

# Driver Elipse Ping

<b>Nome do Arquivo</b>	PingDrv.dll
<b>Fabricante</b>	Elipse Software
<b>Equipamentos</b>	Qualquer equipamento conectado via protocolo IP (Internet Protocol)
<b>Protocolo</b>	ICMP (Internet Control Message Protocol)
<b>Versão</b>	3.0.14
<b>Última Atualização</b>	05/11/2025
<b>Plataforma</b>	Win32
<b>Dependências</b>	Nenhuma
<b>Leitura de Superblocos</b>	Não
<b>Nível</b>	0

## Introdução

O Driver Elipse Ping permite a execução do comando **ping** em qualquer estação de uma rede IP (*Internet Protocol*), inclusive via Internet. Este Driver é útil em aplicações de rede que necessitam verificar a presença de parceiros em uma rede.

A partir da versão **2.0**, este Driver permite a realização de comandos **ping** em múltiplas *threads*, o que permite a diminuição do tempo de varredura real em aplicações com grande número de Tags. Se o número máximo de *threads* é definido como 0 (zero) na aba **Ping**, o comportamento deste Driver permanece idêntico à versão **1.0**, com todos os comandos **ping** executados sequencialmente.

Na versão **2.0**, este Driver foi portado para usar a biblioteca **IOKit** da **Elipse Software**. Esta biblioteca permite que Drivers tenham acesso ao meio físico desejado, **Serial**, **Ethernet**, **Modem** ou **RAS**, de maneira padronizada. Este Driver, entretanto, não utiliza nenhum meio físico específico da biblioteca **IOKit**, e deve-se portanto configurar a opção **Physical Layer** da aba **Setup** da **janela de configuração deste Driver** com o valor **None**.

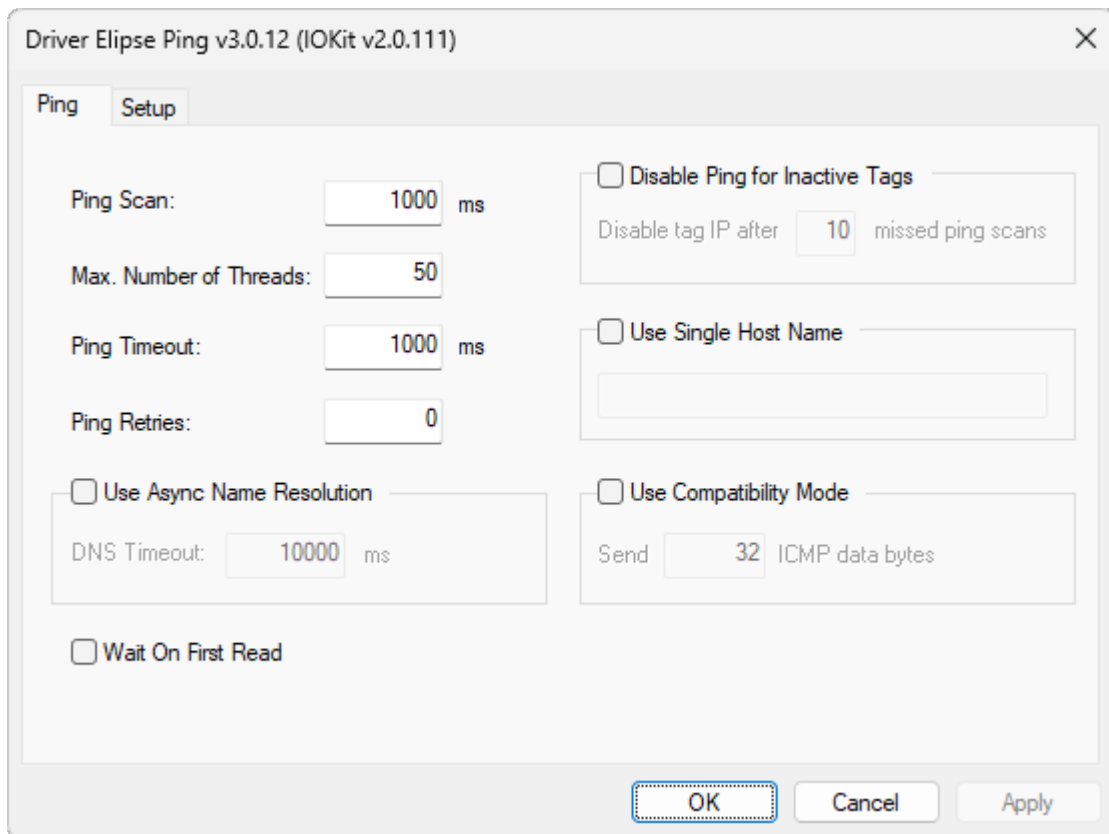
Para mais informações sobre a configuração da biblioteca **IOKit**, consulte o tópico **Documentação das Interfaces de Comunicação**.

## Configuração do Driver

Os parâmetros **[P]** de configuração não são utilizados. Todas as configurações da biblioteca **IOKit** e as específicas devem ser realizadas na janela de configurações deste Driver, ou nas configurações extras no **Elipse SCADA**. Nesta janela, a aba **Ping** contém as configurações específicas deste Driver, enquanto a aba **Setup** apresenta as configurações de comunicação da biblioteca **IOKit** da **Elipse Software**. Para mais informações sobre as configurações da biblioteca **IOKit**, consulte o tópico **Documentação das Interfaces de Comunicação**.

## Configuração das Propriedades

As opções de configuração disponíveis na janela de configuração também podem ser acessadas em modo **Offline**, em tempo de execução, usando as **Strings** descritas no tópico **Configuração das Propriedades Offline**. A figura a seguir mostra as configurações específicas deste Driver na aba **Ping**.



Aba Ping

Esta aba é específica deste Driver e as opções de configuração são descritas neste tópico. Na aba **Setup** deve-se selecionar o item **None** na opção **Physical Layer** e pode-se também habilitar a geração de logs. As demais opções da biblioteca **IOKit** são ignoradas por este Driver. As opções disponíveis nesta aba estão descritas na tabela a seguir.

#### Opções disponíveis na aba Ping

OPÇÃO	DESCRIÇÃO
<b>Ping Scan</b>	Define o tempo interno de varredura para cada Tag, no caso de comandos <b>ping</b> assíncronos ou <i>multi-threaded</i>
<b>Max. Number of Threads</b>	Define o número máximo de <i>threads</i> a serem criadas. Se este valor é maior que o número de Tags de uma aplicação, este Driver cria um <i>thread</i> por Tag. Caso contrário, o limite máximo de <i>threads</i> é dado por este valor. Se este valor é configurado em 0 (zero), os comandos <b>ping</b> são executados de forma síncrona, na própria <i>thread</i> de leitura de cada Tag. Consulte a <b>nota</b> a seguir para mais informações sobre esta opção
<b>Ping Timeout</b>	Tempo limite de um comando <b>ping</b>
<b>Ping Retries</b>	Número de vezes que este Driver reenvia requisições de <b>ping</b> a cada ocorrência de <i>time-out</i> antes de retornar 0 (zero) na propriedade <b>Value</b> dos Tags <b>Simple de Ping</b>
<b>Use Async Name Resolution</b>	Esta opção habilita a resolução de nomes de <i>hosts</i> de forma assíncrona, o que permite configurar um <i>time-out</i> para esta operação com alguma perda de desempenho. Use esta opção apenas nos casos em que uma operação de DNS ( <i>Domain Name System</i> ) tenha a possibilidade de demorar muito tempo ou de falhar com frequência

OPÇÃO	DESCRIÇÃO
<b>DNS Timeout</b>	Tempo limite da resolução de nome de <i>hosts</i> , se a opção <b>Use Async Name Resolution</b> está habilitada
<b>Wait On First Read</b>	Se esta opção está habilitada, este Driver aguarda a realização do primeiro comando <b>ping</b> na primeira leitura de cada Tag, de forma a retornar algum valor. Se esta opção está desabilitada, a primeira leitura de um Tag apenas agenda um comando <b>ping</b> e esta operação retorna sem nenhum valor e sem alterar nenhuma propriedade de um Tag em uma aplicação. Um Tag só passa a retornar valores quando o primeiro comando <b>ping</b> é realizado por este Driver. Deve-se preferencialmente deixar esta opção desabilitada, pois tende a melhorar o desempenho
<b>Disable Ping for Inactive Tags</b>	Esta opção permite a este Driver colocar Tags em estado inativo em suas varreduras internas, caso uma aplicação não solicite uma leitura durante um período de varredura configurável. Em estado inativo, o endereço IP ou o nome de <i>host</i> referente a um Tag não é mais usado nas varreduras internas deste Driver. Um Tag passa para o estado ativo assim que uma nova solicitação de leitura é recebida de uma aplicação
<b>Disable tag IP after <i>n</i> missed ping scans</b>	Informa a este Driver quantas varreduras internas não lidas por uma aplicação devem ocorrer antes que um Tag seja considerado inativo
<b>Use Single Host Name</b>	Esta opção deve ser usada apenas para o <b>Elipse SCADA</b> . O uso desta opção limita uma aplicação a usar apenas um Tag. A caixa de texto permite definir um nome de <i>host</i> usado para um Tag único de uma aplicação, que deve ser definido com os parâmetros <i>N1</i> , <i>N2</i> , <i>N3</i> e <i>N4</i> iguais a 0 (zero). Para o <b>Elipse E3</b> , <b>Elipse Power</b> ou <b>Elipse Water</b> , deixe esta opção desabilitada e use a propriedade <b>Dispositivo</b> ( <i>Device</i> ) de cada Tag para definir um nome de <i>host</i> . Consulte a <b>nota</b> a seguir para mais informações sobre esta opção
<b>Use Compatibility Mode</b>	Esta opção permite a definição de um determinado número de bytes de dados a enviar em um <i>frame</i> de requisição de eco do protocolo ICMP ( <i>Internet Control Message Protocol</i> ). O valor padrão desta opção é desabilitada. Consulte a <b>nota</b> a seguir para mais informações sobre esta opção
<b>Send <i>n</i> ICMP data bytes</b>	Esta opção indica o número de bytes de dados a enviar em um <i>frame</i> de requisição de eco do protocolo ICMP, caso a opção <b>Use Compatibility Mode</b> esteja habilitada. Consulte a <b>nota</b> a seguir para mais informações sobre esta opção

## NOTAS

- A partir da versão **2.1** deste Driver, o número máximo de *threads* está limitado em 100. Caso este Driver seja carregado em aplicações que já tenham configurado um número superior, o número de *threads* é limitado ao valor máximo de 100 e registra esta limitação em log.
- Quando a opção **Use Single Host Name** está habilitada, use sempre a opção **Max. Number of Threads** igual a 0 (zero), ou seja, leitura síncrona.
- A opção **Use Compatibility Mode** foi implementada neste Driver porque é relativamente comum encontrar servidores que não respondem corretamente quando recebem uma requisição de eco sem dados, seja por alguma diretiva de segurança ou por problemas de implementação. Nestes casos, pode ocorrer de este Driver não conseguir obter sucesso em um comando **ping** de servidores que retornam sucesso quando se usa o Prompt de Comando do Windows. Isto porque o comando **ping** do Windows, por padrão, envia 32 bytes de dados em um *frame* de requisição de eco, enquanto este Driver envia um comando **ping** vazio, ou seja, sem bytes de dados. Se esta condição é detectada nos servidores em uso, deve-se habilitar a opção **Use Compatibility Mode** e definir um número adequado de bytes de dados que possa ser aceito por um servidor na opção **Send n ICMP data bytes**. Recomenda-se iniciar com 32 bytes, que é o valor padrão do comando de **ping** no Prompt de Comando do Windows e é aceito pela maioria dos servidores. Caso contrário, recomenda-se manter esta opção desabilitada, mantendo a condição mais otimizada, sem bytes de dados.
- Para testar a necessidade de usar a opção **Use Compatibility Mode**, pode-se personalizar o número de bytes de dados enviados nas requisições de eco do comando **ping** no Prompt de Comando do Windows usando-se a opção **-l, Send buffer size** ou **Envia o tamanho do buffer**. Para mais informações, digite **ping /?** em um Prompt de Comando do Windows.

## Configuração das Propriedades Offline

Os parâmetros disponíveis na **configuração das propriedades** também podem ser acessados em tempo de execução se este Driver é iniciado no modo **Offline**, utilizando-se os parâmetros do tipo de dados **String** descritos na tabela a seguir.

### Parâmetros para configuração offline

PARÂMETRO	TIPO DE DADOS
<b>Ping.PingTimeout</b>	Inteiro sem sinal
<b>Ping.MaxNumberOfThreads</b>	Inteiro sem sinal
<b>Ping.PingScan</b>	Inteiro sem sinal
<b>Ping.WaitOnFirstRead</b>	Booleano (zero ou um)
<b>Ping.DisableInactiveTags</b>	Booleano (zero ou um)
<b>Ping.MissedScans</b>	Inteiro sem sinal
<b>Ping.DNSTimeout</b>	Inteiro sem sinal
<b>Ping.UseAsyncDNS</b>	Booleano (zero ou um)
<b>Ping.SingleHostName</b>	<b>String</b> com o nome de um <i>host</i>
<b>Ping.UseSingleHostName</b>	Booleano (zero ou um)
<b>Ping.UseCompatibleMode</b>	Booleano (zero ou um)
<b>Ping.NumberOfDataBytes</b>	Inteiro sem sinal
<b>Ping.PingDefaultRetries</b>	Inteiro sem sinal

Para mais informações sobre a configuração *offline* em tempo de execução, consulte o tópico **Documentação das Interfaces de Comunicação**.

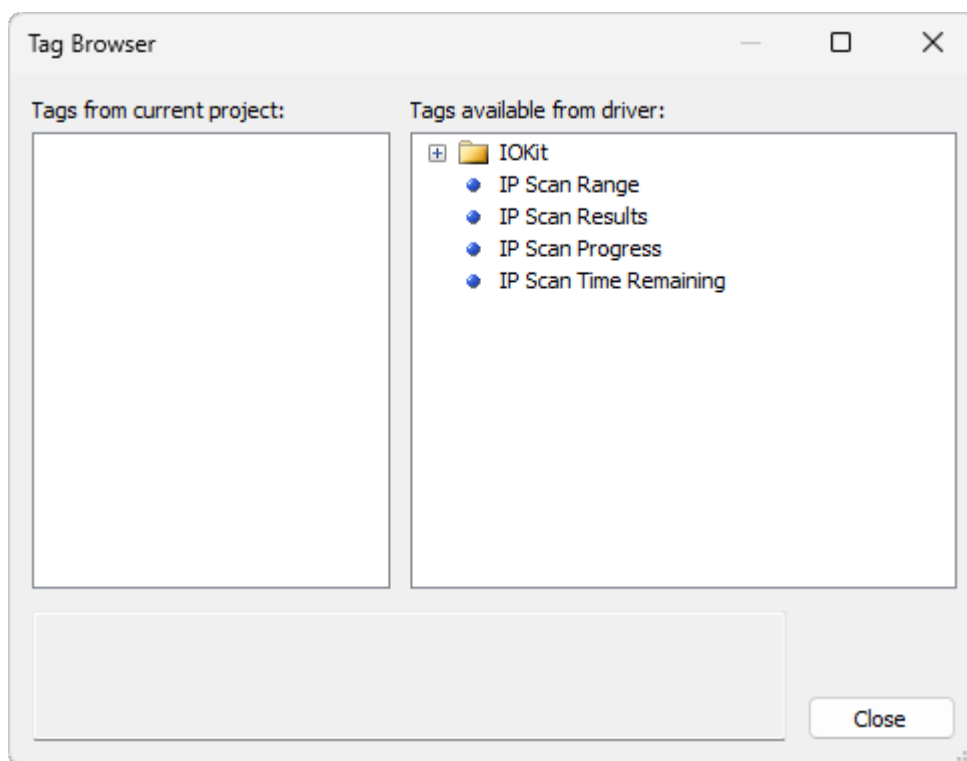
## Referência de Tags

Este Driver possui Tags simples de **ping** e também Tags para varrer endereços IP.

Os Tags **Simple de Ping** realizam comandos **ping** em *hosts* únicos, especificados através da configuração por **Strings** ou por parâmetros *N*.

Os Tags de **Varredura de Endereços IP** permitem apenas a configuração por **Strings**. Este recurso está disponível somente no **Elipse E3**, no **Elipse Power** e no **Elipse Water**, portanto não pode ser usado no **Elipse SCADA**.

Note que não é preciso digitar os parâmetros **Item** dos Tags de varredura, já que este Driver conta com o recurso de Tag Browser do **Elipse E3**, do **Elipse Power** ou do **Elipse Water**, mostrado na figura a seguir.



Janela Tag Browser

Para adicionar Tags de varredura em uma aplicação, arraste-os da lista **Tags disponibilizados pelo driver** (*Tags available from driver*) à direita para a lista **Tags do projeto corrente** (*Tags from current project*) à esquerda.

Os tópicos a seguir descrevem os diferentes tipos de Tags individualmente.

## Tag Simples de Ping

Os Tags **Simple de Ping** correspondem a um *host* ou endereço IP específico, ao qual um comando de **ping** é enviado a cada ciclo de varredura deste Driver.

Os Tags Simples de Ping são só de leitura e retornam o valor 0 (zero) se um comando **ping** falha ou 1 (um) se um comando **ping** foi executado com sucesso.

A especificação do endereço de um *host* pode ser realizada por nome ou diretamente por um endereço IP.

A detecção por endereço IP é mais rápida do que a detecção por nome de *host*, uma vez que a etapa de resolução de nome via DNS é eliminada.

A configuração dos Tags por endereço IP continua idêntica às versões anteriores à versão **2.0**. Já a configuração por nome, no **Elipse E3**, no **Elipse Power** e no **Elipse Water**, agora permite configurar vários Tags, e não é mais realizada na janela de configurações e sim no campo **Dispositivo** (*Device*) de cada Tag.

<b>N1</b>	Primeiro segmento de um endereço IP
<b>N2</b>	Segundo segmento de um endereço IP
<b>N3</b>	Terceiro segmento de um endereço IP
<b>N4</b>	Quarto segmento de um endereço IP

Por exemplo, para definir um endereço IP 200.248.184.102, deve-se configurar o parâmetro *N1* com o valor 200, o parâmetro *N2* com o valor 248, o parâmetro *N3* com o valor 184 e o parâmetro *N4* com o valor 102.

### Configuração por Nome de Host

Para configurar por nome de *host* no **Elipse E3**, no **Elipse Power** e no **Elipse Water**, adicione o nome de um *host* à propriedade **Dispositivo** (*Device*) deste Tag. Não é preciso alterar os demais parâmetros deste Tag. Pode-se definir vários Tags, com nomes diferentes.

Para o **Elipse SCADA**, use a opção **Use Single Host Name** para definir o nome de um *host*. Neste caso, só é possível definir um único Tag em uma aplicação, que deve ter todos os parâmetros *N* configurados em 0 (zero). Este comportamento é o mesmo da opção **Check by name** da versão **1.0** deste Driver.

## Tag de Varredura de Endereços IP

Este tópico descreve os Tags usados para a realização de uma varredura de endereços IP, em que é possível detectar todos os endereços IP válidos, ou seja, que respondem a comandos de **ping** em um determinado intervalo (*range*) de endereços de IP possíveis.

Os Tags de varredura são configuráveis apenas por **Strings**, recurso disponível apenas no **Elipse E3**, no **Elipse Power** e no **Elipse Water**, e não funcionam no **Elipse SCADA**.

O processo de varredura tem início com a escrita de uma **String** especificando um intervalo de endereços IP válidos em um Tag **IP Scan Range**.

Durante esta varredura, é possível monitorar o andamento através dos Tags **IP Scan Progress** e **IP Scan Time Remaining**.

Após o término deste processo, a lista dos endereços IP detectados, isto é, que responderam corretamente a um comando **ping**, pode ser obtida através da leitura do Tag **IP Scan Results**, que retorna esta lista usando o recurso de Tags reportados a eventos. Para mais informações sobre este recurso, consulte o *Manual do Usuário do Elipse E3*.

## IP Scan Range

### Leitura e Escrita

Este Tag define um intervalo de endereços IP que este Driver varre buscando endereços válidos, ou seja, que respondem a um comando de **ping**, e tem também a função de iniciar um procedimento de varredura.

Na leitura, caso haja uma varredura em andamento, retorna uma **String** que define o intervalo sendo varrido. Caso nenhuma varredura esteja em andamento, retorna uma **String** vazia.

Este Tag é configurado por **Strings** nos campos **Dispositivo** (*Device*) e **Item** do **Elipse E3**, do **Elipse Power** e do **Elipse Water**, e o intervalo é configurado no campo **Valor**. Os parâmetros *N3* e *N4*, entretanto, são também utilizados. Para isto, consulte a lista de parâmetros mais adiante neste tópico.

#### NOTA

A escrita de qualquer intervalo válido, descrita mais adiante neste tópico, no campo **Valor** deste Tag inicia imediatamente um processo de varredura.

Para cancelar uma varredura em andamento, escreva uma **String** vazia neste Tag.

O progresso de uma varredura pode ser acompanhado através dos Tags **IP Scan Progress** e **IP Scan Time Remaining**.

Após o término de uma varredura, a lista dos endereços IP detectados pode ser obtida através do Tag **IP Scan Results**.

A tabela a seguir contém a descrição dos parâmetros deste Tag.

#### Parâmetros do Tag IP Scan Range

PARÂMETRO	DESCRIÇÃO
<b>N3</b>	Determina o <i>time-out</i> de cada comando <b>ping</b> durante uma varredura. Se este parâmetro está configurado em 0 (zero), usa o <i>time-out</i> padrão configurado na opção <b>Ping Timeout</b>
<b>N4</b>	Determina o número de tentativas em cada endereço IP. Se este parâmetro está configurado em 0 (zero), assume o valor padrão configurado na opção <b>Ping Retries</b>
<b>Dispositivo</b>	Este parâmetro não é utilizado para este Tag e deve ser deixado em branco
<b>Item</b>	Este parâmetro deve ser configurado com o valor fixo "ScanIP.Range"
<b>Valor</b>	Deve-se escrever neste parâmetro uma <b>String</b> que define um intervalo a ser varrido em um processo de varredura, ou uma <b>String</b> vazia para cancelar qualquer varredura em andamento, se houver. A escrita de um intervalo válido dá início imediato à varredura

O intervalo pode ser definido utilizando-se os formatos descritos a seguir.

#### Notação CIDR (Classless Inter-Domain Routing ou Roteamento Interdomínio Sem Classes)

Nesta notação, pode-se adicionar a um endereço no formato **IPv4** (*Internet Protocol version 4*) um número antecedido por uma barra, que determina o número de bits invariáveis, a partir dos mais significativos no endereço IP de referência fornecido, conforme os exemplos a seguir.

- **192.168.10.0/24**: Escaneia 256 *hosts* entre 192.168.10.0 e 192.168.10.255
- **192.168.0.0/16**: Escaneia 65536 endereços IP (*hosts*) entre 192.168.0.0 e 192.168.255.255

#### Endereçamento Octet Range ou Intervalo de Octetos

Permite especificar intervalos diferentes para cada um dos 4 (quatro) octetos de um endereço no formato **IPv4**, usando um hífen entre os valores mínimos e máximos de um octeto. Se o valor à esquerda do hífen é omitido, ou valor mínimo, assume-se o valor 0 (zero). Se o valor máximo é omitido, assume-se o valor 255, conforme os exemplos a seguir.

- **192.168.0-255.1-254**: Varre todos os endereços IP iniciando em "192.168", mas ignorando os endereços IP que terminam em 0 (zero) ou 255 (quarto segmento)
- **192.168.0-.1-254**: Mesma configuração do exemplo anterior, assumindo o valor máximo para o terceiro segmento como 255
- **192.168.-255.1-254**: Mesma configuração dos dois exemplos anteriores, assumindo o valor mínimo para o terceiro segmento como 0 (zero)
- **0-255.0-255.13.37**: Varre toda a Internet buscando por endereços IP que terminem em "13.37"
- **0-.0-.13.37**: Mesma configuração do exemplo anterior
- **0--255.13.37**: Mesma configuração dos dois exemplos anteriores
- **0-0-0--10**: Varre toda a Internet buscando *hosts* cujo endereço no formato **IPv4** termine em um octeto com valores entre 0 (zero) e 10

## IP Scan Time Remaining

### Somente Leitura

Retorna o número de segundos previstos para que um processo de varredura termine. Este Tag é configurado por **Strings** nos campos **Dispositivo** (*Device*) e **Item** do **Elipse E3**, do **Elipse Power** e do **Elipse Water** e portanto não pode ser usado no **Elipse SCADA**. A tabela a seguir contém a descrição dos parâmetros deste Tag.

**Parâmetros do Tag IP Scan Time Remaining**

PARÂMETRO	DESCRIÇÃO
<b>Dispositivo</b>	Não usado, deixe em branco
<b>Item</b>	<b>String</b> de valor fixo "ScanIP.TimeRemainingSec"
<b>Valor</b>	Número de ponto flutuante de 64 bits ( <b>Double</b> ou IEEE754 de 64 bits), correspondendo à estimativa de segundos remanescentes para finalizar a varredura em andamento. Se não há varredura em andamento, ou se esta varredura já foi concluída, retorna 0 (zero)

## IP Scan Progress

### Somente Leitura

Retorna o percentual de progresso de uma varredura em andamento. Este Tag é configurado por **Strings** nos campos **Dispositivo** (*Device*) e **Item** do **Elipse E3**, do **Elipse Power** e do **Elipse Water** e portanto não pode ser usado no **Elipse SCADA**. A tabela a seguir contém a descrição dos parâmetros deste Tag.

**Parâmetros do Tag IP Scan Progress**

PARÂMETRO	DESCRIÇÃO
<b>Dispositivo</b>	Não usado, deixe em branco
<b>Item</b>	<b>String</b> de valor fixo "ScanIP.Progress"

PARÂMETRO	DESCRIÇÃO
<b>Valor</b>	Número de ponto flutuante de 64 bits ( <b>Double</b> ou IEEE754 de 64 bits) com um valor entre 0 (zero) e 100, correspondendo à percentagem já realizada de uma varredura, entre 0% e 100%. Se não há uma varredura ativa, retorna o valor 100

## IP Scan Results

### Somente Leitura

Este Tag é reportado a eventos e contém a lista de endereços IP detectados, ou seja, que responderam a um comando **ping** após o término da última varredura iniciada através da escrita de um intervalo válido no Tag **IP Scan Range**. Este Tag é configurado por **Strings** nos campos **Dispositivo** (*Device*) e **Item** do **Elipse E3**, do **Elipse Power** e do **Elipse Water** e portanto não pode ser usado no **Elipse SCADA**. A tabela a seguir contém a descrição dos parâmetros deste Tag.

#### Parâmetros do Tag IP Scan Results

PARÂMETRO	DESCRIÇÃO
<b>Dispositivo</b>	Não usado, deixe em branco
<b>Item</b>	<b>String</b> com o valor fixo "ScanIP.Results"
<b>Valor</b>	Os valores deste Tag são reportados a eventos ao final de um processo de varredura. Um a um, são retornados todos os endereços IP detectados como <b>Strings</b> no formato <b>IPv4</b> , com a ocorrência de um evento <b>OnRead</b> deste Tag a cada endereço IP retornado. Após o retorno de todos os endereços IP detectados, sempre retorna uma <b>String</b> vazia indicando o final da lista de endereços IP. Para mais informações sobre como tratar Tags reportados a eventos em uma aplicação, consulte o <i>Manual do Usuário do Elipse E3</i>

## Documentação das Interfaces de Comunicação

Esta seção contém a documentação das Interfaces de Comunicação referentes ao Driver **PingDrv**.

### Configurações de um Driver

A configuração das Interfaces de Comunicação é realizada na caixa de diálogo de configuração de um Driver. Para acessar a configuração da caixa de diálogo no **Elipse E3** na versão 1.0, siga estes passos:

1. Clique com o botão direito do mouse em um objeto Driver (IODriver).
2. Selecione o item **Propriedades** no menu contextual.
3. Selecione a aba **Driver**.
4. Clique em **Outros parâmetros**.

No **Elipse E3** versão 2.0 ou posterior, clique em **Configurar o driver**  na barra de ferramentas de um Driver. No **Elipse SCADA**, siga estes passos:

1. Abra o Organizer.

2. Selecione um Driver na árvore do Organizer.
3. Clique em **Extras** na aba **Driver**.

Atualmente, as Interfaces de Comunicação permitem que apenas uma conexão seja aberta para cada Driver. Isto significa que, no caso de acesso a duas portas seriais, é preciso adicionar dois Drivers em um aplicação e configurar cada um destes Drivers para cada porta serial.

## Caixa de Diálogo de Configuração

A caixa de diálogo das Interfaces de Configuração permite configurar a conexão de I/O que é utilizada por um Driver. Esta caixa de diálogo contém as abas **Setup**, **Serial**, **Ethernet**, **Modem** e **RAS** descritas nos tópicos a seguir. Se um Driver não implementa uma conexão de I/O específica, a respectiva aba não está disponível para configuração. Alguns Drivers podem conter abas adicionais, específicas para aquele Driver, na caixa de diálogo de configuração.

### Aba Setup

A aba **Setup** contém a configuração geral de um Driver. Esta aba é dividida nos seguintes grupos:

- **Configurações gerais:** Configurações da camada física de um Driver, *time-out* e modo de inicialização
- **Connection management:** Configurações de como a Interface de Comunicação mantém a conexão e qual a política de recuperação em caso de falha
- **Logging options:** Controla a geração dos arquivos de log

The screenshot shows the 'Setup' tab of a configuration dialog. It includes the following fields and options:

- Physical Layer:** A dropdown menu set to 'Ethernet'.
- Start driver OFFLINE:** An unchecked checkbox.
- Timeout:** A text box containing '1000' followed by 'ms'.
- Communication check time:** A text box containing '5000' followed by 'ms'.
- Connection management:** A section containing:
  - Mode:** A dropdown menu set to 'Automatic (managed by the driver)'.
  - Retry failed connection every:** A checked checkbox followed by a text box containing '20' and 'seconds'.
  - Give up after:** An unchecked checkbox followed by a text box containing '1' and 'failed retries'.
  - Disconnect if non-responsive for:** An unchecked checkbox followed by a text box containing '0' and 'seconds'.
- Logging Options:** A section containing:
  - Log to File:** An unchecked checkbox followed by a text box containing 'C:\eeLogs\MicrolokII\_%DATE%.log'.
  - File size limit (MB):** A text box containing '0' followed by '('0' is unlimited)'.

Aba Setup

### Opções gerais da aba Setup

OPÇÃO	DESCRIÇÃO
<b>Physical Layer</b>	Selecione a interface física em uma lista. As opções disponíveis são <b>Serial</b> , <b>Ethernet</b> , <b>Modem</b> e <b>RAS</b> . A interface selecionada deve ser configurada na aba específica
<b>Timeout</b>	Configure o <i>time-out</i> , em milissegundos, para a camada física. Esta é a medida de tempo que a interface de I/O aguarda para a recepção de um byte qualquer do <i>buffer</i> de recepção
<b>Communication check time</b>	Configure o tempo, em milissegundos, para definir o intervalo em que a comunicação é considerada em estado inativo. Enquanto um Driver de Comunicação receber dados válidos, o estado de comunicação é considerado ativo. Porém, se durante o funcionamento um Driver de Comunicação não receber dados válidos neste período de tempo, o estado é considerado inativo. O estado de comunicação é mostrado no Tag <b>IO.CommunicationStatus</b>
<b>Start driver OFFLINE</b>	Selecione esta opção para que um Driver inicie em modo <b>Offline</b> ou parado. Isto significa que a interface de I/O não é criada até que se configure um Driver em modo <b>Online</b> utilizando-se um Tag em uma aplicação. Este modo possibilita a configuração dinâmica da interface de I/O em tempo de execução

### Opções para o grupo Connection management

OPÇÃO	DESCRIÇÃO
<b>Mode</b>	Seleciona o modo de gerenciamento de conexão. Selecionar a opção <b>Automatic</b> permite que um Driver gerencie a conexão automaticamente, como especificado nas opções seguintes. Selecionar a opção <b>Manual</b> permite que uma aplicação gerencie a conexão completamente
<b>Retry failed connection every ... seconds</b>	Selecione esta opção para habilitar a retentativa de conexão de um Driver em um determinado intervalo, em segundos. Se a opção <b>Give up after failed retries</b> não está selecionada, este Driver continua retentando até que a conexão seja efetuada, ou que a aplicação seja parada
<b>Give up after ... failed retries</b>	Habilite esta opção para definir um número máximo de retentativas de conexão. Quando o número especificado de tentativas consecutivas de reconexão é atingido, um Driver vai para o modo <b>Offline</b> , assumindo que um problema de hardware foi detectado. Se um Driver estabelece uma conexão com sucesso, o número de retentativas sem sucesso é zerado. Se esta nova conexão é perdida, então o contador de retentativas inicia do zero
<b>Disconnect if non-responsive for ... seconds</b>	Habilite esta opção para forçar um Driver a se desconectar se nenhum byte chegou à interface de I/O no <i>time-out</i> especificado, em segundos. Este <i>time-out</i> deve ser maior que o <i>time-out</i> configurado na opção <b>Timeout</b>

### Opções para o grupo Logging Options

OPÇÃO	DESCRIÇÃO
<b>Log to File</b>	Habilite esta opção e configure o nome do arquivo onde o log é escrito. Arquivos de log podem ser bem extensos, portanto utilize esta opção por curtos períodos de tempo, apenas para o propósito de testes e depurações. Caso se utilize a macro <b>%PROCESS%</b> no nome do arquivo de log, esta é substituída pelo identificador do processo atual. Esta opção é particularmente útil ao se utilizar várias instâncias de um mesmo Driver no <b>Elipse E3</b> , permitindo assim que cada instância gere um arquivo separado de log. Por exemplo, ao configurar esta opção com o valor "c:\e3logs\drivers\sim_%PROCESS%.log", gera-se um arquivo c:\e3logs\drivers\sim_00000FDA.log para o processo <b>0FDAh</b> . Pode-se também utilizar a macro <b>%DATE%</b> no nome do arquivo. Neste caso é gerado um arquivo de log por dia, no formato <b>aaaa_mm_dd</b> . Por exemplo, ao configurar esta opção com o valor "c:\e3logs\drivers\sim_%DATE%.log", gera-se o arquivo c:\e3logs\drivers\sim_2005_12_31.log em 31/12/2005 e o arquivo c:\e3logs\drivers\sim_2006_01_01.log em 01/01/2006. De forma semelhante, a macro <b>%DATE_HOUR%</b> gera um arquivo de log por hora, no formato <b>aaaa_mm_dd_hh</b>
<b>File size limit (MB)</b>	Configure o limite de tamanho do arquivo de log, em megabytes. Um valor igual a 0 (zero) significa que não há limite de tamanho para o arquivo de log

## Configurações Gerais

Esta seção contém informações sobre a configuração dos **Tags de Comunicação** e das **Propriedades** gerais das Interfaces de Comunicação.

### Tags de Comunicação

#### Tags Gerais das Interfaces de Comunicação (N2/B2 = 0)

Os Tags descritos a seguir são fornecidos para todas as Interfaces de I/O suportadas.

#### IO.CommunicationStatus

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Leitura
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	0 (zero)
<b>Parâmetro N4</b>	6 (seis)
<b>Configuração por String</b>	IO.CommunicationStatus

Este Tag informa o estado da comunicação de um Driver. Indica o funcionamento da comunicação em função do recebimento de dados válidos dentro de um período de tempo arbitrado na configuração. Para mais informações, consulte o tópico **Aba Setup**. Os valores possíveis são **0 - Comunicação inativa**: O Driver não recebeu dados válidos ou deixou de receber dados depois de *n* milissegundos, conforme configurado na janela de propriedades, ou **1 - Comunicação ativa**: O Driver está recebendo dados válidos.

## IO.IOKitEvent

<b>Tipo de Tag</b>	Tag Bloco
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro B1</b>	-1 (menos um)
<b>Parâmetro B2</b>	0 (zero)
<b>Parâmetro B3</b>	0 (zero)
<b>Parâmetro B4</b>	1 (um)
<b>Propriedade Size</b>	4 (quatro)
<b>Propriedade ParamItem</b>	IO.IOKitEvent

Este Bloco retorna eventos de Driver gerados por várias fontes nas Interfaces de Comunicação. A propriedade **TimeStamp** de um Bloco representa o momento em que um evento ocorre. Os Elementos de Bloco são os seguintes:

- **Elemento 0**: Tipo de evento. Os valores possíveis são **0**: Informação, **1**: Advertência ou **2**: Erro
- **Elemento 1**: Fonte de um evento. Os valores possíveis são **0**: Driver (específico de um Driver), **-1**: IOKit (eventos genéricos da Interface de Comunicação), **-2**: Interface **Serial**, **-3**: Interface **Modem**, **-4**: Interface **Ethernet** ou **-5**: Interface **RAS**
- **Elemento 2**: Número do erro, específico de cada fonte de evento
- **Elemento 3**: Mensagem de um evento, uma **String** específica de cada evento

### NOTA

Um Driver mantém um número máximo de 100 eventos internamente. Se eventos adicionais são reportados, os eventos mais antigos são descartados.

## IO.PhysicalLayerStatus

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	0 (zero)
<b>Parâmetro N4</b>	2 (dois)
<b>Configuração por String</b>	IO.PhysicalLayerStatus

Este Tag indica o estado da camada física. Os valores possíveis são os seguintes:

- **0**: Camada física parada, ou seja, um Driver está em modo **Offline**, a camada física falhou ao inicializar ou excedeu o número máximo de tentativas de reconexão
- **1**: Camada física iniciada mas não conectada, ou seja, um Driver está em modo **Online**, mas a camada física não está conectada. Se a opção **Connection management** está configurada com o valor **Automatic**, a camada física pode estar conectando, desconectando ou esperando por uma tentativa de reconexão. Se a opção **Connection management** está configurada com o valor **Manual**, então a camada física permanece neste estado até ser forçada a conectar
- **2**: Camada física conectada, ou seja, a camada física está pronta para ser usada. Isto **NÃO** significa que um equipamento esteja conectado, apenas que a camada de acesso está funcionando

## IO.SetConfigurationParameters

<b>Tipo de Tag</b>	Tag Bloco
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro B1</b>	-1 (menos um)
<b>Parâmetro B2</b>	0 (zero)
<b>Parâmetro B3</b>	0 (zero)
<b>Parâmetro B4</b>	3 (três)
<b>Propriedade Size</b>	2 (dois)
<b>Propriedade ParamItem</b>	IO.SetConfigurationParameters

Use este Tag para modificar qualquer propriedade da caixa de diálogo de configuração de um Driver em tempo de execução.

Este Tag funciona somente enquanto um Driver está em modo **Offline**. Para iniciar um Driver em modo **Offline**, selecione a opção **Start driver OFFLINE** na caixa de diálogo de configuração deste Driver. Pode-se tanto escrever em um Tag PLC ou em um Tag Bloco contendo os parâmetros a serem modificados. As escritas de Elementos de Bloco individuais não são suportadas, um Bloco inteiro precisa ser escrito de uma vez só.

No **Elipse SCADA** é necessário usar um Tag Bloco. Cada parâmetro a ser configurado utiliza dois Elementos de Bloco. Por exemplo, caso seja necessário configurar 3 (três) parâmetros, então o tamanho do Bloco deve ser 6 (seis, 3 × 2). O primeiro Elemento é o nome da propriedade, como uma **String**, e o segundo Elemento é o valor desta propriedade, conforme o exemplo a seguir.

```
// 'Block' deve ser um Tag Bloco com leitura automática,
// leitura por varredura e escrita automática desabilitadas.
// Configura os parâmetros
Block.element001 = "IO.Type" // Parâmetro 1
Block.element002 = "Serial"
Block.element003 = "IO.Serial.Port" // Parâmetro 2
Block.element004 = 1
Block.element005 = "IO.Serial.BaudRate" // Parâmetro 3
Block.element006 = 19200
// Escreve o Bloco inteiro
Block.Write()
```

Ao usar o **Elipse E3**, a habilidade de criar *arrays* em tempo de execução permite o uso tanto de um Tag de Comunicação quanto de um Tag Bloco. Pode-se utilizar o método **Write** de um Driver para enviar os parâmetros diretamente para este Driver, sem a necessidade de criar um Tag, conforme o exemplo a seguir.

```
Dim arr(6)
' Configura os elementos do array
arr(1) = "IO.Type"
arr(2) = "Serial"
arr(3) = "IO.Serial.Port"
arr(4) = 1
arr(5) = "IO.Serial.BaudRate"
arr(6) = 19200
' Há dois métodos de enviar os parâmetros
' Método 1: Usando um Tag de Comunicação
tag.WriteEx arr
' Método 2: Sem utilizar um Tag
Driver.Write -1, 0, 0, 3, arr
```

Uma variação do exemplo anterior usa um *array* bidimensional.

```
Dim arr(10)
' Configura os elementos do array. Note que o array foi redimensionado
' para 10 elementos. Elementos vazios são ignorados pelo Driver
arr(1) = Array("IO.Type", "Serial")
arr(2) = Array("IO.Serial.Port", 1)
arr(3) = Array("IO.Serial.BaudRate", 19200)
Driver.Write -1, 0, 0, 3, arr
```

Um Driver não valida nomes de parâmetros ou valores passados, por isto tenha cuidado ao escrever parâmetros e valores. O método **Write** falha se o *array* de configuração é criado incorretamente. Pode-se consultar o log de um Driver ou usar o parâmetro *writeStatus* do método **WriteEx** para descobrir a causa exata de um erro.

```
Dim arr(10), strError
arr(1) = Array("IO.Type", "Serial")
arr(2) = Array("IO.Serial.Port", 1)
arr(3) = Array("IO.Serial.BaudRate", 19200)
If Not Driver.WriteEx -1, 0, 0, 3, arr, , , strError Then
    MsgBox "Falha ao configurar os parâmetros do Driver: " + strError
End If
```

## IO.WorkOnline

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Leitura ou Escrita
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	0 (zero)
<b>Parâmetro N4</b>	4 (quatro)
<b>Configuração por String</b>	IO.WorkOnline

Este Tag informa o estado atual de um Driver e permite iniciar ou parar a camada física. Os valores possíveis são os seguintes:

- **0 - Driver Offline:** A camada física está fechada ou parada. Este modo permite uma configuração dinâmica dos parâmetros de um Driver através do Tag **IO.SetConfigurationParameters**
- **1 - Driver Online:** A camada física está aberta ou em execução. Enquanto está em modo **Online**, a camada física pode ser conectada ou desconectada e o estado atual pode ser conferido no Tag **IO.PhysicalLayerStatus**

No exemplo a seguir, utilizando o **Elipse E3**, um Driver é colocado em modo **Offline**, a porta COM é modificada e então é colocado em modo **Online** novamente.

```
'Configura o Driver em modo Offline
Driver.Write -1, 0, 0, 4, 0
'Muda a porta para COM2
Driver.Write -1, 0, 0, 3, Array("IO.Serial.Port", 2)
'Configura o Driver em modo Online
Driver.Write -1, 0, 0, 4, 1
```

O método **Write** pode falhar ao configurar um Driver em modo **Online**, ou seja, escrevendo o valor 1 (um). Neste caso, este Driver permanece em modo **Offline**. A causa da falha pode ser:

- Tipo de camada física configurada incorretamente, provavelmente um valor inválido foi configurado para a propriedade **IO.Type**
- Este Driver pode ter ficado sem memória
- A camada física pode ter deixado de criar a *thread* de trabalho. Procure no arquivo de log pela mensagem "Failed to create physical layer thread!"
- A camada física não conseguiu inicializar. A causa da falha depende do tipo de camada física. Pode ser um número de porta serial inválida, falha ao inicializar o Windows Sockets ou falha ao inicializar o TAPI (modem), entre outras. A causa é gravada no arquivo de log

#### IMPORTANTE

Mesmo que a configuração de um Driver para o modo **Online** seja bem-sucedida, isto não significa necessariamente que a camada física esteja pronta para uso, ou seja, pronta para executar operações de entrada e saída com um equipamento externo. O Tag **IO.PhysicalLayerStatus** deve ser verificado para assegurar que a camada física esteja conectada e preparada para a comunicação.

## Propriedades

Estas são as propriedades gerais de todas as Interfaces de I/O suportadas.

### IO.ConnectionMode

**9** Controla o modo de gerenciamento da Conexão. Os valores possíveis são **0**: Modo automático, em que um Driver gerencia a conexão ou **1**: Modo manual, em que uma aplicação gerencia a conexão.

### IO.GiveUpEnable

Quando configurada para Verdadeiro, define um número máximo de tentativas de reconexão. Se todas as reconexões falharem, um Driver entra em modo **Offline**. Se configurada para Falso, um Driver tenta até que uma reconexão seja bem-sucedida.

### IO.GiveUpTries

**9** Número de tentativas de reconexão antes que esta seja abortada. Por exemplo, se o valor desta propriedade é igual a 1 (um), um Driver tenta apenas uma reconexão quando a conexão é perdida. Se esta falhar, este Driver entra em modo **Offline**.

### IO.InactivityEnable

Configure em Verdadeiro para habilitar e em Falso para desabilitar a detecção de inatividade. A camada física é desconectada se está inativa por um certo período de tempo. A camada física é considerada inativa apenas se é capaz de enviar dados mas não de recebê-los de volta.

## IO.InactivityPeriodSec

9 Número de segundos para a verificação de inatividade. Se a camada física está inativa por este período de tempo, então é desconectada.

## IO.RecoverEnable

☑ Configure em Verdadeiro para habilitar um Driver a recuperar conexões perdidas e em Falso para deixar um Driver em modo **Offline** quando uma conexão é perdida.

## IO.RecoverPeriodSec

9 Tempo de espera entre duas tentativas de conexão, em segundos.

### NOTA

A primeira reconexão é executada imediatamente após a conexão ser perdida.

## IO.StartOffline

☑ Configure em Verdadeiro para iniciar um Driver em modo **Offline** e em Falso para iniciar um Driver em modo **Online**.

### NOTA

Não faz sentido modificar esta propriedade em tempo de execução, já que esta só pode ser modificada quando um Driver já está em modo **Offline**. Para configurar um Driver em modo **Online** em tempo de execução, escreva o valor 1 (um) no Tag **IO.WorkOnline**.

## IO.TimeoutMs

9 Define o *time-out* da camada física, em milissegundos. Um segundo equivale a 1000 milissegundos.

## IO.Type

A Define o tipo de interface física utilizada por um Driver. Os valores possíveis são os seguintes:

- **N ou None**: Não utiliza uma interface física, ou seja, um Driver deve fornecer uma interface personalizada
- **S ou Serial**: Utiliza uma porta serial local (COM $n$ )
- **M ou Modem**: Utiliza um modem local, interno ou externo, acessado via TAPI (*Telephony Application Programming Interface*)
- **E ou Ethernet**: Utiliza um *socket* TCP/IP ou UDP/IP
- **R ou RAS**: Utiliza uma Interface **RAS** (*Remote Access Server*). Um Driver conecta-se a um equipamento RAS através da Interface **Ethernet** e então emite um comando **AT** (*dial*)

## Configuração de Estatísticas

Esta seção contém informações sobre a configuração dos **Tags de Comunicação** e das **Propriedades** das estatísticas das Interfaces de Comunicação.

## Tags de Comunicação

### Tags de Estatísticas das Interfaces de Comunicação (N2/B2 = 0)

Os Tags descritos a seguir mostram estatísticas para todas as Interfaces de Comunicação.

#### IO.Stats.Partial.BytesRecv

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	0 (zero)
<b>Parâmetro N4</b>	1101
<b>Configuração por String</b>	IO.Stats.Partial.BytesRecv

Este Tag retorna a quantidade de bytes recebidos na conexão atual.

#### IO.Stats.Partial.BytesSent

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	0 (zero)
<b>Parâmetro N4</b>	1100
<b>Configuração por String</b>	IO.Stats.Partial.BytesSent

Este Tag retorna a quantidade de bytes enviados na conexão atual.

#### IO.Stats.Partial.TimeConnectedSeconds

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	0 (zero)
<b>Parâmetro N4</b>	1102
<b>Configuração por String</b>	IO.Stats.Partial.TimeConnectedSeconds

Este Tag retorna o número de segundos que um Driver está conectado na conexão atual ou 0 (zero) se um Driver está desconectado.

## IO.Stats.Partial.TimeDisconnectedSeconds

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	0 (zero)
<b>Parâmetro N4</b>	1103
<b>Configuração por String</b>	IO.Stats.Partial.TimeDisconnectedSeconds

Este Tag retorna o número de segundos que um Driver está desconectado desde o término da última conexão ou 0 (zero) se um Driver está conectado.

## IO.Stats.Total.BytesRecv

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	0 (zero)
<b>Parâmetro N4</b>	1001
<b>Configuração por String</b>	IO.Stats.Total.BytesRecv

Este Tag retorna a quantidade de bytes recebidos desde que um Driver foi carregado.

## IO.Stats.Total.BytesSent

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	0 (zero)
<b>Parâmetro N4</b>	1000
<b>Configuração por String</b>	IO.Stats.Total.BytesSent

Este Tag retorna a quantidade de bytes enviados desde que um Driver foi carregado.

## IO.Stats.Total.ConnectionCount

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	0 (zero)
<b>Parâmetro N4</b>	1004
<b>Configuração por String</b>	IO.Stats.Total.ConnectionCount

Este Tag retorna a quantidade de conexões que um Driver já estabeleceu, com sucesso, desde que foi carregado.

## IO.Stats.Total.TimeConnectedSeconds

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	0 (zero)
<b>Parâmetro N4</b>	1002
<b>Configuração por String</b>	IO.Stats.Total.TimeConnectedSeconds

Este Tag retorna o número de segundos que um Driver permaneceu conectado desde que foi carregado.

## IO.Stats.Total.TimeDisconnectedSeconds

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	0 (zero)
<b>Parâmetro N4</b>	1003
<b>Configuração por String</b>	IO.Stats.Total.TimeDisconnectedSeconds

Este Tag retorna o número de segundos que um Driver permaneceu desconectado desde que foi carregado.

## Propriedades

Atualmente, não existem propriedades definidas especificamente para mostrar as estatísticas das Interfaces de Comunicação em tempo de execução.

## Histórico de Revisões do Driver

VERSÃO	DATA	AUTOR	COMENTÁRIOS
3.0.14	05/11/2025	M. Ludwig	<ul style="list-style-type: none"> <li>Corrigido um GPF (<i>General Protection Failure</i>) ao executar um comando <b>ping</b> (<i>Case 39008</i>).</li> </ul>
3.0.13	02/09/2025	M. Ludwig	<ul style="list-style-type: none"> <li>Driver atualizado para a biblioteca <b>IOKit</b> versão <b>3.0</b> e Visual Studio 2022 (<i>Case 37963</i>).</li> </ul>
3.0.12	18/09/2019	C. Mello	<ul style="list-style-type: none"> <li>Driver portado para o Visual Studio 2017 (<i>Case 27518</i>).</li> </ul>
3.0.11	11/05/2017	A. Hertzog	<ul style="list-style-type: none"> <li>Ajustes de compatibilidade para múltiplas instâncias (<i>Case 17678</i>).</li> </ul>
3.0.9	29/10/2014	A. Qites	<ul style="list-style-type: none"> <li>Implementada uma varredura de intervalos de endereços IP (<i>Case 15917</i>).</li> <li>Implementada uma opção de retentativa de comandos <b>ping</b> falhados antes de retornar um valor 0 (zero) (<i>Case 17412</i>).</li> <li>Driver portado para a biblioteca <b>IOKit</b> versão <b>2.0</b> (<i>Case 15977</i>).</li> </ul>
2.1.1	01/09/2011	A. Qites	<ul style="list-style-type: none"> <li>Criado um modo de compatibilidade para evitar problemas com servidores que não aceitam comandos <b>ping</b> sem bytes de dados (<i>Case 12077</i>).</li> <li>Corrigido um erro em que este Driver poderia retornar sucesso de um comando <b>ping</b> para servidores não existentes ao utilizar valores altos de <i>time-out</i> (<i>Case 12064</i>).</li> <li>Implementado um valor limite de 100 ao número máximo de <i>threads</i> (<i>Case 12445</i>).</li> <li>Pequenas otimizações de desempenho.</li> </ul>

VERSÃO	DATA	AUTOR	COMENTÁRIOS
2.0.1	24/04/2009	A. Quites	<ul style="list-style-type: none"><li>• Implementada a execução de comandos <b>ping</b> em paralelo em múltiplas <i>threads</i> (<i>Case 10210</i>).</li><li>• Driver portado para a biblioteca <b>IOKit</b>.</li></ul>
1.0.1		R. Haetinger	<ul style="list-style-type: none"><li>• Todas as publicações anteriores ao controle de revisões.</li></ul>

**Matriz**

Rua Mostardeiro, 322/Cj. 902, 1001 e  
1002

90430-000 — Porto Alegre — RS

Fone: (+55 51) 3346-4699

Fax: (+55 51) 3222-6226

E-mail: [elipse-rs@elipse.com.br](mailto:elipse-rs@elipse.com.br)

**Filial no Paraná**

Av. Sete de Setembro, 4698/1708

80240-000 — Curitiba — PR

Fone: (+55 41) 4062-5824

E-mail: [elipse-pr@elipse.com.br](mailto:elipse-pr@elipse.com.br)

**Filial no Rio de Janeiro**

Av. José Silva de A. Neto, 200/Bl. 4/Sl.  
109B

22250-044 — Rio de Janeiro — RJ

Fone: (+55 21) 2430-5912

Suporte Técnico: (+55 21) 2430-5963

E-mail: [elipse-rj@elipse.com.br](mailto:elipse-rj@elipse.com.br)

**Filial em São Paulo**

Rua dos Pinheiros, 870/Cj. 141 e 142

05422-001 — São Paulo — SP

Fone: (+55 11) 3061-2828

Fax: (+55 11) 3086-2338

E-mail: [elipse-sp@elipse.com.br](mailto:elipse-sp@elipse.com.br)

**Filial em Minas Gerais**

Rua Antônio de Albuquerque, 156/705

30112-010 — Belo Horizonte — MG

Fone: (+55 31) 4062-5824

E-mail: [elipse-mg@elipse.com.br](mailto:elipse-mg@elipse.com.br)

**Filial em Taiwan**

9F., No.12, Beiping 2nd St., Sanmin Dist.

807 — Kaohsiung City — Taiwan

Fone: (+886 7) 323-8468

Fax: (+886 7) 323-9656

E-mail: [evan@elipse.com.br](mailto:evan@elipse.com.br)

Consulte nosso website para informações sobre o representante do seu estado.

[www.elipse.com.br](http://www.elipse.com.br)

[kb.elipse.com.br](http://kb.elipse.com.br)

[forum.elipse.com.br](http://forum.elipse.com.br)

[www.youtube.com/elipsesoftware](http://www.youtube.com/elipsesoftware)

[elipse@elipse.com.br](mailto:elipse@elipse.com.br)



Gartner, Cool Vendors in Brazil 2014, April 2014.

Gartner does not endorse any vendor, product or service depicted in its research publications, and does not advise technology users to select only those vendors with the highest ratings. Gartner research publications consist of the opinions of Gartner's research organization and should not be construed as statements of fact. Gartner disclaims all warranties, expressed or implied, with respect to this research, including any warranties of merchantability of fitness for a particular purpose.

**Microsoft Partner**

Gold Independent Software Vendor (ISV)