



CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS

CONTROLE DE PROCESSOS INDUSTRIAIS

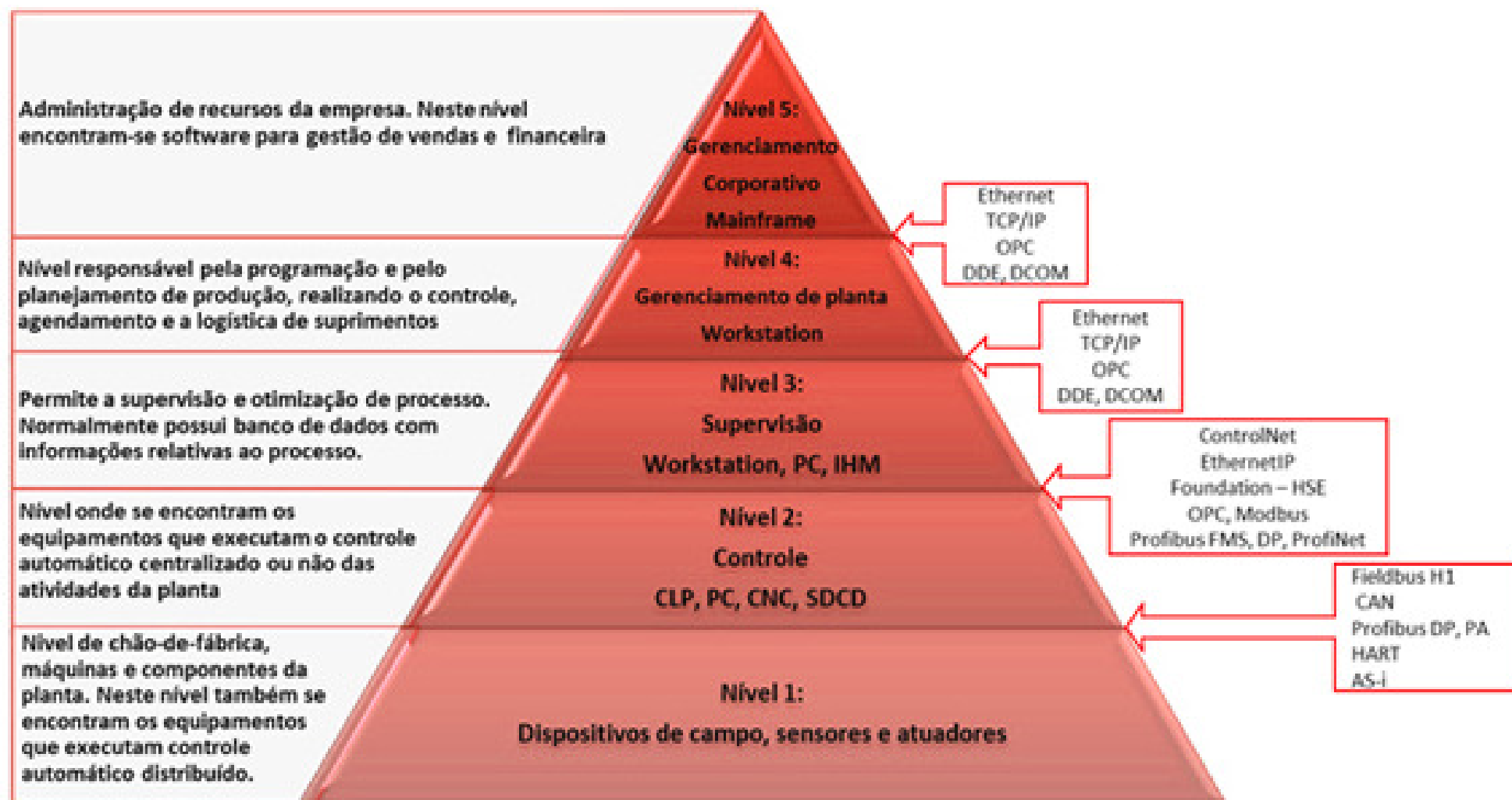
SISTEMAS SUPERVISÓRIOS - *SCADA*

Softwares Supervisórios

1.Introdução

Podemos definir software supervisorio como sendo um programa de computador capaz de monitorar e comandar dados de processos automatizados por meio da aquisição e do envio de informações vitais do sistema: **SCADA - Supervisory Control and Data Acquisition**.

Estas informações encontram-se nos equipamentos de automação (PLC's, Placas de aquisição, Controladores digitais, Inversores de Frequência, etc.). Estes dados são visualizados nos formatos de figuras, fotos, animações computacionais, gráficos, tabelas, relatórios, etc.



Os principais softwares disponíveis no mercado para supervisão e controle de processos:

- InTouch HMI – Schneider Electric
- Wonderware System Platform – Schneider Electric
- SIMATIC WinCC – Siemens
- Factory Talk – Rockwell
- RSView32 – Rockwell
- Cimplicity – General Electric
- iFIX – General Electric
- Experion – Honeywell
- Eclipse SCADA – Elipse Software
- Elipse E3 – Elipse Software
- Vijeo Citect – Schneider Electric
- PowerSCADA – Schneider Electric
- Sage – CEPEL

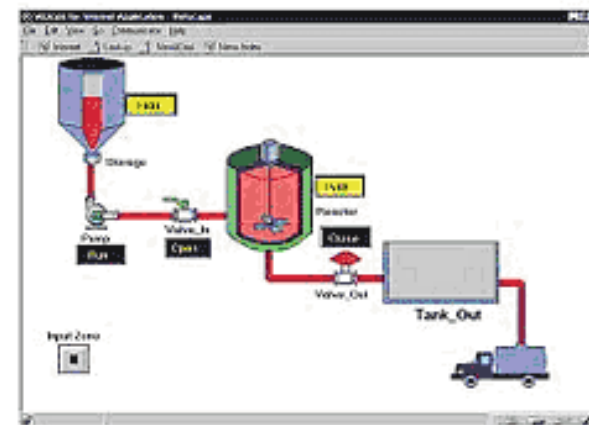
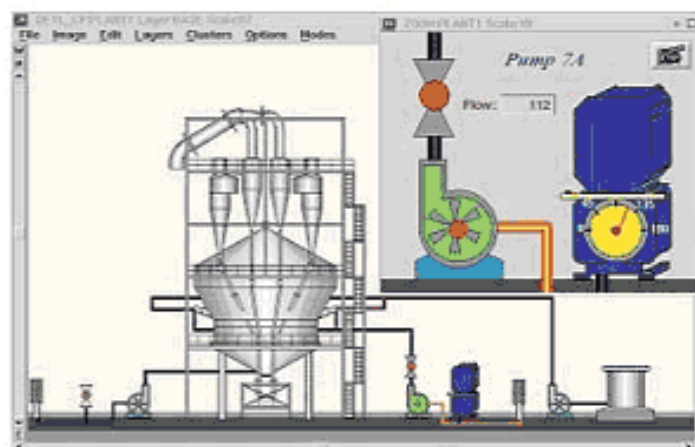
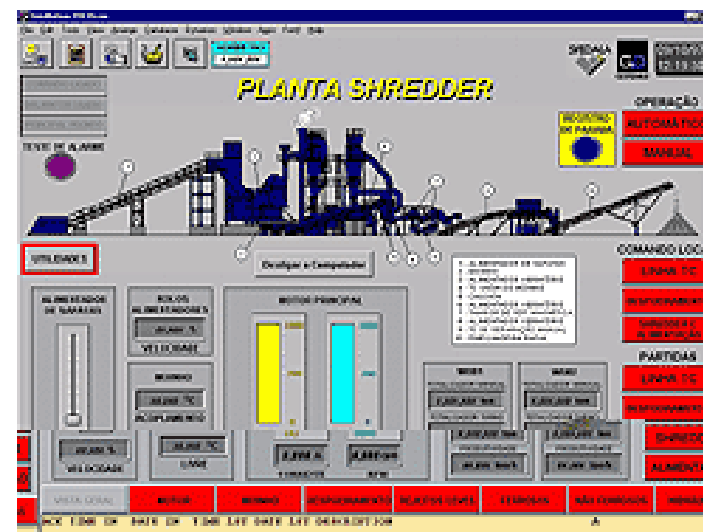
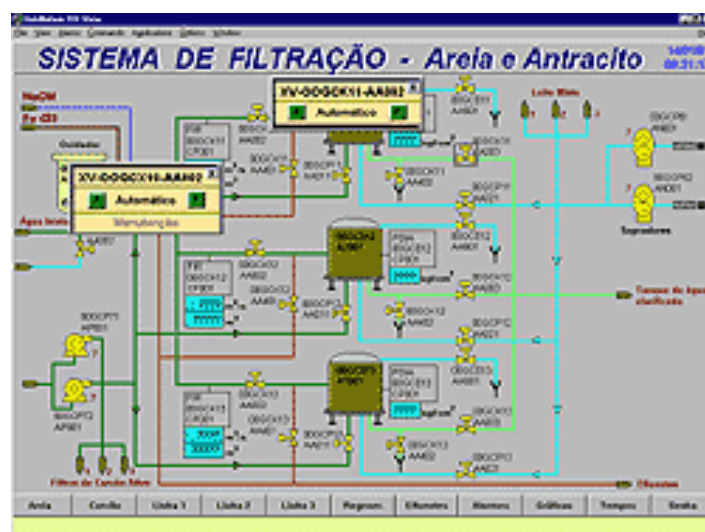


Figura 1 – Exemplos de Telas de Supervisão (Sistemas Wizcon e Altus)

Para pequenas e médias aplicações (até alguns milhares de pontos de monitoração) softwares como o **Elipse for Windows** apresentam uma relação custo/ benefício muito interessante.

2. Princípio de Funcionamento

O funcionamento de um software supervisor baseia-se nas seguintes atividades (bidirecionais):

- 1- Coleta das informações a serem monitoradas, a partir do equipamento de automação;
- 2- Transcodificação das informações recebidas para um formato computacional padrão;
- 3- Disponibilização individualizada destes dados em elementos chamados **TAG's**;
- 4- Exibição dos dados nas telas, no formato escolhido pelo programador;

A conexão completa de uma planta com automação e supervisão apresenta, basicamente, o seguinte lay-out:

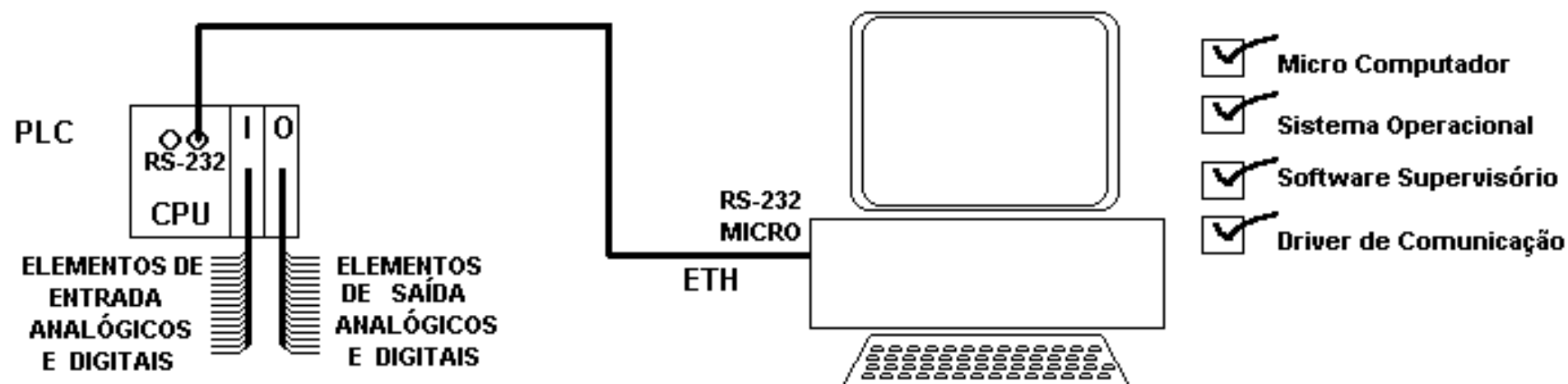


Figura 2- Configuração Básica de um Sistema de Supervisão

Função dos elementos:

- **PLC:** Controlar o processo por meio do correto acionamento das saídas em função das entradas, conforme o programa desenvolvido pelo usuário, e transmitir ao computador todas variáveis solicitadas pelo programa de supervisão;
- **Porta de comunicação do PLC:** apresenta duas funções distintas: em um primeiro momento é utilizada no desenvolvimento do programa do usuário que irá controlar a planta (processos de transferência e monitoração do programa) e, em um segundo momento, será usada para a comunicação com o computador a fim de transmitir/receber dados via programa supervisorio;
- **Cabo de interligação:** é o cabo responsável pelo tráfego das informações digitais entre o PLC e o computador, geralmente apresenta a configuração padrão do sinal serial RS232, RS485 ou Ethernet;
- **Porta serial RS232 ou ETH do Micro:** é o ponto de conexão entre o PLC e o computador, presente em todo computador;

- **Computador:** sua função é gerenciar os dados, controlar a comunicação e abrigar os programas necessários à comunicação PLC $\leftarrow \rightarrow$ PC;
- **Sistema Operacional:** é o programa base que apoiará o supervisório. É a plataforma do sistema supervisório. Os sistemas operacionais mais utilizados são: Windows, Linux, Mac OS e Android;
- **Software Supervisório:** é o programa que controlará a comunicação com o micro e a exibição dos dados na tela do mesmo;
- **Driver de comunicação:** é um pequeno programa que permite acessar os códigos internos dos equipamentos de controle e automação, funciona como um tradutor para o supervisório. Cada tipo e cada modelo de PLC apresentam seu driver próprio, que é desenvolvido pelo fabricante do supervisório e vendido em separado. Este arquivo é acessado toda vez que a comunicação com o equipamento de controle está ativa.

3. Processo de supervisão

A criação de uma aplicação de supervisão de sistemas automatizados consiste na elaboração de desenhos e telas e sua respectiva interação com o PLC.

Trata-se, portanto, de uma atividade técnica que reúne conhecimentos de informática e uma certa habilidade na arte da criação de figuras, símbolos, ambientes, animações, etc; é, portanto, a união das **habilidades técnica e artística** do programador.

Para uma mesma aplicação, cada programador terá o seu ponto de vista e a sua maneira de expressar as informações, não existe limite quanto à qualidade do resultado, a única limitação é a imaginação do programador.

4. Software Supervisório Elipse for Windows



O Elipse Windows é uma ferramenta de supervisão que oferece alto desempenho aliado a poderosos recursos que facilitam a tarefa de desenvolvimento de aplicações. Ele é totalmente configurável pelo usuário e as variáveis do processo podem ser visualizadas de forma gráfica, permitindo, em tempo real, uma fácil e rápida compreensão do que está acontecendo.

Com este objetivo vários Objetos de Tela estão disponíveis como, por exemplo, Gráfico de Barras, Tendências (Gráfico de linhas), Display's, Gauges (galvanômetros analógicos), etc. Além disso, é possível enviar ou receber informações dos equipamentos de aquisição de dados por meio de Setpoints, Sliders ou botões. Permite ainda a aplicação em ambiente de rede, aplicações remotas, aplicações tipo CEP (controle estatístico de processo) e possui conectividade aberta para banco de dados (ODBC).

Uma estrutura verticalizada, chamada Organizer, permite uma visão global de todo o aplicativo de uma forma clara e organizada, facilitando edições e configurações de todos os objetos envolvidos no sistema, como uma árvore hierárquica de opções.

O Elipse for Windows possui 3 módulos disponíveis:

- Módulo **Configurador**: é o módulo utilizado para o desenvolvimento de aplicações de supervisão de processos (nº de TAGs). Para a realização de teste o Configurador permite rodar aplicações por até 10 minutos.
- Módulo **RunTime**: módulo para a execução de um aplicativo de supervisão desenvolvido com o uso do configurador. O runtime não permite qualquer alteração no aplicativo gerado pelo configurador, a menos que existam telas pré-definidas para alterações no aplicativo em tempo de execução.
- Módulo **Master**: Inclui o Configurador e o Runtime.

Visão da Tela Principal do Elipse:

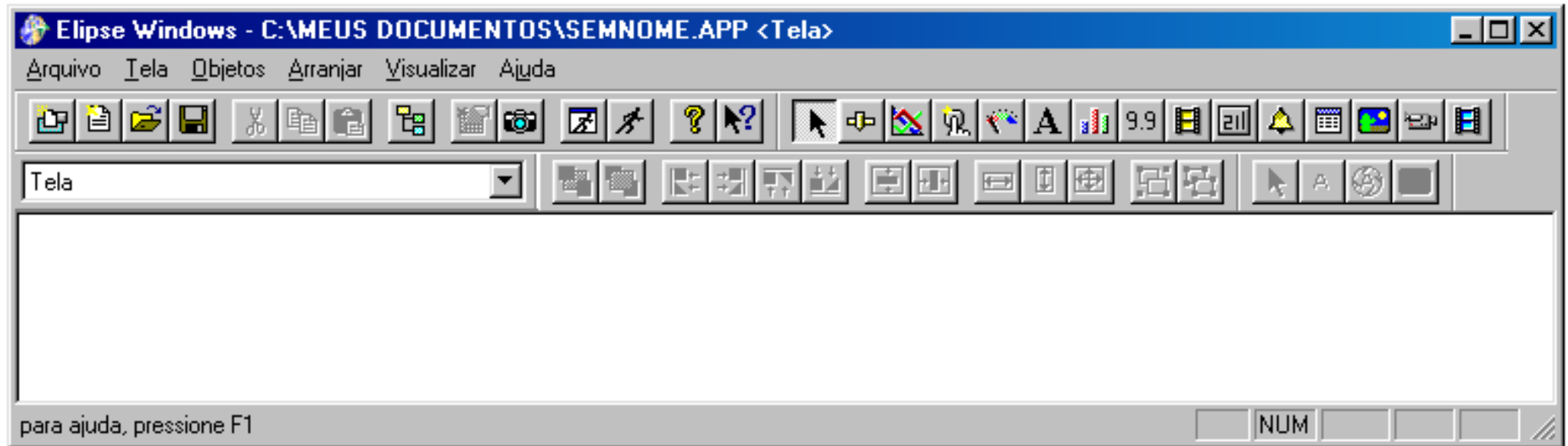


Figura 3- Elipse for Windows

Definições importantes:

- **TAG:** Os Tags são todas as variáveis numéricas ou alfanuméricas envolvidas num aplicativo, ou seja, cada elemento de entrada, saída, memória, constante, etc. Estas variáveis podem assumir vários tipos de acordo com a utilização do tag selecionado, definidos em *Tag Type*.
- **Tela:** Uma tela no Eclipse Windows pode ser definida como uma janela de monitoração de um aplicativo. Cada aplicativo pode conter um número ilimitado de telas, abertas ou fechadas.
- **Aplicativo:** O aplicativo é o conjunto de telas, Tags, Históricos, Receitas e Scripts que podem ser definidos.
- **Objetos de animação:** São elementos gráficos que se relacionam com os Tags. Estes mostram graficamente o comportamento dessas variáveis Tags. As telas de um aplicativo são compostas por bitmaps de fundo e objetos de animação, que compõe as figuras de tela. Os objetos que são inseridos sobre o bitmap constituem um plano secundário na tela, de forma que podem ser deletados, copiados, movidos e redimensionados, sem prejudicar o desenho de fundo.

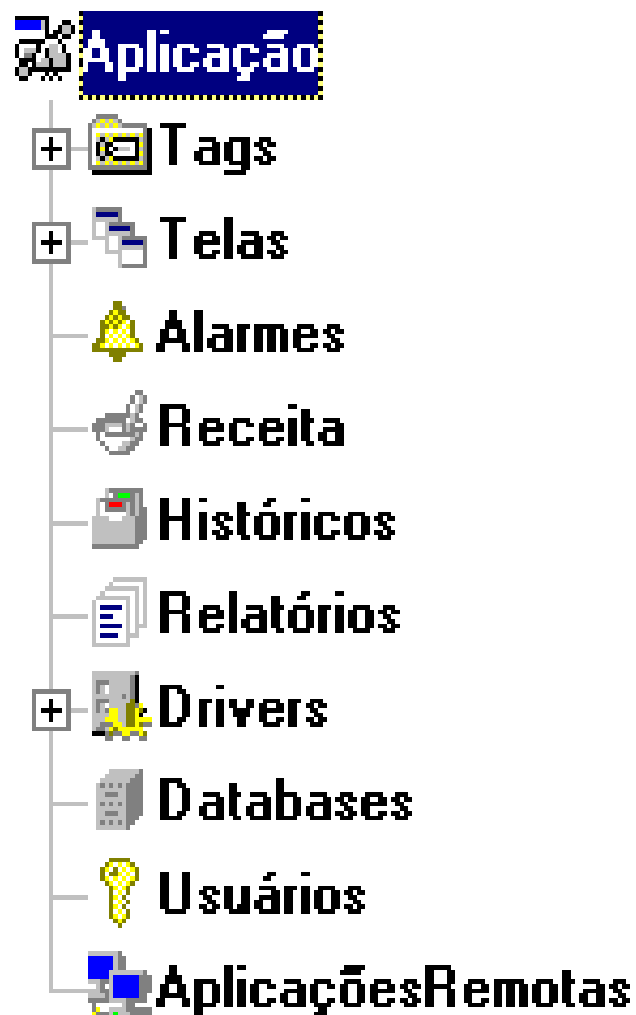
Recursos do Sistema - Organizer

O Eclipse Windows apresenta uma concepção estrutural para a criação e organização do seu sistema supervisório: o **Organizer**. Ele permite uma visão simples e organizada de toda a aplicação, ajudando na edição e configuração de todos os objetos envolvidos no sistema por meio de uma árvore hierárquica de opções.

A estrutura do Organizer pode ser comparada à árvore de diretórios do Gerenciador de Arquivos do Windows. Desta forma a estrutura da aplicação começa no canto superior esquerdo com a raiz da aplicação. Todos os objetos da aplicação descem a partir da raiz agrupados de acordo com seu tipo: Tags, Telas, Alarmes, Receitas, Históricos, Relatórios, Drivers, Databases; que se constituem nos principais elementos de sua aplicação.

Selecionando-se qualquer um dos ramos da árvore da aplicação, o mesmo irá se expandir mostrando seu conteúdo, desta forma pode-se facilmente navegar pela aplicação, tendo disponíveis todas as opções de configuração desde a criação de Tags até o redimensionamento de objetos em uma tela específica.

A estrutura básica do Organizer é apresentada a seguir:



Pode-se ativar o Organizer (para uma aplicação aberta) selecionando o comando *Organizer* do menu *Arquivo (File)* ou pressionando o botão do *Organizer* na Barra de Ferramentas. A seguinte janela irá aparecer:

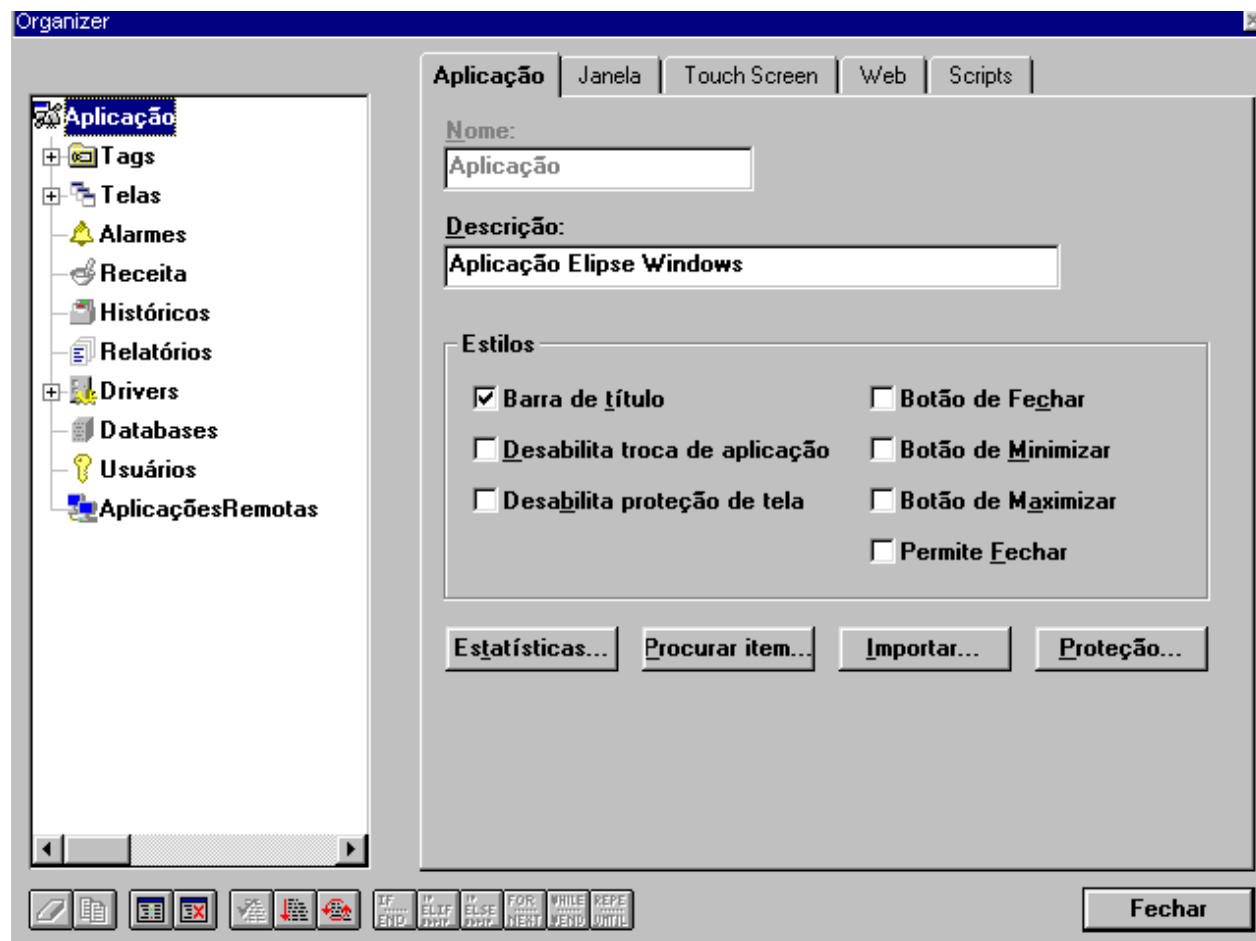


Figura 4- Organizer do Elipse

A partir do Organizer pode-se criar toda a sua aplicação simplesmente navegando na árvore da aplicação. Selecionando-se qualquer um de seus ramos, as propriedades do objeto selecionado serão mostradas no lado direito da janela onde poderão ser editadas.

Por exemplo, selecionando-se Tags, na árvore do Organizer, poderão ser criados novos Tags e editadas suas propriedades, selecionando-se a página desejada a partir dos *tabs* no topo da janela.

As seções a seguir irão explicar cada objeto da aplicação presentes na árvore do Organizer:

4.1- Aplicação

Pode-se criar uma aplicação com o comando *Nova Aplicação* do menu *Arquivo* ou pressionando o botão *Nova Aplicação* na Barra de Ferramentas. Uma aplicação pode ser considerada como sendo um grupo de Tags, Telas, Alarmes, Receitas, Históricos, Relatórios, Drivers, Databases e Scripts que podem definir o sistema.

O comando *Salvar Aplicação* do menu *Arquivo* ou o botão *Salvar* grava a aplicação corrente em um arquivo com extensão APP.

As opções de configuração da aplicação serão definidas no Organizer selecionando-se a raiz da Aplicação.

4.2- Criação de um aplicativo de supervisão

1) Criação de um aplicativo para abrigar as telas:

Arquivo → Nova Aplicação → Digite o nome para a aplicação **→ Ok.**

2) Criar um nome para a tela:

Em uma aplicação às vezes faz-se necessário a utilização de várias telas para um melhor detalhamento das etapas de monitoração, para isso é necessário que as telas tenham nomes que as identifiquem para que seja feita a correta exibição das mesmas.

Para criar um nome e demais definições para a tela:

Tela → Propriedades → Digite o nome desejado na barra **“Título” → Fechar**

3) Criação do arquivo de comunicação (configuração do **driver de comunicação**):

Nesta etapa é necessário determinar alguns dados referentes ao equipamento de controle, no caso o PLC, para uma correta conexão:

Arquivo → Organizer → Drivers → Novo → Localize o arquivo **driver de comunicação** referente ao PLC, no caso será o arquivo Slc500_32.dll

Configurar → A seguir é necessário configurar os parâmetros P1, P2, P3 e P4, conforme as instruções abaixo:

Rockwell SLC-500

Arquivo: SLC500_32.DLL

Fabricante: Rockwell

Equipamentos: SLC5/01, SLC5/02, SLC5/03, Micrologix

Protocolo: DF1

Parâmetros (p) de comunicação do Driver:

p1 - porta serial (só usado para conexão sem modem):

COM1=1

COM2=2

COM3=3

COM4=4

➔ No laboratório, o PC está conectado na porta COM1. **Portanto: p1=1**

p2 - velocidade de transmissão de dados (só usado para conexão sem modem) :

0 - 110,

1 - 300,

2 - 600,

3 - 1200,

4 - 2400,

5 - 4800,

6 - 9600,

7 - 19200;

➔ No laboratório, o PC está configurado com a velocidade de 19.200 kbps na porta COM1. **Portanto: p2=7**

p3 - endereço do micro e CRC:

Endereço do PC (source) + CRC*100; através deste parâmetro você informa ao driver o número do PC na rede e se deseja usar BCC (0) ou CRC (1). Veja os exemplos a seguir:

Ex 1: PC=7 e desejo usar BCC: $p3 = 7 (7+0*100)$

Ex 2: PC=7 e desejo usar CRC: $p3 = 107 (7+1*100)$

➔ No laboratório, o PC está conectado diretamente à CPU do PLC, assim sendo, o endereço do PC é 0 e o do PLC é 1. O código de verificação de erro configurado no PC é do tipo CRC. Desta forma, a equação fica: $p3=0+1*100$. **Portanto: p3=100**

p4 - time-out em milésimos de segundo (sugerido: entre 100 e 200 milésimos).

➔ No laboratório, o PC está conectado muito próximo ao PLC, assim sendo, poder-se utilizar um time-out mais reduzido. **Portanto: p4=100.**

Terminada a configuração ➔ **Fechar**

4) Criação dos elementos de supervisão, os **TAG's**, para cada elemento de monitoração

Arquivo → Organizer → Tags → Novo → Tipo

- Nome: Nome do Tag. Podem ser usados tantos caracteres quanto se queira, mas espaços e caracteres especiais não são permitidos quando os Tags forem usados nos Scripts.
- Quantos: Define o número de Tags que serão criados com as mesmas características especificadas sendo que os nomes serão acrescidos de um índice numérico crescente.

- Tipo dos TAGs:
 - **Tags PLC (CLP)** são usados para trocar valores com os equipamentos de aquisição de dados. Os parâmetros mudam de acordo com cada driver, estes possuem um arquivo .TXT que descreve os detalhes de configuração.
 - Tags ODE (Dynamic Data Exchange) são usados para trocar dados com servidores ODE Excel, Word, Access, Elipse.
 - Tags Demo são usados para gerar dados randômicos ou funções pré-definidas. Tags Expressão permitem a entrada de uma expressão numérica ou alfanumérica (permitem concatenação de strings).
 - Tags Bloco são usados para ler um bloco de valores simultaneamente.
 - Tags RAM são variáveis em geral, são voláteis e usados para armazenar valores na memória. Os valores do Tag RAM podem ser números ou strings.

5) Configuração dos TAG's tipo PLC:

Arquivo → Organizer → TAGs → Novo Tag → Digite a quantidade de TAG's a serem monitorados do PLC → Em **Tipo de TAG** clique em **PLC → OK**

A configuração de um TAG tipo PLC consiste na criação de um endereçamento que associe o elemento com seu respectivo endereço de memória no PLC. A sequência será:

Arquivo → Organizer → TAGs → tagXXX → Driver → Driver1 – Allen Bradley SLC-500 → A seguir é necessário configurar os parâmetros N1, N2, N3, N4 e SCAN, conforme as instruções abaixo:

Parâmetros de endereçamento para Tags tipo PLC ou BLOCO Allen Bradley SLC-500:

size - número de elementos do bloco.

n1/b1 - número do clp combinado com Número do arquivo: $n1 = (clp * 1000) + \text{arquivo (decimal)}$.

Variável	Arquivo	Tipo (n1)
WORD	SLC_OUTPUT	0
WORD	SLC_INPUT	1
WORD	SLC_STATUS	2
WORD	SLC_BIT	3
WORD	SLC_TIMER	4
WORD	SLC_COUNTER	5
WORD	SLC_CONTROL	6
WORD	SLC_INTEGER	7
FLOAT	SLC_FLOAT	8

Exemplo: com $clp=5$ e arquivo tipo memória inteira=7, $n1=5007$.

➔ No laboratório, o PC está conectado diretamente à CPU do PLC, assim sendo, o endereço do PC é 0 e o do PLC é 1. Desta forma, a equação fica: $n1=1*1000+\text{arquivo}$.

- **Portanto: para saídas $n1=1000+0=1000$**
para entradas $n1=1000+1=1001$
para memórias inteiras $n1=1000+7=1007$
para memórias reais $n1=1000+8=1008$

$n2/b2$ - tipo da variável (ver tabela-1);

Tabela-1:

Variável	Arquivo	Tipo ($n2/b2$)	Variável	Arquivo	Tipo ($n2/b2$)
WORD	SLC_OUTPUT	0	BCD	SLC_OUTPUT	8
WORD	SLC_INPUT	1	BCD	SLC_INPUT	9
WORD	SLC_STATUS	2	BCD	SLC_STATUS	10
WORD	SLC_BIT	3	BCD	SLC_BIT	11
WORD	SLC_TIMER	4	BCD	SLC_TIMER	12
WORD	SLC_COUNTER	5	BCD	SLC_COUNTER	13
WORD	SLC_CONTROL	6	BCD	SLC_CONTROL	14
WORD	SLC_INTEGER	7	BCD	SLC_INTEGER	15
			FLOAT	SLC_FLOAT	24

n3/b3 - número do elemento inicial do arquivo;

→ É o número do endereço do elemento (byte ou word)

n4/b4 - número do sub-elemento inicial do arquivo (usado apenas para variáveis do tipo bit).

→ É o número do bit do endereço do elemento (byte ou word)

Scan: Intervalo entre atualizações das leituras/escritas do TAG em milisegundos.

→ Nas aplicações desenvolvidas no laboratório, normalmente utiliza-se um tempo de scan da ordem de 100 ms. **Portanto: scan=100.**

Exemplos:

TAG para N7:2 → n1=1007, n2=7, n3=2, n4=0, scan=100.

TAG para F8:5 → n1=1008, n2=24, n3=5, n4=0, scan=100.

→ **Propriedades do Tag PLC**

O Tag PLC (CLP) é usado para trocar informações com os equipamentos de aquisição de dados usando drivers de I/O fornecidos pela Elipse Software de acordo com o tipo do equipamento.

Os drivers de I/O do Elipse são arquivos separados com extensão DLL. Um arquivo TXT é fornecido com cada driver contendo informações importantes a respeito de sua configuração. As configurações disponíveis são:

- Nome do Tag. Pode-se usar tantos caracteres quanto queira mas espaços e caracteres especiais não são permitidos quando os Tags forem usados nos Scripts.
- Mudar Tipo para: Permite que se mude o tipo do Tag.
- Descrição: Uma breve descrição sobre o Tag.
- Driver: Permite a seleção de um driver de comunicação para o Tag corrente. Os drivers devem ser instalados através da janela de Drivers a fim de que estejam disponíveis.
- Parâmetros "N": Permite a configuração dos parâmetros de endereçamento 'N' para o Tag corrente de acordo com o driver selecionado. Esta configuração está contida no arquivo TXT que acompanha o driver.

- Escala: Marcando esta opção os valores do Tag serão convertidos para uma nova escala de valores determinada pelo usuário conforme os limites definidos em CLP Inferior, Sist. Inferior, CLP Superior e Sist. Superior.
- CLP Inferior: Define o valor mínimo a ser lido do PLC (CLP).
- Sist. Inferior: Define o novo valor mínimo para a conversão dos valores lidos.
- CLP Superior: Define o valor máximo a ser lido do CLP (PLC).
- Sist. Superior: Define o novo valor máximo para a conversão dos valores lidos.
- Testa conexão aqui: Testa a comunicação com o PLC lendo e escrevendo alguns valores
- Habilita a leitura para um Tag PLC (CLP).
- Habilita a leitura automática para o Tag PLC (CLP).
- Habilita escrita automática para o Tag PLC (CLP).

6) Criação dos objetos de animação

6.1- Objetos de Exibição (Leitura de TAGs)

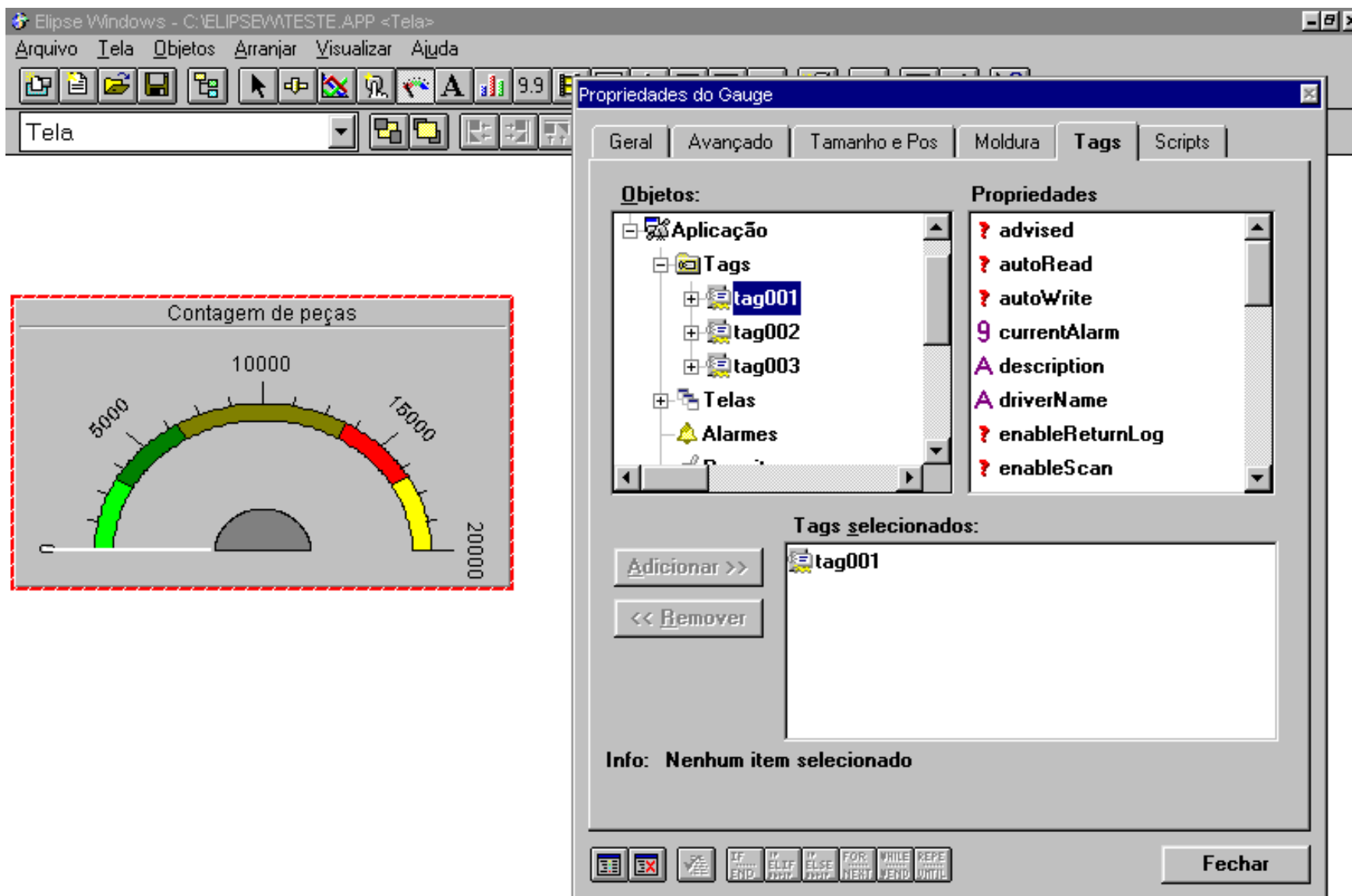
Gauge (Medidor)

O objeto Gauge (mostrador analógico) é utilizado para a visualização de variáveis analógicas através do indicador gráfico. Os valores de escala e propriedades do Gauge são definidos livremente pelo usuário. Existem 4 posições para a definição do objeto, podendo aparecer para cima, para baixo, para esquerda ou para a direita.

→ Objeto de Leitura 

Exemplo:

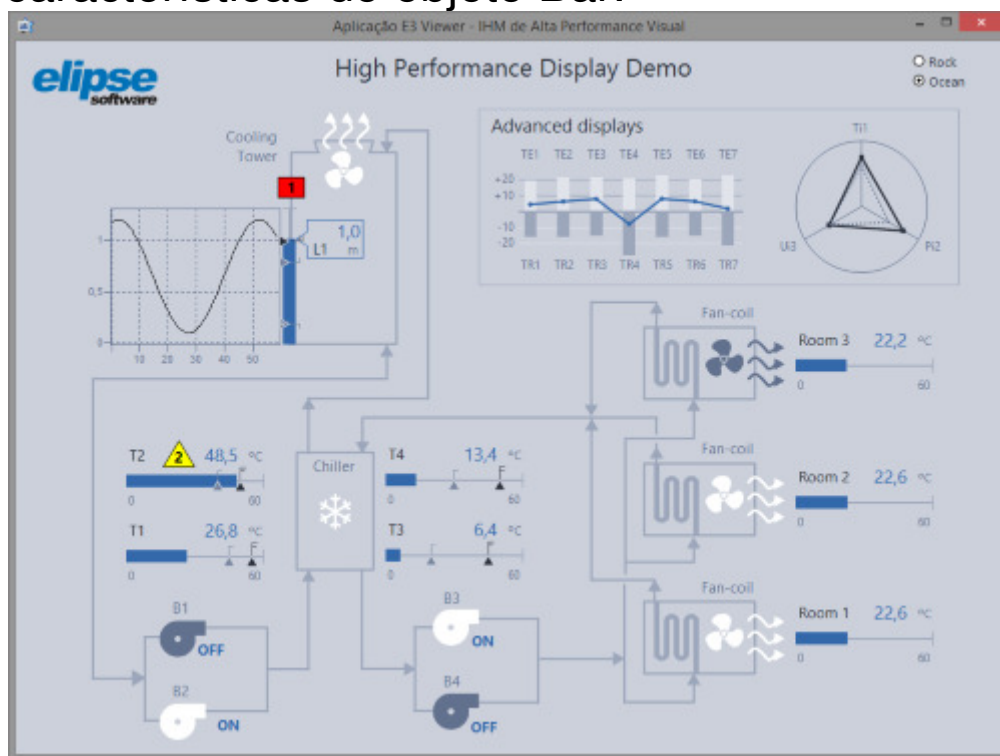
Criar um instrumento analógico (Gauge) que meça o valor do TAG001.



Bar Graph



O objeto Bar permite a visualização gráfica de valores através de gráfico de barras. O usuário poderá definir um Tag cujo valor será mostrado em forma de barra, possibilitando assim, a visualização de níveis e volumes de forma mais clara e agradável. Através das propriedades da barra pode-se definir cores, divisões de escala, orientação da barra e outras características do objeto Bar.



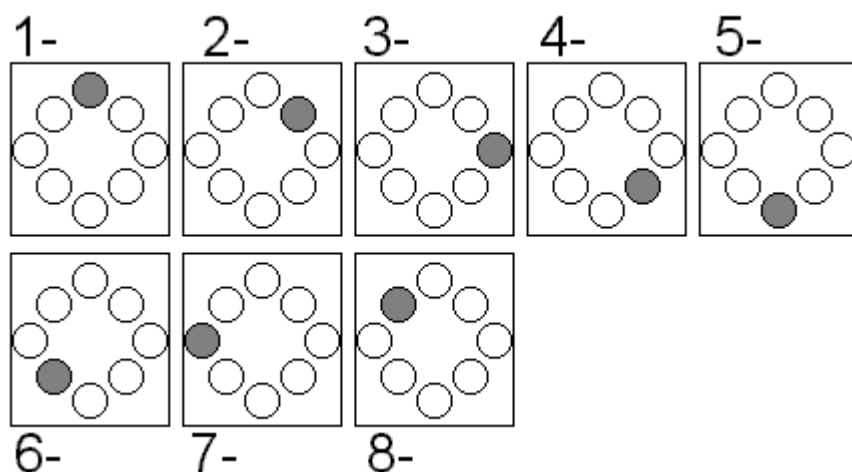
→ Objeto de Leitura



Animation

O objeto Animation permite criar animações, quadro a quadro, com os desenhos criados pelo usuário. Deve-se definir faixas de valores do Tag associado para que os desenhos sejam trocados conforme a variação do Tag, criando a animação desejada. Os bitmaps para a troca dos quadros podem ser de quaisquer tamanhos e cores diversas, sendo que não existe limites para tais entradas. O botão Add permite adicionar um bitmap.

A criação de figuras animadas ou que mudam de cor são obtidas com o objeto *animation* do Elipse. A confecção destes elementos é baseada na teoria da animação visual, caracterizada pela rápida superposição de imagens gerando a ilusão do movimento. Assim, podem-se conseguir efeitos marcantes utilizando-se desenhos simples.



→ Objeto de Leitura



Alarms

O objeto Alarme permite verificar, em tempo de execução, os alarmes que estão ativos no sistema ou os alarmes que aconteceram e foram registrados em disco (Summary/History). Os alarmes são características dos Tags (definidos individualmente), que podem ou não ter seus valores verificados para situações de alarme para até quatro faixas distintas e com diversas prioridades. As faixas de alarmes vão de LOLO (baixo crítico), passando por LO (baixo) e HI (Alto), chegando até HHI (alto crítico). Por meio das propriedades do Alarme o usuário poderá definir formatos de data, hora e mensagens que surgirão na ocorrência de um alarme, bem como definir as cores para as diferentes situações de alarmes.



→ Objeto de Leitura



Browser (Mostrador de Tabelas)



O objeto browser tem como finalidade facilitar ao usuário a visualização dos dados de seus arquivos de banco de dados. Estes arquivos de banco de dados, devem ser no formato Elipse, ou seja, devem ser gerados pelo Elipse Windows. Para localizar um arquivo de dados basta identificar a extensão do arquivo de dados, que deverá ser .dat ou .hdr. Nesses casos a extensão mais comum é .dat que é utilizada em arquivo de históricos e de bateladas.

Nome	Qualidade	Marca de Tempo	Valor
[RTU_SICAM-MINA-SHO].SE_MINA_TPO002			
AMP060110ME	216	15/05/2017 12:58:46.828	9 2895
AMP060111ME	216	15/05/2017 12:58:46.828	9 2895
AMP060112ME	216	15/05/2017 12:58:46.828	9 2895
Count	192	15/05/2017 12:58:34.890	9 79
DocString	192	15/05/2017 12:58:34.886	9 1sformador TPO002 60 kV a 34.5 kV
F03TPO200AG	216	15/05/2017 12:24:05.844	9 0
F03TPO201AG	216	15/05/2017 12:24:05.844	9 0
F03TPO202AG	216	15/05/2017 12:24:05.844	9 0
F03TPO203AG	216	15/05/2017 12:24:05.844	9 0
F03TPO204AG	216	15/05/2017 12:24:05.844	9 0
F03TPO205AG	216	15/05/2017 12:24:05.844	9 0
F03TPO206AG	216	15/05/2017 12:24:05.844	9 0
F03TPO207AG	216	15/05/2017 12:24:05.844	9 0
F03TPO208AG	216	15/05/2017 12:24:05.844	9 0
F03TPO209AG	216	15/05/2017 12:24:05.844	9 0
F03TPO20COM	216	15/05/2017 12:24:05.844	9 0
F03TPO210AG	216	15/05/2017 12:24:05.844	9 0
F03TPO211AG	216	15/05/2017 12:24:05.844	9 0
F03TPO212AG	216	15/05/2017 12:24:05.844	9 0
F03TPO213AG	216	15/05/2017 12:24:05.844	9 0
F03TPO215AG	216	15/05/2017 12:24:05.844	9 0
F03TPO216AG	216	15/05/2017 12:24:05.844	9 0
F03TPO218AL	216	15/05/2017 12:24:05.844	9 0
F03TPO200AI	216	15/05/2017 12:24:05.844	9 0

Status: Conectado

→ Objeto de Leitura



Display

O objeto Display permite visualizar de forma numérica o valor de um Tag. O valor que aparece no display representa o valor do Tag no instante atual. As propriedades do objeto Display permitem definir parâmetros tais como cor e tamanho dos fontes de texto, alinhamento do número do display e outras características.



→ Objeto de Leitura 

Trend (Tendência)

O objeto Trend é utilizado para a visualização gráfica de até 16 Tags em forma de tendência. O gráfico é animado e mostra os valores em tempo real, ou seja, conforme varia o processo e suas variáveis, varia o gráfico de Tendências.

Através do objeto Trend é possível ainda gerar gráficos de variável contra variável em tempo real, possibilitando uma análise imediata dos dados de processo.

Para editar os parâmetros do Trend dê um duplo clique sobre o objeto.



→ Objeto de Leitura

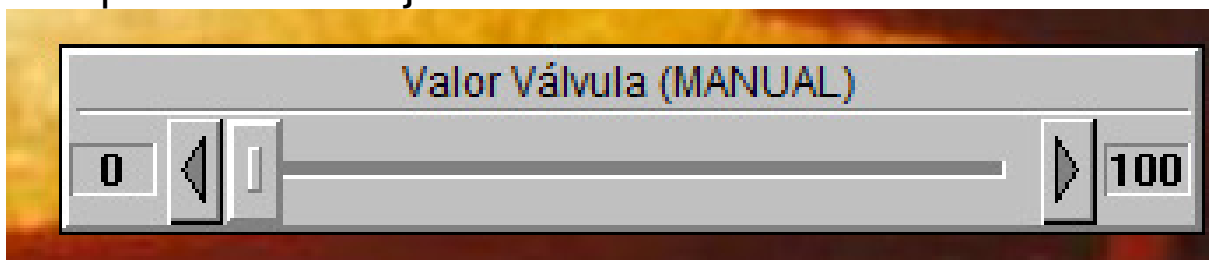


6.2- Objetos de Edição (Escrita de TAGs)

Slider (Deslizante)

O objeto Slider é utilizado para forçar valores (**escrita**) no Tag selecionado. O valor do tag varia de acordo com a posição do potenciômetro (botão deslizante) ou pressionando-se as setas de direção.

Para mover o potenciômetro clique com o mouse sobre o botão e, mantendo pressionado o mouse, arraste o botão na posição que desejar. As setas de direção nas extremidades do Slider permitem a alterar o valor do Tag passo a passo, clicando-se sobre as mesmas quantas vezes forem necessárias. Para editar os parâmetros do Slider dê um duplo clique sobre o objeto.



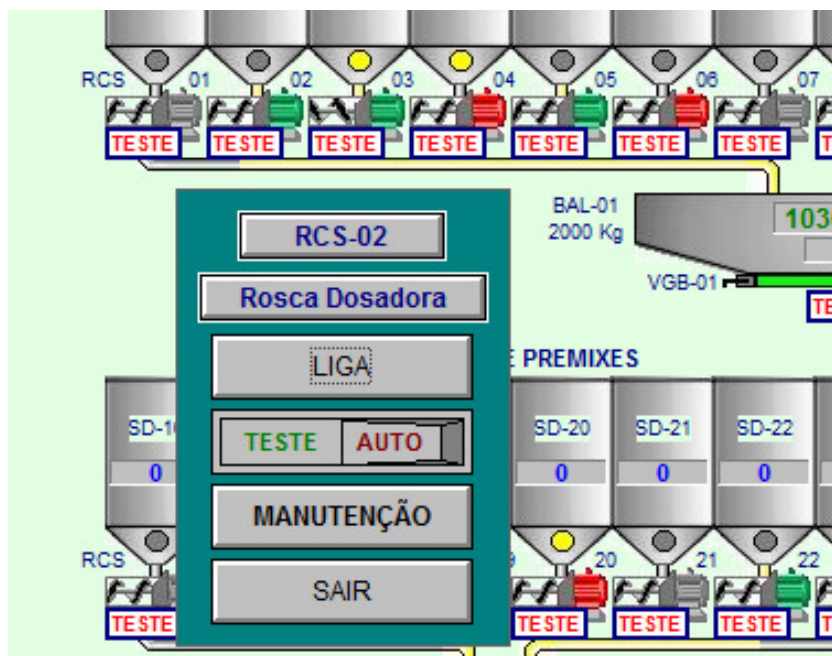
→ Objeto de Escrita 

Button (Botão)

O objeto Button é utilizado para acionamentos ou execuções de tarefas especificadas pelo usuário, através do mouse ou teclado. Através do objeto Button o usuário poderá definir teclas (botões) para a troca de telas.

Existem 3 tipos básicos de botões que podem ser configurados:

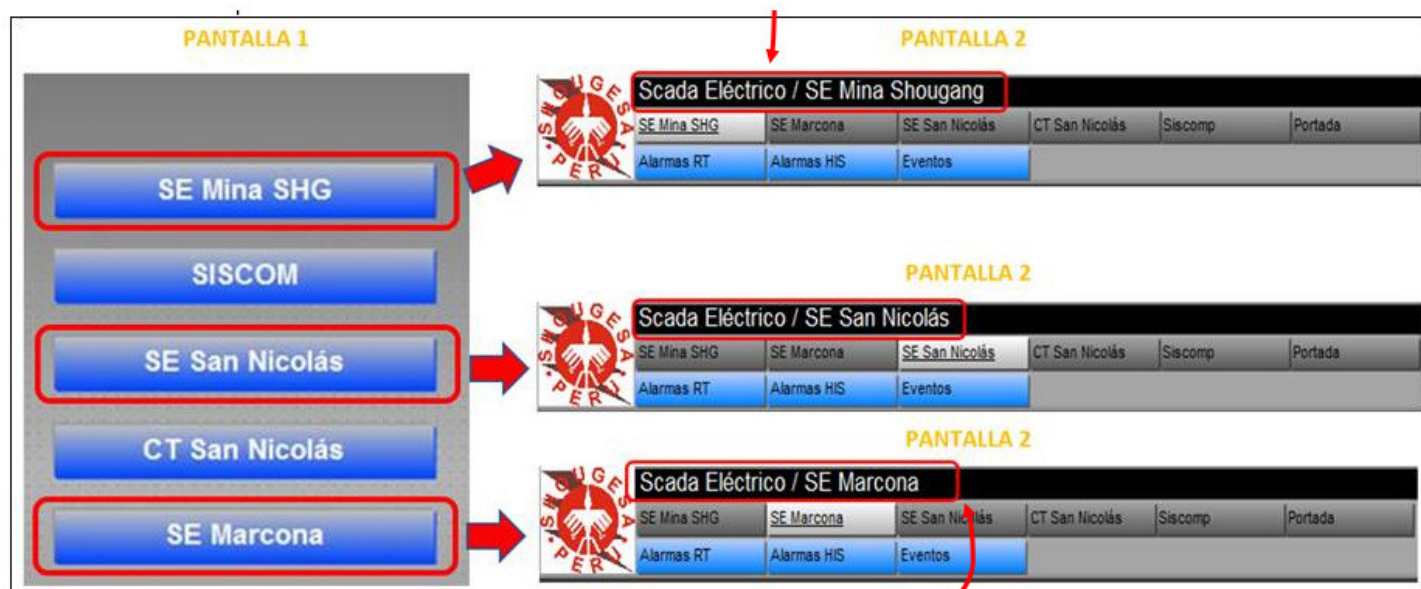
- Momentary (momentâneo) - para forçar um valor ou executar a troca de telas
- Toggle (liga-desliga) - para acionamentos do tipo Liga/Desliga (On/Off)
- Jog - para alternar entre um valor enquanto pressionado e outro quando solto.



→ Objeto de Escrita 

Text

O objeto Text permite definir-se mensagens associadas à faixas de valores de um Tag. O usuário pode definir quantas faixas (zones) desejar, sendo que cada faixa possui um texto que varia de acordo com o valor do Tag associado.



→ Objeto de Escrita



Setpoint

O objeto Setpoint permite alterar (escrever), via teclado, o conteúdo de um Tag. O valor ou texto digitado é enviado para o Tag após o ENTER. Através das propriedades do Setpoint o usuário poderá definir o tipo de valor de entrada (inteiro, decimal ou texto), bem como o tamanho e cor dos caracteres do objeto.



→ Objeto de Escrita 

7) Outras Funcionalidades

- **Scripts**: Os Scripts são pequenos procedimentos (programas) escritos em linguagem de programação própria – Elipse Basic – que permitem a execução de tarefas extras na aplicação. Os scripts são sempre associados a eventos.
- **Receita**: Uma Receita é um conjunto de valores pré-definidos que podem ser carregados para um grupo de tags a fim de configurar um processo específico. Esta lista de tags também é chamada de modelo de receita.
- **Históricos**: Os Históricos são objetos responsáveis pelo armazenamento de valores de tags. O armazenamento pode ser feito por tempo ou por evento, que devem ser especificados para a gravação dos dados.
- **Relatórios**: Os relatórios permitem realizar a impressão dos dados históricos, cabeçalhos e alarmes, e ainda dados instantâneos.

Referências:

- Elipse SCADA Tutorial - Elipse Software Ltda. 10/12/2008 - Versão 2.29
- Manual do Usuário Elipse SCADA - Elipse Software Ltda. 13/8/2010 - Versão 2.29