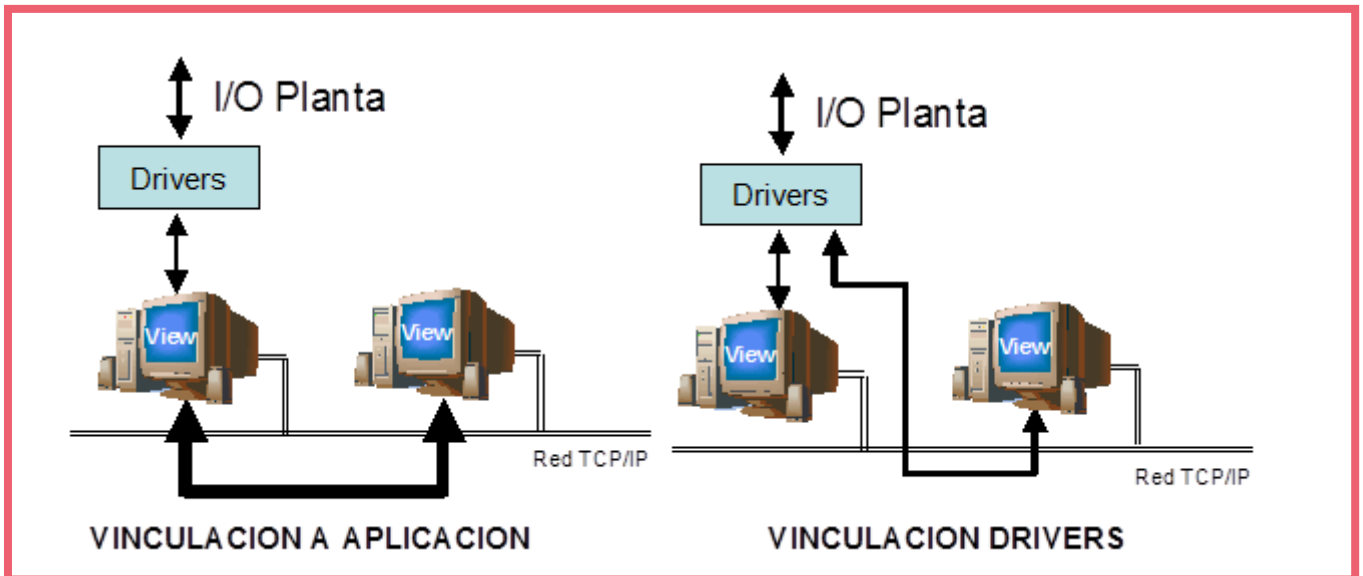
Por último otra situación se da cuando se desea tener una redundancia del software SCADA y que la misma se encuentre activa en caso que la estación Maestra tenga algún problema, dicha configuración es denominada Hot StandBy.



Es en estas situaciones donde se requiere que se relacionen las aplicaciones entre si, lo cual se puede hacer fácilmente utilizando una red Ethernet, la cual podrá ser del tipo dedicada o la misma red administrativa en cuyo caso habrá que utilizar seguridad de acceso para evitar malas operaciones o sabotajes.

En este modo de funcionamiento se pueden tener estos esquemas:

- ❑ Vinculación a nivel aplicaciones: Para esto se direccionan todas las variables o TAG's de una aplicación a la aplicación de visualización primaria que recibe los datos. La información es la misma de la que se ve en todas las aplicaciones pero en el caso que se detenga la aplicación View no se recibe más información.
- ❑ Vinculación a nivel de drivers de comunicación: En este caso el re direccionamiento se hace a nivel de los drivers de comunicación que se hallan corriendo en la máquina Primaria. En este caso si se detiene la aplicación de visualización pero los drivers están corriendo se sigue teniendo la información. El inconveniente es que se carga el tráfico de la red y el funcionamiento de los drivers, y además se pueden tener diferentes valores en una misma aplicación producto de la sincronización (si existe un retardo en la transmisión de datos, las distintas máquinas pueden mostrar un valor distinto para una misma variable).



En ambos casos se deberá definir el manejo de comandos, es decir si las aplicaciones remotas permiten comando o si son solo de visualización y el manejo de alarmas en cuanto al reconocimiento de las mismas.

4. ALARMAS E HISTORICOS

En los procesos actuales la cantidad de variables a monitorear y sobre las cuales se pueden tomar decisiones para controlar el proceso y mejorar la producción, o evitar paros de planta, es tan grande que sería casi imposible estar viendo a todas en forma permanente.

Con el mismo objetivo de realizar un mejor control del proceso y permitir análisis posteriores que permitan realizar acciones que mejoren el mismo, se encuentran los gráficos de tendencias y los gráficos históricos.

Ambas herramientas son parte fundamental de los sistemas de monitoreo como los soft SCADA ya que ayudan al control del proceso.

ALARMAS Y EVENTOS

En principio se definen dos tipos de situaciones, las alarmas y los eventos, siendo estos últimos una situación no común e importante que se desea distinguir y registrar.

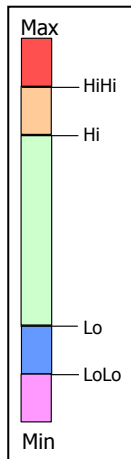
Conceptualmente una **alarma es un aviso que se ha producido una desviación de un parámetro normal**, bajo este mismo concepto los sistemas de monitoreo desarrollan un esquema para facilitar el manejo de las mismas.

En el caso de las alarmas se definen 3 estados en los cuales se puede encontrar una alarma:

- ☐ Sin alarma
- ☐ Alarma Nueva Sin Reconocer (unack alarm)
- ☐ Existente y Reconocida (ack alarm)

Cuando ocurre una alarma, el operador debe acusarla, indicando que ha visto la misma, a esto se denomina reconocer una alarma, una alarma sin reconocer implica que el operador todavía no se ha percatado de la misma.

Dependerá del tipo de variable los niveles a establecer, en general para los software SCADA en las variables analógicas se distinguen 4 niveles de alarmas y en las discretas o digitales solo dos niveles de alarmas. En las alarmas analógicas los niveles son:



- Muy Alto: cuando se encuentra por encima de un valor de referencia HiHi
- Alto: cuando se encuentra por encima de un valor de referencia Hi
- Bajo: cuando se encuentra por debajo de un valor de referencia Lo
- Muy Bajo: cuando se encuentra por debajo de un valor de referencia LoLo (Low Low)

Para mejorar el manejo de las alarmas se realizan generalmente dos acciones, la agrupación y la priorización de las mismas, de manera que cuando se analice un conjunto de alarmas puedan realizarse filtros por grupos o prioridad. En algunos casos las alarmas pueden asignarse no solo al cambio de un valor sino a la velocidad de cambio de una variable, a este tipo de alarmas se las denomina "ROC alarms" (Rate of Change Alarm).

En varios software SCADA la definición de la alarma se realiza en la definición del Tag. La visualización de la alarma se puede realizar de dos modos, en **forma gráfica** asociando alguna animación a la presencia o no de la alarma, o a través de un **panel de alarmas** o Banner de alarmas en donde se listan todas las alarmas presentes y el estado en que se encuentran.

Estos banners ya se encuentran predefinidos como objetos y pueden incorporarse en cualquier pantalla. En estos objetos se puede configurar cuáles son los datos a visualizar (Fecha, Hora, Descripción, Grupo, prioridad, Estado, Nivel de seteo de alarma, Valor), los colores que representan cada estado (Nueva, reconocida, retorno a normal)

Día	Hora	Estado	Tipo	Nombre	Grupo	Valor	Límite	Comentario
07/05/04	19:13:29	UNACK	DSC	FT_146_42R	Boleadoras	ON	ON	
07/05/04	19:13:20	UNACK	DSC	FI_016R	Boleadoras	ON	ON	
07/05/04	19:12:58	UNACK	DSC	FT_146_01R	Boleadoras	ON	ON	
07/05/04	19:12:03	UNACK	DSC	FT_146_62R	Boleadoras	ON	ON	
07/05/04	19:10:57	UNACK	DSC	FT_146_02R	Boleadoras	ON	ON	
07/05/04	19:09:39	UNACK	HIHI	LOY_Bbeo_EE02_RTU	ROC_BBEO	13.75...	12	
07/05/04	19:09:28	UNACK	DSC	FT_146_21R	Boleadoras	ON	ON	
07/05/04	19:08:36	ACK	DSC	FT_146_41R	Boleadoras	ON	ON	
07/05/04	19:08:36	ACK	DSC	LAL_146_01	CBO_Oeste	ON	ON	Bajo nivel petroleo SG-146-01
07/05/04	19:08:36	ACK	DSC	LAL_523	CBO_Sen_Media	ON	ON	Bajo nivel sen. control flash 4.

En la configuración general de alarmas se puede solicitar que las mismas sean registradas, de manera de poder ser analizadas con posterioridad, esto se realiza almacenándolas en un archivo de texto.

HISTORICOS Y TENDENCIAS

La graficación de la variación de las variables en el tiempo es una herramienta muy útil en el control de procesos ya que permiten anticiparse a futuros acontecimientos y tomar las decisiones correctas. De acuerdo a como se visualizan respecto al tiempo los mismos se denominan tendencias o históricos.

Las tendencias es un gráfico histórico de corta duración generalmente de 1 hora como máximo, que se va actualizando permanentemente o en tiempo real según la tasa de variación de la variable graficada.

Los gráficos históricos son similares, solo que el periodo de tiempo en el que se muestra suele ser más grande varias horas o días y no se actualizan en forma permanente sino que visualizan un periodo pasado. Estos gráficos generalmente cuentan con herramientas auxiliares para modificar tanto el periodo de tiempo (Zoom In y Zoom Out) como la escala de las variables (Max y Min).

Tanto las tendencias como los gráficos históricos generalmente pueden visualizar una variable o varias variables, llegando hasta 16 variables (pens) en un solo gráfico, esto permite analizar varias variables de interés en un solo gráfico y con una misma base de tiempo. Un ejemplo clásico de esto es los gráficos de tendencias utilizados para la sintonización de un lazo PID, donde en un mismo gráfico se muestran la variable de proceso (PV), el Setpoint (SP) y la salida del lazo (OUT) y se puede observar como responde el mismo a cambios en las variable de proceso, y el setpoint ajustando los diferentes parámetros del lazo.

Para la generación de históricos se debe habilitar esta opción en cada variable que se desea almacenar, esto es válido tanto para tendencias como para históricos. En algunos softwares SCCADA es posible también configurar la banda de variación de la variable por encima de la cual se genera un registro histórico, esta se denomina LOG DEAD BAND y está expresada en % de la variable en unidades de ingeniería.

Por ejemplo en el caso de InTouch la alarma se define de la siguiente manera:

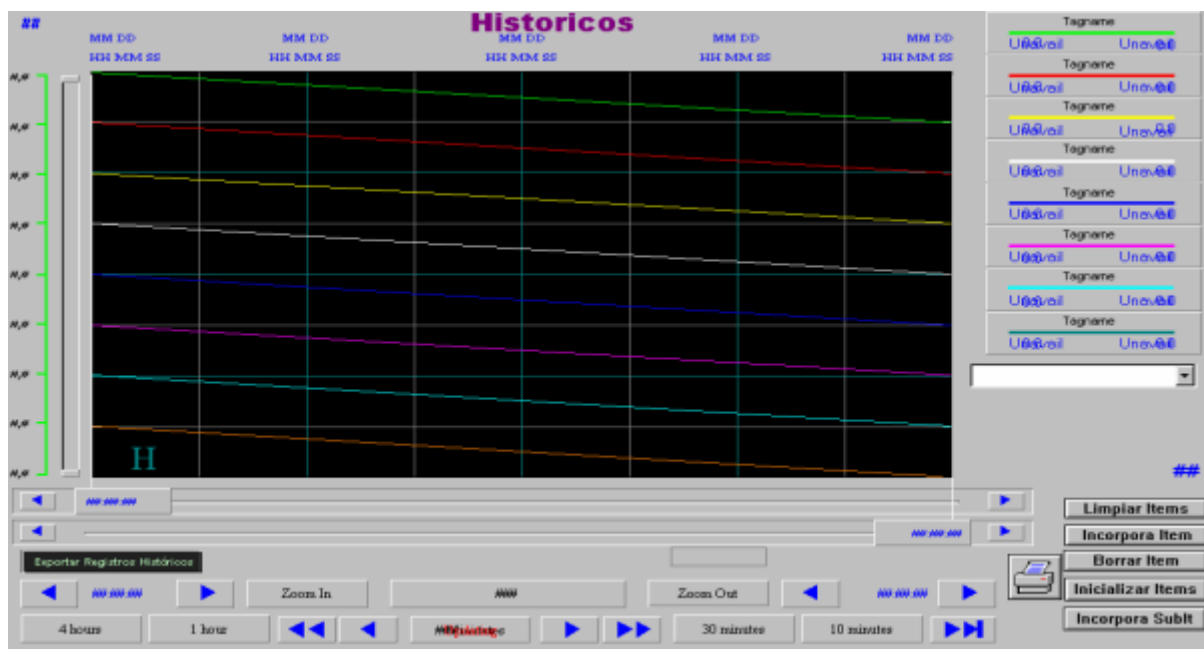
The screenshot shows the 'Tagname Dictionary' dialog box with the 'Details & Alarms' tab selected. The configuration for tag 'BAT06_LT-0202G' is as follows:

- Tagname: BAT06_LT-0202G
- Type: I/O Real
- Group: BATERIA06
- Comment: Nivel Tanque 2 General
- ☒ Log Data
- ☐ Log Events
- ☐ Retentive Value
- ☒ Retentive Parameters
- Initial Value: 7
- Min EU: 7
- Max EU: 251
- Deadband: 0
- Min Raw: 0
- Max Raw: 4095
- Eng Units: cm
- Conversion: ☒ Linear, ☐ Square Root
- Access Name: BAT06
- Item: 40282
- ☐ Use Tagname as Item Name
- Log Deadband: 0.5

Two red boxes with arrows point to specific settings:

- A box labeled 'Realización de Históricos de esta variable' points to the 'Log Data' checkbox.
- A box labeled 'Variación para generar un valor histórico' points to the 'Log Deadband' field.

Un ejemplo de visualización de Históricos:



5. REPORTES

Permiten disponer de un resumen en papel o en soporte magnético de todo el estado del sistema en un determinado momento para su posterior análisis.

En la realización de Reportes es muy importante antes de armar el formato y los datos que va a contener el mismo, definir a quien está dirigido para poder determinar cual dato conviene o no incluir. Se debe tomar como premisa que solo se debe colocar la información necesaria y útil.

Los datos pueden ser generados directamente en los dispositivos de adquisición de datos ya sea a través de cálculos o históricos o bien a través del sistema de supervisión, la primera opción tiene como ventaja que antes fallas de comunicación el reporte se podrá obtener una vez reestablecida la comunicación, la desventaja es que es más complejo realizarlos.

Existen diferentes tipos de reportes en relación a como son obtenidos los datos en función del tiempo, los mismos son:

REPORTE INSTANTANEO DE VARIABLES

Permite sacar una "foto" de la información adquirida en un determinado momento, el mismo se puede realizar imprimiendo una pantalla definida como reporte o exportando a un archivo gráfico una pantalla para luego ser impreso o almacenado.

La desventaja de este modelo es que solo se cuenta con la información de un instante sin saber como ha evolucionado

REPORTE HISTORICO

Permite incluir información dentro un periodo de tiempo. Se realiza un archivo con formato texto separado por comas (CSV) el cual va incluyendo registros cada un cierto periodo.

En este reporte generalmente se guardan la hora, fecha y los valores de las variables. Este almacenamiento debe ser programado dentro del software de supervisión.

Para su lectura se pueden usar alguna planilla de cálculo como Excel, donde a través de macros se puede realizar la importación de los datos y su formateo para la presentación.

-----000000000000-----