Driver IEC 61850 (25 Conexões)

Nome do Arquivo	IEC61850.dll
Fabricante	International Electrotechnical Commission
Equipamentos	Servidores (IEDs) compatíveis com a norma IEC 61850 (ED1 e ED2)
Protocolo	IEC 61850 MMS sobre Ethernet TCP/IP
Versão	3.0.59
Última Atualização	03/11/2026
Plataforma	Win32
Dependências	IOKit versão 2.0
Leitura com Superblocos	Não
Nível	31298 (31201 com licença Elipse Power)

Introdução

O Driver IEC 61850 (25 Conexões) comunica com relés de proteção e outros dispositivos usando o protocolo IEC61850 sobre Ethernet TCP/IP (MMS). Este Driver permite:

- Comunicação com diversos equipamentos em um mesmo Driver
- Importação de Tags de equipamentos ou através de arquivos de configuração no formato **SCL**
- Suporte à mensagens sem confirmação de **Reports**, **Buffered** ou **Unbuffered**
- *Polling* de variáveis que não pertençam a **Reports**
- Suporte à informação de qualidade e estampa de tempo com precisão de 1 (um) milissegundo
- Coleta de arquivos de oscilografia no formato **COMTRADE**

Consulte os documentos a seguir, fornecidos juntamente com este manual, para mais informações sobre o funcionamento deste Driver:

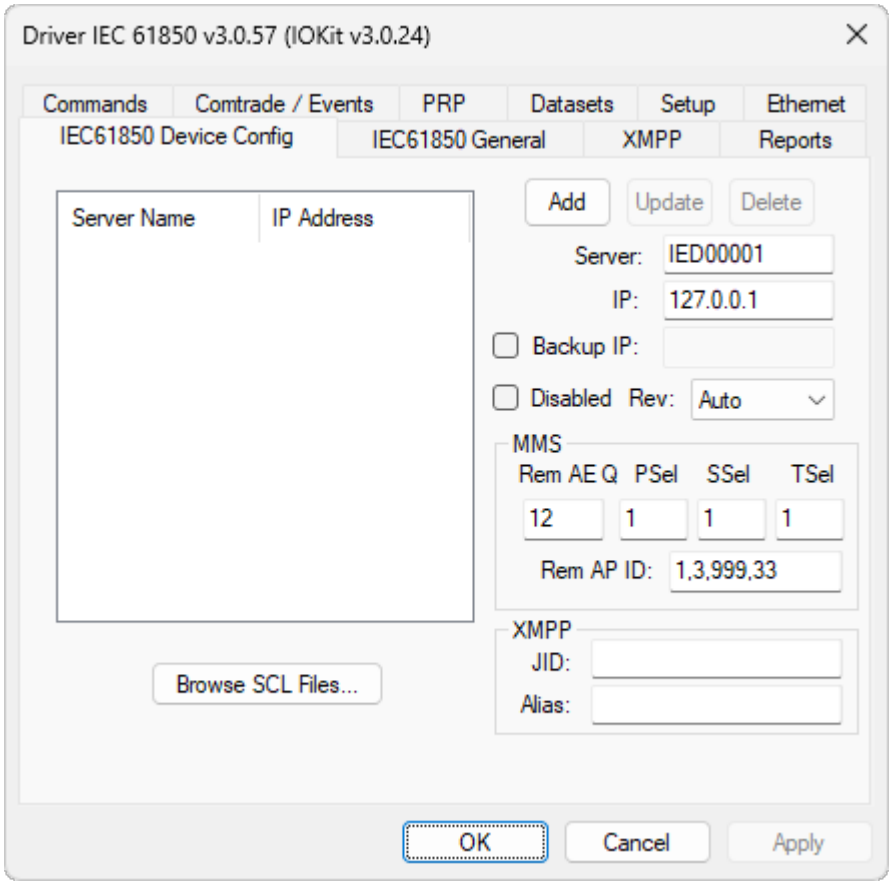
- Elipse IEC 61850 Client Driver PIXIT Ed2.pdf
- Elipse IEC 61850 Client Driver TICS Ed2.pdf
- Elipse IEC 61850 Client Driver PICS Ed2.pdf
- Elipse IEC 61850 Client Driver MICS Ed2.pdf

Configuração

Os parâmetros **[P]** de configuração deste Driver não são utilizados. Todas as configurações são realizadas na janela de configurações deste Driver. As abas de configuração estão descritas nos tópicos a seguir.

Aba IEC61850 Device Config

Esta aba permite a definição dos dispositivos (*Servers*) para estabelecer a comunicação.



Aba IEC61850 Device Config

As opções disponíveis nesta aba estão descritas na tabela a seguir.

Opções disponíveis na aba IEC61850 Device Config

OPÇÃO	DESCRIÇÃO
Browse SCL Files	Os Tags dos relés podem ser criados de forma *online*, ou seja, quando se pode comunicar com os relés, ou de forma *offline* através da importação de arquivos SCL. Esta opção abre uma janela para selecionar arquivos ICD ou SCD para importação. Após selecionar um arquivo, para cada descrição de relé encontrada cria-se uma identificação na lista de servidores, e também é criado um arquivo com a descrição dos **Logical Devices** (LDs) e **Logical Nodes** (LNs) de um relé, que podem ser importados para uma aplicação pela janela **Tag Browser**. Caso não haja arquivos SCL, pode-se configurar diretamente cada relé usando as opções **Add** (Adicionar), **Update** (Atualizar) e **Delete** (Apagar)
Server	Informe o nome de um dispositivo. Usado apenas para identificar um dispositivo para este Driver, mapeando-o para um endereço IP
IP	Informe o endereço IP de um equipamento. Opcionalmente é possível informar a porta TCP/IP, caso não seja a porta padrão do protocolo MMS, usando o formato **[0-255].[0-255].[0-255].[0-255]:[0-65535]**, como por exemplo 192.168.0.10:102
Backup IP	Informe o endereço IP de *backup* de um equipamento, se disponível. Use a mesma sintaxe da opção **IP**

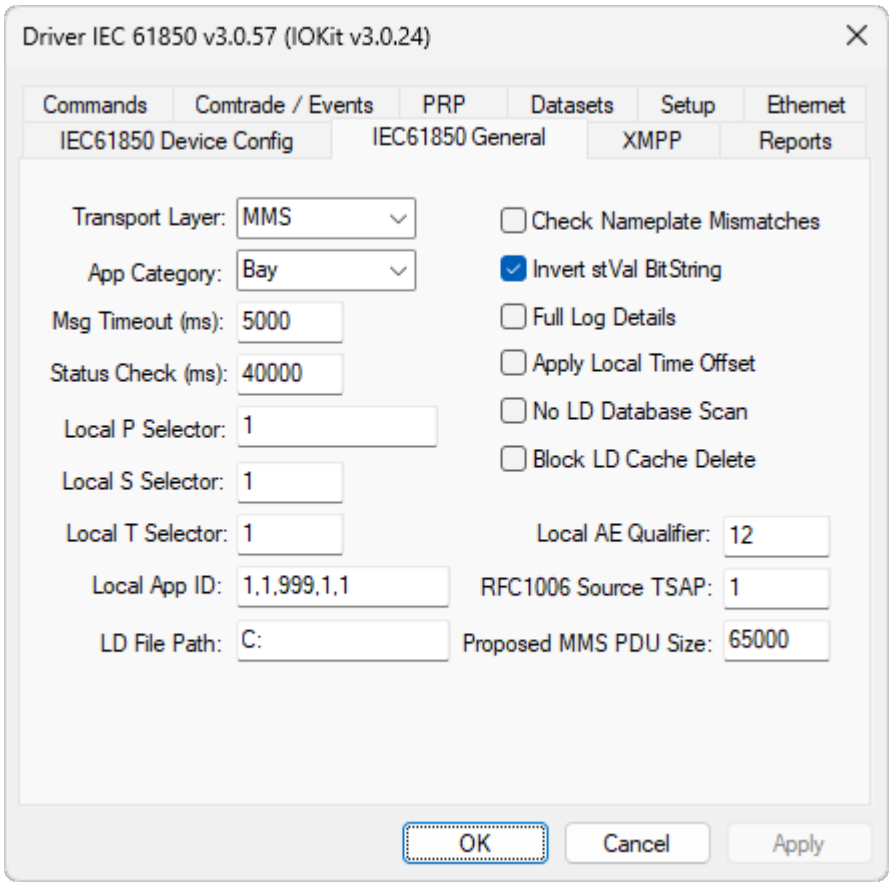
OPÇÃO	DESCRIÇÃO
PSel (Presentation Selector)	Valor de seleção para o dispositivo usado pela camada de apresentação OSI (ISO/IEC 8823). Deve ser consultado, se fixo, ou informado na configuração deste equipamento. Geralmente tem o valor 1 (um)
SSel (Session Selector)	Valor de seleção para o dispositivo usado pela camada de sessão OSI (ISO/IEC 8327). Deve ser consultado, se fixo, ou informado na configuração deste equipamento. Geralmente tem o valor 1 (um)
TSel (Transport Selector)	Valor de seleção para o dispositivo usado pela camada de transporte OSI (ISO/IEC 8073). Deve ser consultado, se fixo, ou informado na configuração deste equipamento. Geralmente tem o valor 1 (um)
Rem AP ID (Remote Application Process Identification)	Identificador utilizado pela camada de associação OSI (ISO/OSI 8650), em formato **ASN.1** (*Abstract Syntax Notation 1*). Indica a formatação dos dados adotados pela função **AARQ** (*Association Request*). Geralmente tem o valor 1,1,999,1,1 (iso.1.999.1.1)
Rem AE Qual (Remote Application Entity Qualifier)	Identificador utilizado pela camada de associação como formatador. Deve ser consultado, se fixo, ou informado na configuração de um equipamento. Geralmente tem o valor 12
Disable	Desabilita este dispositivo. Portanto, ao iniciar este Driver, não há comunicação com este dispositivo
Use Backup IP	Informa se o endereço IP de *backup* é usado
Revision	Revisão da Norma IEC61850 para este IED. Os valores disponíveis para esta opção são **Auto**: Este Driver tenta identificar automaticamente, **v1/v2**: Força a revisão 1 ou 2 ou **v2.1**: Força a revisão 2.1
XMPP	Permite definir o endereço JID (*Jabber ID*) e *Alias* (apelido) para acessar o dispositivo através do protocolo XMPP (IEC 61850-8-2).

NOTA

O tópico **Limitações deste Driver** contém informações sobre o número máximo de IEDs suportados por este Driver.

Aba IEC61850 General

Permite a definição dos demais comportamentos deste Driver.



Aba IEC61850 General

As opções disponíveis nesta aba estão descritas na tabela a seguir.

Opções disponíveis na aba IEC61850 General

OPÇÃO	DESCRIÇÃO
Transport Layer	Permite escolher o método de transporte, entre o padrão **MMS (IEC 61850-8-1)** e o **XMPP (IEC 61850-8-2)**
App Category	Para envio de comandos, este Driver usa a informação selecionada nesta opção para o preenchimento da propriedade **OrCat** (*Origin Category*), que especifica o tipo de aplicação enviando um comando, para efeitos de segurança ou resolução de conflitos de comandos. Os valores disponíveis são **Bay**, **Station**, **Remote** ou **Maintenance**
Msg Timeout (ms)	Tempo de espera para resposta de um comando ou mensagem completa, que pode ser formada por várias mensagens intermediárias. O tempo de espera de cada byte ou mensagem intermediária é definido na aba **Setup**. Para mais informações, consulte o tópico **Documentação das Interfaces de Comunicação**
Status Check (ms)	Intervalo para envio de mensagem de status, que deve ser respondida pelo IED. Na ausência de resposta, a saúde da conexão pode ser verificada forçando uma desconexão e reconexão do IED. O valor desta opção deve ser maior que o valor da opção **Msg Timeout (ms)** e menor que o intervalo de desconexão configurado na aba **Setup**. Para mais informações, consulte o tópico **Documentação das Interfaces de Comunicação**

OPÇÃO	DESCRIÇÃO
Local P Selector	Valor de seleção para este Driver, usado pela camada de apresentação OSI (ISO/IEC 8823)
Local S Selector	Valor de seleção para este Driver, usado pela camada de sessão OSI (ISO/IEC 8327)
Local T Selector	Valor de seleção para este Driver, usado pela camada de transporte OSI (ISO/IEC 8073)
Local App ID Local Application Process Identification	Identificador utilizado pela camada de associação OSI (ISO/OSI 8650), no formato **ASN.1** (*Abstract Syntax Notation 1*). Indica a formatação dos dados adotados pela função **AARQ** (*Association Request*). Geralmente tem o valor 1,1,999,1,1 (iso.1.999.1.1)
LD File Path	Diretório padrão onde este Driver armazena arquivos descritores de cada *Logical Device* encontrado, com o objetivo de acelerar o processo de inicialização. Na próxima inicialização, caso seja encontrado um arquivo no diretório especificado, cujo nome é dado pelo formato **SERVER_LDNAME.LD**, onde **SERVER** é o nome do equipamento e **LDNAME** é o nome do *Logical Device*, então este LD é descrito a partir deste arquivo. Este Driver oferece várias formas de detectar mudanças na base de dados de um IED, de forma a atualizar estes arquivos
Check Nameplate Mismatches	Habilita a verificação dos parâmetros de identificação de um IED (*nameplate*) ao iniciar a comunicação, de forma a comparar se houve alguma modificação. Em caso positivo, uma atualização do *cache*, ou reconstrução dos arquivos .LD, é realizada
Invert stVal BitString	A interpretação de *bitstrings* com dois ou mais bits pode ser alterada por esta opção, alterando o significado dos estados **Aberto** ou **Fechado** em uma aplicação. O comportamento padrão é obtido com esta opção desmarcada
Full Log Details	Habilita a inserção no log deste Driver, habilitado na aba **Setup**, de informações detalhadas sobre a notificação de eventos para qualquer Tag. Para mais informações, consulte o tópico **Documentação das Interfaces de Comunicação**
Apply Local Offset to Timestamps	As estampas de tempo adotadas pelo padrão IEC 61850 sempre se referem ao padrão **UTC** (*Coordinated Universal Time*). Através desta opção, indica-se que este Driver deve aplicar o *offset* local, ou seja, fuso horário e horário de verão, à estampa de tempo enviada por um equipamento
No LD Database Scan	Informa que não deve ser processada nenhuma solicitação de *Logical Device* ou *Logical Node* para os IEDs. Esta opção pode ser usada quando este Driver tem a função somente de transferência de arquivos
Block LD Cache Delete	Em casos onde o processo de inicialização é incompleto, ou quando alguma inconsistência é detectada entre os arquivos de *cache* (LD) e a base de dados atual de um IED ou Servidor, este Driver apaga e atualiza os arquivos de *cache*. Habilite esta opção para bloquear a remoção dos arquivos

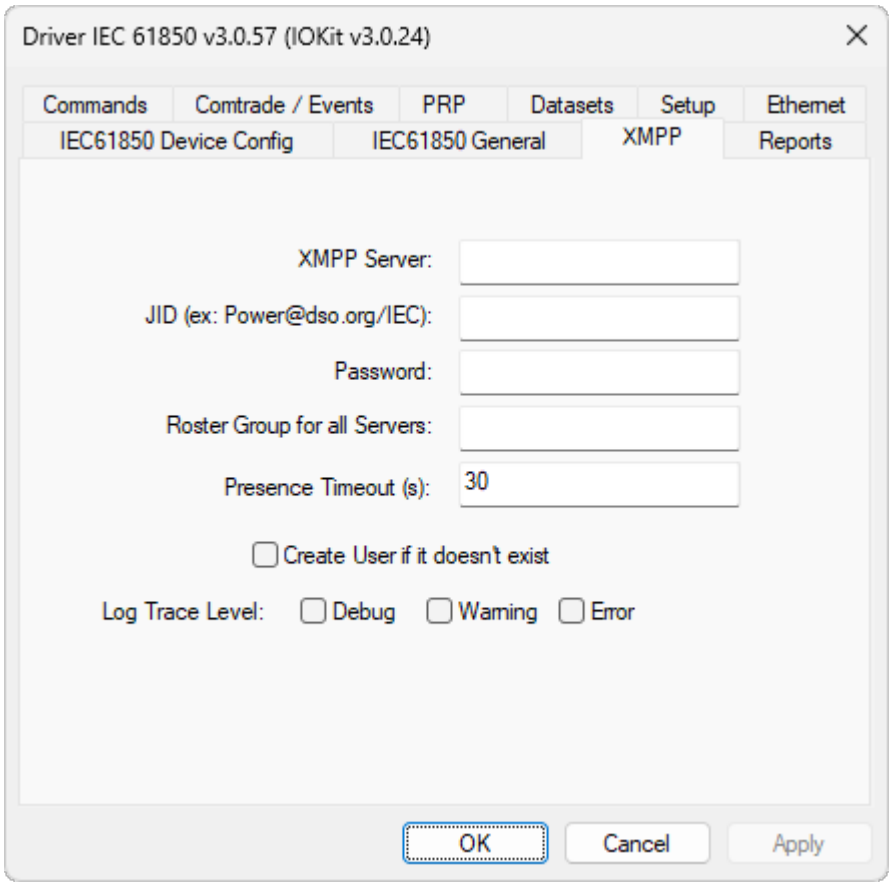
OPÇÃO	DESCRIÇÃO
Local AE Qual - Local Application Entity Qualifier	Identificador utilizado pela camada de associação como formatador. Geralmente tem o valor 12
RFC 1006 Source TSAP	Este Driver utiliza a especificação RFC 1006 como forma de transporte dos pacotes ISO sobre TCP. Para isto é necessário informar o TSAP (*Transport Service Access Point*) utilizado para estabelecer a conexão por este protocolo. O valor padrão é 1 (um)
Proposed MMS PDU Size	Quando a opção **Transport Layer** está configurada com o valor **MMS**, esta opção define o tamanho da PDU negociada com o servidor. O valor padrão desta opção é 65000 bytes

Aba XMPP

Realiza as configurações quando a camada de transporte selecionada é **XMPP**. A implementação da camada de transporte **XMPP** permite uma comunicação confiável, segura e adaptável, especialmente em sistemas que requerem troca de dados em tempo real, tais como aplicações de automação de subestações conforme a norma IEC 61850-8-2.

Para a conexão entre um Servidor e um Cliente IEC 61850 sobre XMPP, é necessário a utilização de pelo menos um Servidor XMPP, como por exemplo Ejabberd ou OpenFire. Cada Cliente ou Servidor IEC 61850 deve se conectar primeiramente a um Servidor XMPP utilizando uma conta, conhecida como JID (*Jabber ID*), no formato **jidname@dnsname/IEC** ou **jidname@serverIP/IEC**, como por exemplo "client@myxmppserver.org/IEC", e uma senha.

Cada conta JID possui associada um *Roster*, que é uma lista mantida no Servidor XMPP de outras contas JID cuja comunicação está habilitada ou autorizada. Cada conta JID também está organizada no Servidor XMPP em um grupo. No padrão da norma IEC 61850, o nome deste grupo deve ser igual a "61850".



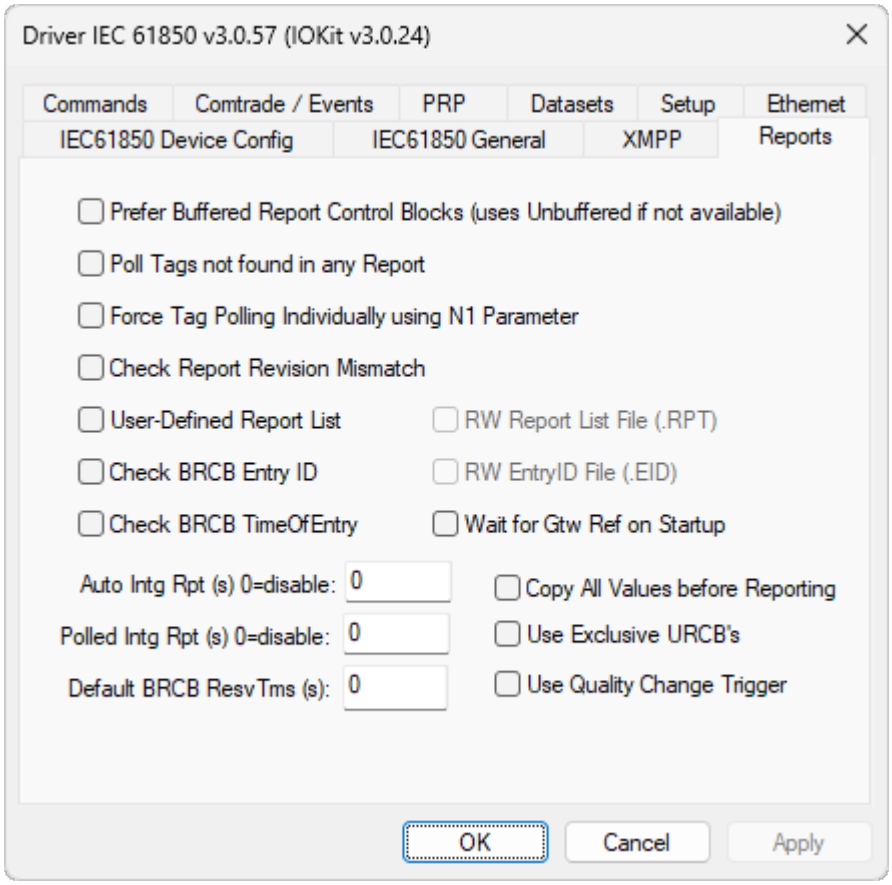
Aba XMPP

As opções disponíveis nesta aba estão descritas na tabela a seguir.

Opções disponíveis na aba XMPP

OPÇÃO	DESCRIÇÃO
XMPP Server	Endereço IP ou DNS de um Servidor XMPP
JID	*Jabber ID* utilizado por este Driver para conexão com um Servidor XMPP
Password	Senha associada ao *Jabber ID*
Roster Group for all Servers	O valor padrão desta opção é "61850"
Presence Timeout (s)	Define o intervalo padrão para a detecção da presença de comunicação. Após este intervalo, Tags são notificados com qualidade ruim
Accept all clients	Permite que qualquer Cliente se conecte, mesmo que um JID não esteja em um *Roster*. Esta opção não é recomendada para ambientes de produção
Create User if it doesn't exist	Permite que um usuário seja criado dinamicamente se não existir. Esta opção só tem efeito se um Servidor XMPP permite este tipo de operação e não é recomendada para ambientes de produção
Log Trace Level	Selecione os tipos de mensagens XMPP desejadas para logs, **Debug**, **Warning** ou **Errors**

Aba Reports



Aba Reports

As opções disponíveis nesta aba estão descritas na tabela a seguir.

Opções disponíveis na aba Reports

OPÇÃO	DESCRIÇÃO
Prefer Buffered Report Control Blocks (uses Unbuffered if not available)	O protocolo IEC 61850 prevê a utilização de objetos **Report** para notificar aplicações Clientes acerca de modificações nos dados. Os dados a serem reportados são definidos pelo usuário em um **Dataset**. Cada **Report** pode ter associado apenas um **Dataset** e é possível que um equipamento possua vários **Reports** e **Datasets**. Consulte a seção **Prefer Buffered Report Control Blocks** para mais informações sobre esta opção
Poll Tags not Found in any Report	Quando um Tag não é encontrado em nenhum **Report**, esta opção permite realizar uma leitura cíclica (*polling*) deste Tag, de acordo com a taxa de *scan*. Lembre-se que este método não é o mais recomendado nem o mais eficiente, pois está sujeito a uma maior lentidão de atualização dos dados e perda de eventos rápidos
Force Tag Polling Individually using N1 Parameter	Um pedido de *polling* para variáveis específicas pode ser definido com um valor diferente de 0 (zero) no parâmetro *N1* se esta opção está configurada
Check Report Revision Mismatch	Instrui este Driver a verificar a versão dos **Reports** ao iniciar a comunicação. Em caso de incompatibilidade de versões, uma atualização dos arquivos de *cache*, no formato **LD**, é realizada
User Defined Report List	Caso o usuário não deseje que este Driver realize uma busca automática por **Reports**, é possível definir uma lista de **Reports** que devem ser habilitados. Consulte a seção **User-Defined Report List** a seguir para mais informações sobre esta opção
RW Report List File (.RPT)	Indica se a lista de **Reports** definida na opção **User Defined Report List** deve ser salva em um arquivo, de forma que o script de exemplo da seção **User Defined Report List** não seja necessário. A escrita deste arquivo pode ser realizada automaticamente a partir da escrita em um Tag ou através da edição direta deste arquivo, que deve estar no diretório padrão dos arquivos de *cache*, no formato **LD**, e deve ter o nome do IED com a extensão .rpt. Só é permitido um IED por arquivo. Consulte a seção **Formato do Arquivo RPT** para mais informações
Check BRCB Entry ID	Através desta opção o usuário informa que, ao habilitar um **Buffered Report** (BRCB), deve-se configurar o parâmetro *EntryID* que contém um identificador único para cada mensagem de **Report** processada. Esta opção portanto permite reiniciar uma aplicação, ou quando há um *switch-over* de uma aplicação redundante, que um **Report** envie apenas as mensagens que ainda não foram processadas. Ao selecionar esta opção, é necessário que uma aplicação mantenha sincronizados os parâmetros *EntryID* de cada **Buffered Report** em uso e também realizar escritas na inicialização deste Driver, informando o último valor recebido. Para mais informações, consulte o tópico **Utilizando EntryID**

OPÇÃO	DESCRIÇÃO
RW EntryID List File (.EID)	Indica se os valores de *EntryID* expostos na opção **Check BRCB Entry ID** devem ser salvos em um arquivo, de forma que não seja necessária a execução de scripts para obter ou informar o valor de um campo na inicialização. Estes arquivos são salvos no diretório padrão dos arquivos de *cache*, no formato **LD**, e devem ter o nome de um IED com a extensão .eid. Só é permitido um IED por arquivo. Para mais informações, consulte o tópico **Utilizando Arquivos EID**
Check BRCB TimeOfEntry	Esta opção foi descontinuada e deve ser substituída pelo uso de *EntryID*
Wait for Gtw Ref on Startup	Através desta opção o usuário habilita o uso da funcionalidade de controle automático de *EntryIDs* para o uso em aplicações de *gateway*. Para mais informações, consulte o tópico **Configuração de Gateway**
Copy All Values Before Reporting	Se o **Dataset** associado a um **Report** contém objetos com qualidade e estampa de tempo depois das propriedades correspondentes, esta opção pode ser selecionada para evitar que a notificação de valor para uma aplicação aconteça sem a correta atualização da qualidade e da estampa de tempo
Auto Integrity Rpt (s)	Caso o valor informado seja diferente de zero (0), usa este valor ao habilitar um **Report** como o intervalo para que uma mensagem não solicitada de interrogação geral (GI) seja gerada por um equipamento
Polled Integrity Report (s)	Caso o valor informado seja diferente de zero (0), usa este valor para realizar a leitura de um **Report** via *polling*
Default BRCB ResvTms (s)	Valor padrão para a propriedade **ResvTms**, em segundos, usada para habilitar **Buffered Reports**. Este valor padrão é usado na revisão 2.1 da norma IEC 61850, definindo o tempo que um **Report** é reservado para este Cliente após uma queda de comunicação
Use Exclusive URCBs	Define se, ao habilitar um **Report Unbuffered** (URCB), este Driver é colocado em modo exclusivo, ou seja, a propriedade **Reserved** configurada para 1 (um), ou não, ou seja, a propriedade **Reserved** configurada para 0 (zero). Um URCB habilitado em modo exclusivo não pode ser utilizado por outros Clientes
Use Quality Change Trigger	Indica a opção de disparo por mudança de qualidade que deve ser usada por padrão, quando não está em uso a lista fixa de **Reports** ou se a opção **TrgOps** não foi informada

Prefer Buffered Report Control Blocks

Os **Reports** podem ser do tipo **Buffered** ou **Unbuffered**. O tipo **Buffered** significa que todas as alterações que ocorrem nos elementos de um **Dataset** durante uma desconexão são armazenadas em filas, ou *buffers*, de forma que uma aplicação Cliente recebe uma notificação de todas estas alterações, desde que exista espaço em memória suficiente em um equipamento e que as configurações estejam corretas. Este tipo de **Report** é usado principalmente para dados do tipo SOE (*Sequenciamento de Eventos*). Já os **Reports Unbuffered** nesta situação apenas armazenam o último valor.

Ambos os tipos podem ser configurados para enviar eventos espontaneamente, a intervalos cíclicos ou aguardar que uma aplicação Cliente requisite explicitamente, através de *polling* ou interrogação geral (GI), os dados daquele **Report**. Entretanto, vale ressaltar os seguintes pontos:

- Se duas aplicações Clientes, como por exemplo duas aplicações **Elipse Power**, estão conectadas em um equipamento, somente uma destas aplicações pode conectar-se a cada **Report Buffered**. Isto se deve ao fato de que, ao enviar dados, estes são apagados do *buffer* interno de um **Report**
- Duas ou mais aplicações Clientes podem se conectar a um mesmo **Report Unbuffered**, caso não estejam usando a opção de uso exclusivo

Desta forma, a opção **Prefer Buffered Report Control Blocks** instrui este Driver para que, sempre que um Tag entrar em processo de comunicação, ou seja, entrar em *advise* ou em *scan*, procure dentre todos os **Reports Buffered** de um equipamento se o Tag especificado pertence ao respectivo **Dataset** de cada **Report**.

Caso seja encontrado, então o **Report** é habilitado por este Driver, caso ainda não esteja habilitado, passando a receber notificações de mudanças. Caso negativo, este Driver passa a repetir o mesmo processo de busca, agora nos **Reports Unbuffered** pelo mesmo Tag. Se encontrado, o **Report** é habilitado, caso ainda não esteja habilitado. Caso novamente a resposta seja negativa, um Tag pode comunicar em modo *polling* se a opção **Poll Tags not found in any Report** está habilitada.

Caso a opção **Prefer Buffered Report Control Blocks** esteja desabilitada, este Driver repete o procedimento anterior, porém buscando diretamente os **Reports Unbuffered** e descartando a busca pelos **Reports Buffered**.

User-Defined Report List

- **TagName**: UserReportList
- **Device**: ServerName
- **Item**: UserDefinedReportList

```
Sub DRV_61850_AfterStart()
'O script a seguir monta um vetor
'contendo a configuração de Logical Device e o respectivo Report,
'os quais são ativados ao escrever no Tag UserReportList.
'Note que está sendo definido um vetor de duas posições.
'Caso seja necessário ativar mais Reports,
'Ajuste o tamanho do vetor

  Dim arr(1)
  arr(0) = Array("LogicalDeviceName", "LLN0$BR$brcbEV101")
  arr(1) = Array("LogicalDeviceName", "LLN0$BR$brcbEV102")
  Set Cmd = Application.GetObject("DriverName.IEDName.UserReportList")
  Cmd.WriteEx(arr)
End Sub
```

NOTA

Ao selecionar a opção **User Defined Report List**, este Driver só completa o processo de inicialização após receber a escrita de um Tag **UserReportList**.

Formato de um Arquivo RPT

```
Number_Of_LogicalDevices
LogicalDeviceN; Number_Of_Reports
ReportName; ReportOption1:ReportOption1Value; ReportOptionN:ReportOptionNValue
```

Opções disponíveis para o formato de arquivo RPT

PARÂMETRO	DESCRIÇÃO
Number_Of_LogicalDevices	Número total de *Logical Devices* neste arquivo
LogicalDeviceN;Number_Of_Reports	Para cada *Logical Device*, insira o nome e quantos **Reports** são utilizados
ReportName	Logo após o nome de cada *Logical Device*, segue uma lista com os **Reports** utilizados e campos opcionais
ReportOption;ReportValue	A tabela **Campos opcionais** contém os campos opcionais permitidos em um arquivo no formato **RPT**

Campos opcionais

CAMPO	DESCRIÇÃO
DatSet:DataSetName	Nome do **Dataset** que deve ser associado a um **Report**
TrgOps:Options	Opções de disparo para um **Report**. O valor **Options** corresponde a uma máscara de 6 (seis) bits com os valores **0**: Não usado, **1**: Data Change, **2**: Quality Change, **3**: Data Update, **4**: Integrity e **5**: GI (*General Interrogation*)
IntgPd:Period	Período, em milissegundos, para integridade
BufTm:Value	Período de temporização para aguardar novos eventos antes do envio, após a notificação do primeiro evento a transmitir, em milissegundos
Resv:Value	Uso de URCB em modo exclusivo. Os valores possíveis são **0** (zero) ou **1** (um)
ResvTms:Value	Tempo de espera, em segundos, após a desconexão para que o BRCB seja reservado para o mesmo Cliente conectado anteriormente
PurgeBuf:Options	Apaga o *buffer* de mensagens. Os valores possíveis são **0**: Not Set (do nothing), **1**: Set Always ou **2**: Set if EntryID Write Fails
RptID:RptName	Identificador de um **Report**
OptFlds:Options	Campos opcionais para a mensagem de **Report**. O valor **Options** corresponde a uma máscara de 10 bits com os valores **0**: Reserved, **1**: SequenceNumber*, **2**: ReportTimeStamp*, **3**: ReasonForInclusion*, **4**: DataSetName**, **5**: DataReference, **6**: BufferOverflow*, **7**: EntryID*, **8**: ConfRevision* e **9**: Segmentation. **NOTA**: Os valores marcados com * são usado por padrão quando o campo **OptFields** não é especificado. O campo **DataSetName** é obrigatório e os demais parâmetros são fixos e não podem ser modificados

Exemplo de um arquivo RPT.

```
1
Device;1
LLN0$BR$BRCB1;ResvTms:1000
```

User-Defined Datasets

- **TagName**: DeclareDSList
- **Device**: ServerName
- **Item**: DeclareClientDSList

Deve-se informar um *array* de descritores, cada descritor contendo um *array* com 3 (três) Elementos, o nome de um *Logical Device*, o nome de um **Dataset** (adicione o caractere @ no início do nome se volátil) e um índice único para cada **Dataset**.

```
Dim arr
arr = Array("LogicalDeviceName", "DataSetName", 1)
Set Cmd = Application.GetObject("DriverName.IEDName.DeclareDSList")
Cmd.WriteEx(arr)
```

- **TagName**: PopulateDSList
- **Device**: ServerName
- **Item**: PopulateClientDSList

Deve-se informar um *array*, e cada item deve conter um *array* com 2 (dois) Elementos, o índice de um **Dataset** e o nome de um LN/DO/DA no formato "LogicalDevice\LN$DO$DA".

```
Dim arr(4)
arr(0) = Array(1, "LDName\GGIO1$ST$stval")
arr(1) = Array(1, "LDName\GGIO2$ST$stval")
arr(2) = Array(1, "LDName\GGIO3$ST$stval")
arr(3) = Array(1, "LDName\GGIO4$ST$stval")
arr(4) = Array(1, "LDName\GGIO5$ST$stval")
Set Cmd = Application.GetObject("DriverName.IEDName.PopulateDSList")
Cmd.WriteEx(arr)
```

Formato de um Arquivo CDS

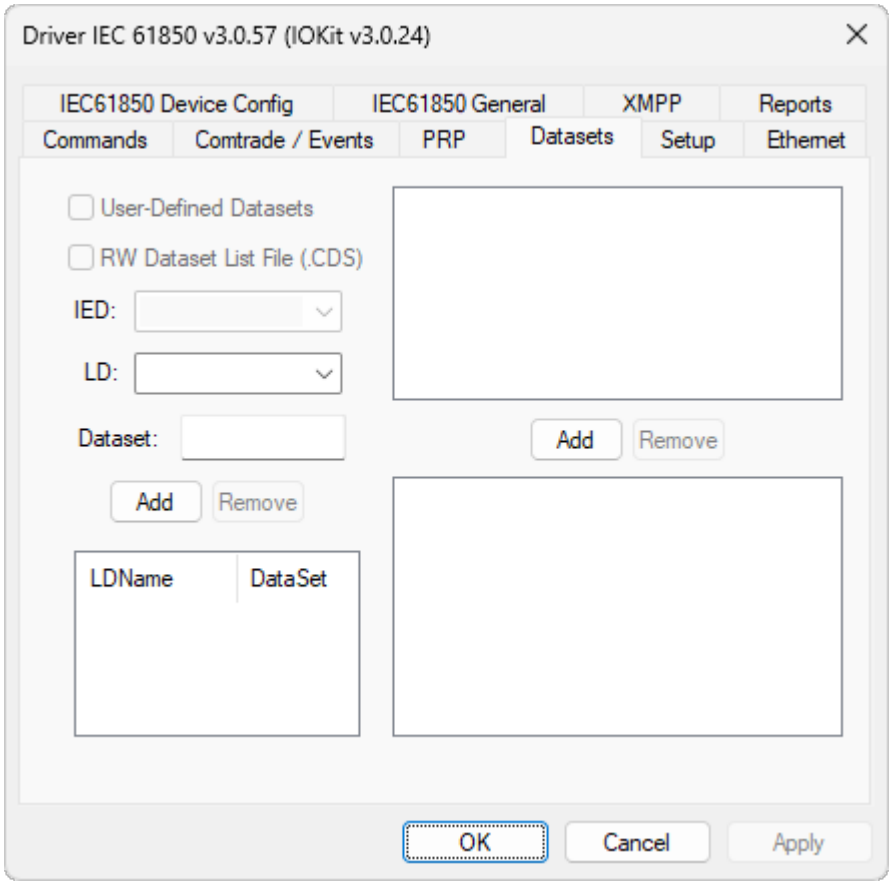
```
Number_Of_LogicalDevices
LogicalDeviceN; Number_Of_DataSets
DataSetName; NumberofMembers
MemberNames1...N
```

Exemplo de arquivo CDS.

```
1
Device;1
MyDataset1;3
Device/LLN0$DC$NamPlt
Device/LLN0$ST$Mod
Device/LPHD1$DC$PhyNam
```

Aba Datasets

Utilize esta aba para definir **Datasets** com o auxílio de um editor visual. Estes **Datasets** são salvos em arquivos com extensão .CDS e criados nos dispositivos de destino durante a inicialização.



Aba Datasets

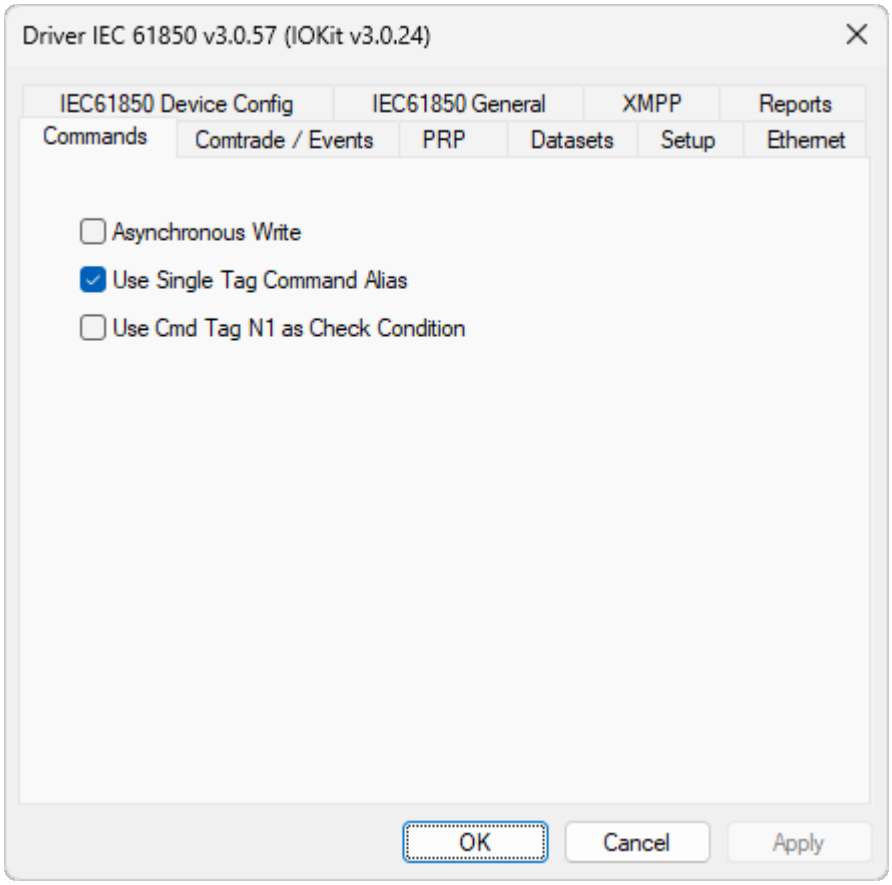
As opções disponíveis nesta aba estão descritas a seguir.

Opções disponíveis na aba Datasets

OPÇÃO	DESCRIÇÃO
User Defined Datasets	Ao utilizar uma lista pré-definida de **Reports**, pode-se informar se os **Datasets** são declarados dinamicamente por este Driver. Consulte a seção **User-Defined Datasets** para mais informações sobre esta opção
RW DataSet List File (.CDS)	Indica se a lista de **Datasets** definida no item anterior deve ser salva em um arquivo, de forma que a execução do script de exemplo da seção **User-Defined Datasets** não seja necessária. A escrita deste arquivo pode ser realizada automaticamente a partir dos exemplos anteriores ou através da edição direta deste arquivo, que deve estar no diretório padrão dos arquivos de cache (LD) e deve ter o nome do IED com a extensão .cds. Só é permitido um IED por arquivo. Consulte a seção **Formato do Arquivo CDS** para mais informações
IED	Permite selecionar um dispositivo para edição
LD	Permite selecionar um **Logical Device** no dispositivo selecionado na opção **IED** ou adicionar um **Dataset**
Dataset	Permite definir o nome de um **Dataset**
Add ou Remove Dataset	A opção **Add** adiciona um **Dataset** conforme o **Logical Device** selecionado e o nome de **Dataset** informado na opção **Dataset**. Para remover um **Dataset**, selecione-o e clique em **Remove**

OPÇÃO	DESCRIÇÃO
Add ou Remove Dataset Items	Com um **Dataset** selecionado na lista à esquerda, navegue pelos objetos dos **Logical Devices** existentes à direita e adicione itens usando a opção **Add**. Para remover um item, selecione-o e clique em **Remove**

Aba Commands



Aba Commands

As opções disponíveis nesta aba estão descritas na tabela a seguir.

Opções disponíveis na aba Commands

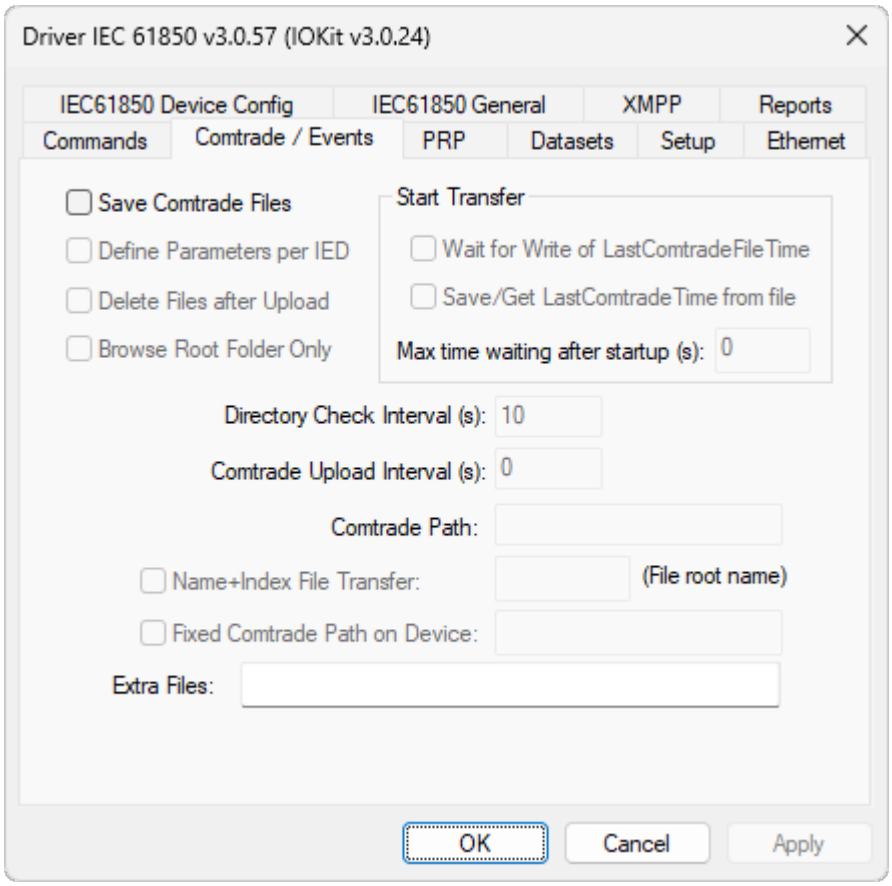
OPÇÃO	DESCRIÇÃO
Asynchronous Write	Permite que as solicitações de comandos retornem sucesso imediatamente, sem esperar pela resposta, desde que o envio tenha sido bem-sucedido, como por exemplo quando há uma conexão ativa. Este comportamento visa facilitar operações que envolvam o envio de um número muito grande de comandos, a fim de aumentar a velocidade geral. Entretanto, não verifica se houve alguma falha no processamento do comando em um equipamento
Use Single Tag Command Alias	Instrui este Driver a criar, durante o processo de *Tag Browsing*, um Tag único para o envio dos comandos. Para mais informações, consulte a **nota** a seguir. Se esta opção está desmarcada, utiliza-se um Tag Bloco com 5 (cinco) Elementos. Para mais informações, consulte o tópico **Tags de Comunicação**

OPÇÃO	DESCRIÇÃO
Use Cmd Tag N1 as Check Condition	No caso de utilizar um Tag simples para envio dos comandos, esta opção informa se o parâmetro *N1* deste Tag é usado para indicar o parâmetro de verificação de segurança de um comando. O parâmetro *N1*, neste caso, deve ser um número entre 0 (zero) e 3 (três), como resultado de um **OR** lógico entre os bits **0**: INTERLOCKING e **1**: SYNCHROCHECK

NOTA

Ao utilizar o **Elipse Power** como um Servidor OPC, um Tag de comando pode ter a leitura habilitada, ou seja, a propriedade **AllowRead** configurada em Verdadeiro. Isto permite que este Tag receba o valor inicial da propriedade **CtlVal** do objeto de comando referenciado, para que um Cliente OPC conheça o tipo de dados previamente. Após o envio de um comando, este Driver retorna uma leitura automática para o mesmo Tag, com o valor escrito caso a operação seja bem-sucedida. Desta forma, o parâmetro *WriteFeedBackMode* deste Driver, que indica como um Tag recebe a confirmação de uma operação mantendo o valor escrito, pode ser configurado em **0**: WaitNextRead, preferencialmente.

Aba Comtrade / Events



Aba Comtrade / Events

As opções disponíveis nesta aba estão descritas na tabela a seguir.

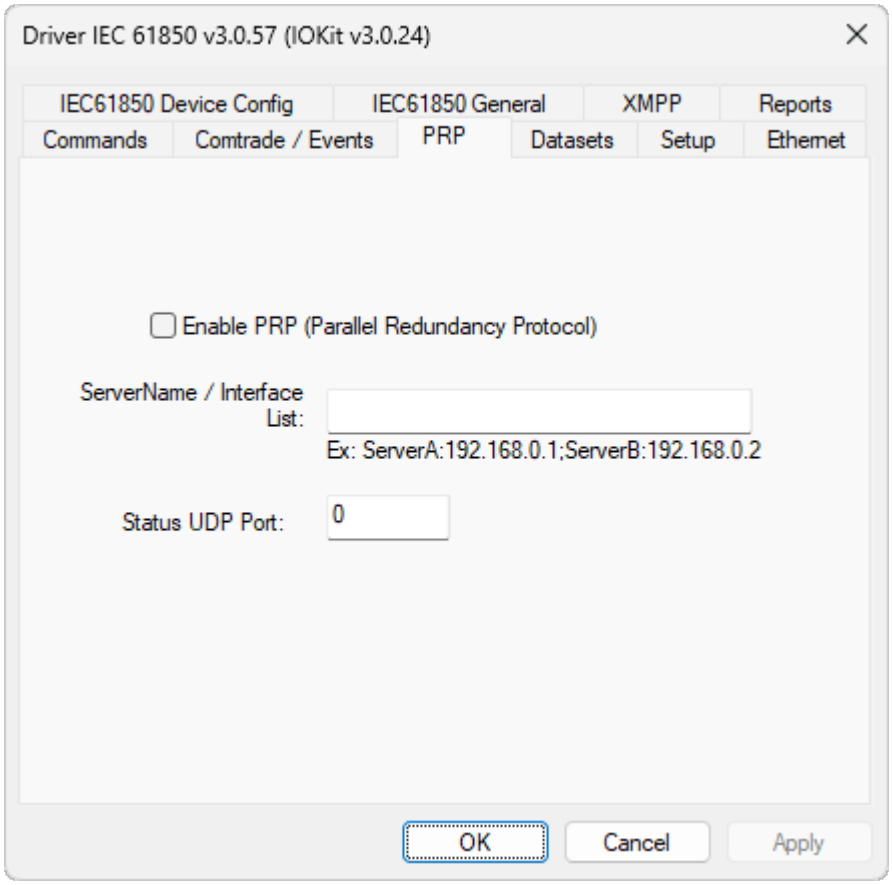
Opções disponíveis na aba Comtrade / Events

OPÇÃO	DESCRIÇÃO
Save Comtrade Files	Indica se este Driver deve realizar a busca por arquivos no formato **COMTRADE** (perturbações), salvando-os no diretório indicado na opção **Comtrade Path**
Define Parameters per IED	Indica se os parâmetros *Save Comtrade Files*, *Delete Files after Upload*, *Browse Root Folder Only*, *Comtrade Path*, *Use Fixed Comtrade Path on Device*, *Name+Index File Transfer* e *File Root Name* são definidos por IED e não globalmente para este Driver. A definição individual se dá através da escrita dos parâmetros internos deste Driver via biblioteca **IOKit**, conforme explicado no tópico **Configuração Dinâmica**
Delete Files After Upload	Instrui este Driver a apagar o arquivo de um equipamento após a conclusão com sucesso da transferência
Browse Root Folder Only	Alguns relés inserem arquivos no formato **COMTRADE** em uma pasta **COMTRADE** a partir do diretório raiz, e outros relés colocam estes arquivos no próprio diretório raiz, indicando como parte do nome do arquivo esta pasta, como por exemplo "\COMTRADE\Arquivo.cfg". Selecione esta opção caso um equipamento tenha este último comportamento
Wait for Write of LastComtradeFileTime	Com esta opção selecionada, este Driver espera por uma operação de escrita com o horário do último arquivo no formato **COMTRADE** transferido, por IED, de forma a comparar este horário com a lista de arquivos existentes em um IED, definindo assim qual arquivo deve ser transferido
Save/Get Last Comtrade time from File	Obtém o horário do último arquivo no formato **COMTRADE** transferido através de um arquivo COMTRADE.LST
Max Time Waiting after Startup	Indica quantos segundos este Driver deve esperar pela escrita de um Tag com o horário do último arquivo, ou pela leitura do arquivo COMTRADE.LST. Se este tempo expira, este Driver inicia a busca pelo primeiro arquivo disponível. Se o valor deste campo é igual a 0 (zero), a espera é infinita
Comtrade Directory Check Interval	Intervalo, em segundos, em que este Driver solicita a lista dos arquivos de um equipamento. Este Driver compara a data do último arquivo transferido com os arquivos da lista, e o arquivo ou arquivos mais recentes são transferidos
Comtrade Upload Interval (s)	No caso de transferir mais de um arquivo, este Driver aguarda o intervalo definido nesta opção para realizar cada transferência, a fim de separar uma janela de tempo específica para esta atividade. Entretanto, a recepção de eventos e o envio de comandos não são interrompidos enquanto um arquivo está sendo transferido
Comtrade Path	Indica o diretório onde este Driver salva os arquivos no formato **COMTRADE**

OPÇÃO	DESCRIÇÃO
Name+Index File Transfer	Indica que a transferência de arquivos se dá com base em um nome raiz mais um índice, ao invés da comparação de datas, que é o comportamento padrão. O nome raiz deve ser informado na opção **File Root Name**. Por exemplo, se um IED gera arquivos no padrão Osc1.dat/cfg, Osc2.dat/cfg e assim por diante, então deve-se selecionar esta opção e informar o texto "Osc" na opção **File Root Name**
Fixed Comtrade Path on Device	Indica que este Driver deve ignorar a busca pela pasta **COMTRADE** em um dispositivo, assumindo um diretório fixo informado na opção correspondente

Aba PRP

Esta aba permite configurar a operação deste Driver em uma rede redundante PRP (*Parallel Redundancy Protocol*).



Aba PRP

As opções disponíveis nesta aba estão descritas na tabela a seguir. Para mais informações, consulte o tópico **Redundância PRP**.

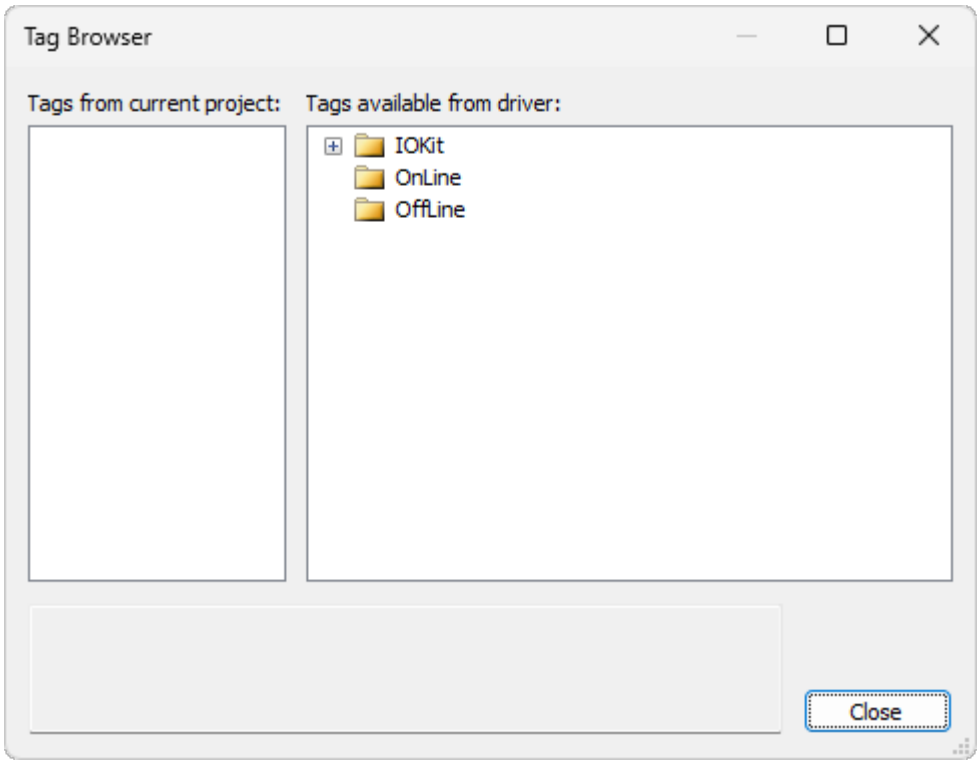
Opções disponíveis na aba PRP

OPÇÃO	DESCRIÇÃO
Enable PRP (Parallel Redundancy Protocol)	Indica se este Driver deve utilizar um serviço PRP. Esta opção é utilizada para que este Driver configure a interface de rede conforme a opção **ServerName / Interface List**, e também passe a escutar a porta de status UDP

OPÇÃO	DESCRIÇÃO
ServerName / Interface List	Deve ser informada uma lista de nomes de servidores (DNS) e endereços IP no formato **ServerA:IPA;ServerB:IPB**, para que um mesmo arquivo de projeto contendo este Driver possa ser usado sem alterações em ao menos 2 (dois) servidores. Quando este Driver inicia, obtêm-se o nome do computador local e verifica-se se há nesta lista um endereço IP que corresponda a este nome. Em caso positivo, este endereço IP é usado como o endereço IP da interface virtual
Status UDP Port	Informe uma porta local livre para ser utilizada por este Driver para receber estatísticas e logs do serviço ElipsePRPSvc.exe. Deve ser a mesma porta informada na configuração de um serviço PRP

Tag Browser

Clique em **Tag Browser** na barra de ferramentas deste Driver no **Elipse Power** para abrir a janela Tag Browser, que permite listar e arrastar para este Driver os Tags identificados em um equipamento, conforme a figura a seguir.



Janela Tag Browser

A listagem **Tags available from driver** (*Tags disponibilizados pelo driver*) contém as pastas a seguir:

- **IOKit**: Contém Tags padrão da biblioteca **IOKit** da **Elipse Software**, permitindo ler ou escrever parâmetros de conexão em geral e status, entre outros
- **OnLine**: Exibe os Servidores declarados na aba **IEC61850 Device Config**. Ao clicar em um Servidor, este Driver tenta a comunicação com um equipamento, criando uma pasta filha para cada *Logical Device* (LD) encontrado neste equipamento. Clique em um LD para buscar todos os Tags respectivos
- **OffLine**: Exibe todos os Servidores e LDs, porém clicar em um LD exibe os Tags a partir dos arquivos LD criados no diretório de importação, seja a partir da importação de arquivos SCL ou pela execução de uma aplicação

Para utilizar os Tags em uma aplicação, arraste-os da lista **Tags available from driver** (*Tags disponibilizados pelo driver*) para a lista **Tags from current project** (*Tags do projeto corrente*).

NOTA

Este Driver insere em cada *Logical Device* uma pasta **Datasets**, em que podem ser encontrados todos os **Datasets** daquele *Logical Device* referenciados por algum **Report**.

Sobre a Norma IEC 61850

A norma **IEC 61850** é um padrão internacional para comunicação em sistemas de automação de subestações elétricas, amplamente utilizado em redes inteligentes (*smart grids*). Esta norma define um modelo de comunicação robusto, flexível e interoperável para dispositivos de proteção, controle, monitoramento e medição em sistemas de energia.

Principais Características

- **Modelo de Dados Padronizado**: Utiliza objetos de dados (*Logical Nodes*) para representar funções e equipamentos, tais como disjuntores e transformadores, garantindo consistência e interoperabilidade
- **Comunicação Baseada em Ethernet**: Suporta protocolos como MMS (*Manufacturing Message Specification*), XMPP (*Extensible Messaging and Presence Protocol*), GOOSE (*Generic Object-Oriented Substation Event*) e SV (*Sampled Values*) para troca de dados em tempo real
- **Configuração via SCL**: Usa a linguagem SCL (*Substation Configuration Language*) para descrever a configuração de dispositivos e sistemas, facilitando a engenharia e manutenção
- **Alta Confiabilidade e Escalabilidade**: Projetada para atender requisitos de desempenho em ambientes críticos, com suporte a redundância e comunicação eficiente

Benefícios

- Interoperabilidade entre dispositivos de diferentes fabricantes
- Redução de custos de configuração e manutenção
- Suporte a aplicações avançadas, tais como automação e monitoramento em tempo real

A implementação da norma IEC 61850 em uma aplicação permite a integração eficiente com dispositivos compatíveis, garantindo operação confiável e adaptável em subestações modernas.

Camada de Transporte

A norma IEC 61850 foi originalmente desenvolvida para comunicação via Ethernet, utilizando o protocolo MMS, conforme especificado na norma IEC 61850-8-1, com foco em redes locais de subestações. Em 2018, a norma IEC 61850-8-2 introduziu uma alternativa baseada no protocolo XMPP (*Extensible Messaging and Presence Protocol*), voltada principalmente para comunicação externa à rede local, ampliando a aplicabilidade.

Desta forma, este Driver permite que um usuário selecione entre estes dois protocolos como camada de transporte.

Esta norma também define uma troca de dados sobre Ethernet conhecida como GOOSE (*Generic Object Oriented Substation Event*), que não está atualmente implementada neste Driver.

Estruturação das Informações

Os sistemas que implementam esta norma como servidores de dados, tipicamente IEDs ou relés de proteção, organizam informações com base nas seguintes entidades:

- **Logical Devices (LD)**: Representam dispositivos lógicos que correspondem a um dispositivo físico ou funcional, como uma baia (*Bay*) mapeada em um IED (*Intelligent Electronic Device*). Correspondem ao primeiro nível hierárquico em um Servidor IEC 61850, agrupando funções lógicas relacionadas para facilitar a modelagem e o gerenciamento
- **Logical Nodes (LN)**: São as representações virtuais das funções ou componentes de um dispositivo real, mapeadas dentro de um **Logical Device**. Por exemplo, uma chave (*Circuit Breaker*) em uma baia é representada por um **Logical Node** com a classe padrão **XCBR**. Um **Logical Device** geralmente contém múltiplos **Logical Nodes**, cada um correspondendo a uma função específica, como proteção, controle ou medição
- **Data Objects (DO) e Data Attributes (DA)**: As informações em um **Logical Node** são organizadas em **Data Objects**, que representam conjuntos de dados específicos relacionados a uma função. Cada **Data Object** é composto por **Data Attributes**, que descrevem propriedades detalhadas, tais como valores, estados ou configurações. O conjunto de **Data Attributes** de um **Data Object** segue uma estrutura definida chamada CDC (*Common Data Class*), garantindo padronização e interoperabilidade
- **Functional Constraint (FC)**: São os serviços específicos que podem ser utilizados em cada **Data Attribute**, e são parte do caminho, ou *path*, de um objeto

O formato utilizado por este Driver para todos os Tags que representam **Data Objects** ou **Data Attributes** é o seguinte:

- **Parâmetro Device**: Server:LD, como por exemplo "AL_07:AL_07PRO"
- **Parâmetro Item**: LNFCData$DataAttribute, como por exemplo "XCBR1STMod$stVal"

Referência de Tags

Esta seção contém informações sobre a configuração de **Tags de Comunicação** e **Tags Internos**.

Tags de Comunicação

Os parâmetros *N* dos Tags de Comunicação não são utilizados. Tags são endereçados apenas com os parâmetros *Device* e *Item*.

- **Device**: Server:LD, configurado na aba **IEC61850 Device Config**
- **Item**: LN$Data$DataAttribute ou LNFCData$DataAttribute, conforme a tabela a seguir

Opções disponíveis para o parâmetro Item

ITEM	OPERAÇÃO	SIGNIFICADO
LN$Data$DataAttribute ou LNFCData$DataAttribute	Leitura ou escrita	Tag qualquer para leitura de acordo com a configuração dos parâmetros de **Reports** ou *polling*. Se é um parâmetro de escrita, é aceito por este Driver
LNRPurcbXXXX ou LNRPbcrbXXXX	--	Este Driver não exibe a árvore correspondente aos **Reports**, para efeito de simplicidade e economia de Tags. Todo o tratamento é realizado internamente. A partir da versão **1.0.14**, este Driver passa a exibir o Tag **TimeofEntry**, leitura e escrita, somente nos **Buffered Reports**. A partir da versão **1.1.18**, passa também

ITEM	OPERAÇÃO	SIGNIFICADO
		a exibir o Tag **EntryID** nos **Buffered Reports**
LNCOxxxx ou LNSPxxxxx	Somente escrita	Bloco de comando ou controle. De forma a simplificar o uso de comandos, este Driver disponibiliza, no lugar da árvore de um objeto de controle, um Tag Bloco com 5 (cinco) Elementos, descritos na tabela **Elementos de um bloco de comando ou controle**. O resultado imediato da aceitação ou não de um comando pode ser obtido via script, através do método **WriteEx** de um Tag Bloco no parâmetro *wWriteStatus*, ou via bloco de status. Consulte o item a seguir para mais informações
LNCOxxxx$Oper, LN$CO$xxxx$TimeActOper, LNSPxxxxxSBO, LNCO$xxxx$SBOw ou LNCOxxxx$Cancel	--	Caso a opção **Use Single Tag Command Alias** esteja selecionada, este Driver cria na janela Tag Browser um único Tag no lugar de um Tag Bloco com 5 (cinco) Elementos. Neste caso, o envio de um comando é realizado conforme a seção **Envio de Comandos com Tag Único**
LNCOxxxx ou LNSPxxxxx	Somente leitura	Bloco de status de um comando. Para cada bloco ou Tag de comando é criado outro bloco, com 2 (dois) Elementos, que exibe o status das operações. Este status pode ser alterado como resultado da negação do envio de um comando, como por exemplo um comando não suportado, como resultado da aceitação, como por exemplo um comando aceito ou não aceito, ou como resultado da efetivação de uma ação, como por exemplo a abertura ou fechamento de uma chave. O primeiro Elemento deste Tag Bloco (**Status**) contém um código numérico e o segundo Elemento (**StatusText**) contém uma descrição textual deste código numérico. Consulte a tabela **Códigos numéricos de um bloco de status de comando** para mais informações

Elementos de um bloco de comando ou controle

ELEMENTO	DESCRIÇÃO
Operation	Deve receber a operação a ser realizada em formato **Texto**. Os valores possíveis são **OPERATE**, **SELECT**, **SELECTWITHVALUE** ou **CANCEL**

ELEMENTO	DESCRIÇÃO
Value	Valor a ser enviado, dependendo do tipo de objeto, na propriedade **CtlVal**
Time	Instante de tempo, no padrão Visual Basic em dias desde 1900, para uso em comando de **SELECT** por tempo e preenchimento da propriedade **T**
Test	Indica se o comando é uma operação de teste (valor igual a um) ou não (valor igual a zero)
Check	Tipo de verificação realizada antes da efetivação de um comando. Deve ser um número entre 0 (zero) e 3 (três) como resultado de um **OR** lógico entre os bits **0**: INTERLOCKING e **1**: SYNCHROCHECK

Códigos numéricos de um bloco de status de comando

STATUS	STATUSTEXT
0	Terminated OK
2	Select Accepted
3	Select With Value Accepted
4	Cancel Accepted
5	Operate Accepted
50	Terminate Fail
100	AppError:Unknown
101	AppError:NotSupported
102	AppError:BlockedBySwitchingHierarchy
103	AppError:SelectFailed
104	AppError:InvalidPosition
105	AppError:PositionReached
106	AppError:ParameterChangeInExecution
107	AppError:StepLimit
108	AppError:BlockedByMode
109	AppError:BlockedByProcess
110	AppError:BlockedByInterlocking
111	AppError:BlockedBySynchrocheck
112	AppError:CommandAlreadyInExecution
113	AppError:BlockedByHealth
114	AppError:1_Of_N_Control
115	AppError:AbortionByCancel
116	AppError:TimeLimitOver
117	AppError:AbortionByTrip

STATUS	STATUSTEXT
118	AppError:ObjectNotSelected
200	WriteError:Object_invalidated
201	WriteError:Hardware_fault
202	WriteError:Temporarily_unavailable
203	WriteError:Object_access_denied
204	WriteError:Object_undefined
205	WriteError:Invalid_address
206	WriteError:Type_unsupported
207	WriteError:Type_inconsistent
208	WriteError:Object_attribute_inconsistent
209	WriteError:Object_access_unsupported
210	WriteError:Object_non_existent
211	WriteError:Object_value_invalid
212	WriteError:Error_unknown
213	WriteError:Timeout
214	WriteError:OutOfMemory
215	WriteError:DecodeError
216	WriteError:WrongParameters
217	WriteError:CmdnotSupported

Envio de Comandos com Tag Único

- O valor de um Tag é o valor usado na propriedade **CtlVal**
- A estampa de tempo de um Tag é usada na propriedade **T**
- A propriedade **Test** tem o valor fixo 0 (zero)
- A propriedade **Check** tem o valor fixo 3 (três, INTERLOCKING e SYNCHROCHECK), exceto se o uso do parâmetro *N1* é habilitado na opção **Use Cmd Tag N1 as Check Condition**

Uma operação é definida na propriedade **Item** com os sufixos **Oper** (*Operate*), **SBO** (*Select Before Operate*), **SBOw** (*Select Before Operate With Value*) ou **Cancel**.

Um Tag para a operação de **TimeActivatedOperate**, com o sufixo **TimeActOper**, está disponível quando um **Logical Node** contém a propriedade **OpertTm**. Esta operação ocorre de forma semelhante ao comando **Operate**, com a diferença de que a estampa de tempo deste Tag é usada como o horário da operação e uma operação agendada pode ser cancelada com o uso de um Tag **Cancel**.

Tags Internos

DEVICE	ITEM	OPERAÇÃO	SIGNIFICADO
ServerName	DeviceConnectionStatus	Somente leitura	Retorna o status da conexão TCP/IP com um dispositivo. Os valores possíveis são **0**: Não conectado ou **1**: Conectado
ServerName	ServerStatus	Somente leitura	Retorna o status interno deste Driver. Se o parâmetro *Device* é igual apenas a *ServerName*, então o status é relativo a um equipamento como um todo. Os valores possíveis são **0**: Inicializando, **1**: Lendo arquivos .LD, **2**: Obtendo o diretório (**GetServerDirectory**), **3**: Processando um único LD (*Tag Browsing*), **4**: Obtendo o diretório de arquivos, **5**: Processando LDs, **6**: Concluindo, **7**: Finalizado OK, **8**: Concluindo com erro, **9**: Finalizado com erro, **10**: Reconectando ou **11**: Concluído (em espera)
ServerName:LDName	ServerStatus	Somente leitura	Se o parâmetro *Device* é igual a *ServerName:LDName*, então o status é relativo a um **Logical Device**. Os valores possíveis são **0**: Em espera, **1**: Lendo arquivo LD, **2**: Obtendo o diretório do LD, **3**: Obtendo o diretório do LN, **4**: Obtendo a definição de dados, **5**: Criando *cache* de Tags, **6**: Descobrindo **Reports**, **7**: Obtendo **Datasets**, **8**: Obtendo o conteúdo de **Datasets**, **9**: Obtendo *Nameplates*, **10**: Obtendo versões de **Reports**, **11**: Verificando versões, **12**: Programando **Reports**, **13**: Operação (*runtime*), **14**: Concluindo ou **15**: Concluindo com erro
ServerName	ComtradeTransferStatus	Somente leitura	Retorna o status da atividade de transferência de arquivos via COMTRADE. Os valores possíveis são **0**: Não conectado, **1**:

DEVICE	ITEM	OPERAÇÃO	SIGNIFICADO
			Conectado, **4**: Aguardando para listar os arquivos, **5**: Obtendo lista de arquivos, **6**: Arquivos transferidos (sincronizado), **7**: Aguardando para transferir arquivo, **8**: Transferindo arquivo, **101**: Lista vazia de arquivos em um relé, **102**: Erros de formatação no comando de leitura, **103**: Erro ao salvar um arquivo no formato **COMTRADE** ou **104**: Erro de transferência de arquivos
ServerName	LastComtradeFileName	Somente leitura	Informa o nome do último arquivo no formato **COMTRADE** transferido
ServerName	ServerRebuild	Somente escrita	Remove os arquivos de *cache* (LD) e reinicia a comunicação com um dispositivo. O parâmetro *Device* pode ser igual ao nome de um dispositivo, acompanhado ou não de um **Logical Device**
ServerName	UserDefinedReportList	Somente escrita	Permite a definição da lista de **Reports** do usuário
ServerName	LastComtradeFileTime	Leitura e escrita	Indica a data do último arquivo no formato **COMTRADE** coletado. Ao iniciar uma aplicação e evitar que arquivos já coletados sejam trazidos novamente, este Tag deve receber uma escrita com a data do último arquivo existente. Recomenda-se portanto que esta aplicação armazene o valor deste Tag ao sair e escreva este valor ao iniciar, ou se utilize a opção **Save/Get Last Comtrade Time from File**. Quando a opção **Name+Index File Transfer** está configurada, este Tag indica o último índice de arquivo a ser transferido, e pode da mesma forma ser lido e escrito. Importante observar que, caso um IED seja totalmente reiniciado, este índice pode voltar a um

DEVICE	ITEM	OPERAÇÃO	SIGNIFICADO
			valor inicial, como por exemplo 1 (um) e, para que este Driver volte a coletar arquivos a partir deste índice, é necessário escrever neste Tag este novo valor. **NOTA**: Arquivos são sobrescritos se têm nomes duplicados
ServerName	ComtradeInfo	Somente leitura	Retorna um Tag Bloco de 3 (três) Elementos com as propriedades COMTRADE. Os valores para os Elementos são **1**: ComtradeTransferStatus, **2**: LastComtradeFileTime e **3**: LastComtradeFileName
ServerName	BRCBActualList	Somente leitura	Retorna um Tag Bloco de 4 (quatro) Elementos com a lista de **Buffered Reports** em uso. Os valores para os Elementos são **1**: LDName, **2**: ReportName, **3**: TimeOfEntry e **4**: EntryID
ServerName	URCBActualList	Somente leitura	Retorna um Tag Bloco de 2 (dois) Elementos com a lista de **Unbuffered Reports** em uso. Os valores para os Elementos são **1**: LDName e **2**: ReportName
ElipseClientInfo	InternalClock	Somente leitura	Retorna um Tag Bloco de 2 (dois) Elementos com o horário corrente e a qualidade do relógio interno. Os valores para os Elementos são **1**: Horário corrente (TIME) e **1**: TimeQuality. O significado dos bits do Elemento 1 (um) é **7**: Leap Seconds Known, **6**: Not used, **5**: Sync Error e **4-0**: Precision (in power of 2-y)
-	ServerInitialBuild	Somente escrita	Permite modificar parâmetros de IEDs antes de iniciar a comunicação. Consulte a seção **Item ServerInitialBuild** para mais informações
-	GatewayRef	Somente escrita	Tag Bloco com 4 (quatro) Elementos que permite informar uma referência externa de *gateway* para um objeto 61850 usando uma

DEVICE	ITEM	OPERAÇÃO	SIGNIFICADO
			operação de escrita. Os Elementos são **1**: IEDName (nome do Servidor 61850 cadastrado neste Driver), **2**: LDName (nome de um **Logical Device**), **3**: Object (nome de um item 61850, como por exemplo "LLN0STData$DataAttr") e **4**: Reference (**String** que representa o endereço deste item em algum Driver externo, como por exemplo "202;34", indicando um objeto 2 (dois), variação 2 (dois) e índice 34 em um Driver **DNP Slave**. Para mais informações, consulte o tópico **Configuração de Gateway**
-	FinishedGtwRef	Somente escrita	Tag que permite informar, através de uma operação de escrita, que as operações de adição de referências de *gateway* (*GatewayRef*) chegaram ao fim. Para mais informações, consulte o tópico **Configuração de Gateway**
-	GatewayEventOK	Somente escrita	Tag Bloco com 4 (quatro) Elementos que permite informar que um evento de um Tag foi recebido por um protocolo externo em uma aplicação de *gateway*, permitindo assim que o último *EntryID* de **Report** seja ajustado de forma que não haja perda ou duplicação de eventos. Os Elementos são **1**: Reference (**String** que representa o endereço deste item em algum Driver externo, como por exemplo "202;34", indicando um objeto 2 (dois), variação 2 (dois) e índice 34 em um Driver **DNP Slave**, **2**: Value (valor do evento que foi recebido), **3**: TSValid (indica se a estampa de tempo é válida e está sendo utilizada no protocolo **Escravo**) e **4**: Timestamp (no formato de segundos desde

DEVICE	ITEM	OPERAÇÃO	SIGNIFICADO
			1970). Para mais informações, consulte o tópico **Configuração de Gateway**
-	DeleteAllDevices	Somente escrita	Remove todos os dispositivos da memória deste Driver e os remove de comunicação. Para que voltem a comunicar, é possível escrever no Tag **ServerInitialBuild**
ServerName ou Empty	GIAllReports	Somente escrita	Solicita um comando GI (*General Interrogation*) de todos os **Reports** habilitados por este Driver para um IED específico, colocando o nome deste IED na propriedade **Device**, ou para todos os IEDs deste Driver se a propriedade **Device** está vazia
ServerName	WorkOnline	Somente escrita	Os valores possíveis são **0**: Desconecta um IED ou **1**: Conecta um IED
ServerName	IPSelect	Somente leitura	Informa qual endereço IP está em operação, no caso de uso de endereços de IP principal e *backup*
ServerName	IOKitEvent	Somente leitura	Retorna eventos da biblioteca **IOKit** para o IED especificado, conforme o Tag **Read Driver Events** disponível no tópico **Documentação das Interfaces de Comunicação**
ServerName	GetFileDirectory	Somente leitura	Informa uma lista de arquivos existentes em um IED após uma solicitação de **GetFileDirectory** (escrita). Se não há uma solicitação previa ou não há dados disponíveis, retorna uma lista vazia
ServerName	FinishedWriteTimeOfEntry	Somente escrita	Consulte o tópico **Aba Reports** para mais informações
ServerName	FinishedWriteEntryID	Somente escrita	Consulte o tópico **Utilizando EntryID** para mais informações
ServerName	IPSwitch	Somente escrita	Solicita a troca do endereço IP principal para o endereço

DEVICE	ITEM	OPERAÇÃO	SIGNIFICADO
			de *backup* ou vice-versa
ServerName	DeclareClientDSList	Somente escrita	Declara a lista de **Datasets** definidos em um Cliente. Consulte o tópico **Aba Reports** para mais informações
ServerName	PopulateClientDSList	Somente escrita	Popula os membros de cada **Dataset** definido no item **DeclareClientDSList**. Consulte o tópico **Aba Reports** para mais informações
ServerName	GetFileDirectory	Somente escrita	Solicita uma lista de arquivos em um IED, baseado em um diretório informado no valor de um Tag ou no diretório raiz se nenhum diretório é informado
ServerName	GetFile	Somente escrita	Solicita a transferência de um arquivo com o nome informado em um Tag. Este arquivo é salvo com o mesmo nome no diretório padrão dos arquivos no formato **COMTRADE**
ServerName	DeleteFile	Somente escrita	Solicita a exclusão de um arquivo com o nome informado em um Tag diretamente em um IED
ServerName	CompareLogicalDeviceDirectory	Somente escrita	Compara o conteúdo de um **Logical Device** obtido a partir de um arquivo de *cache* (LD) com o conteúdo corrente em um IED. Em caso de discordância, um novo arquivo de *cache* é gerado
ServerName	ReportItemsUnified	Somente leitura	Um Tag Bloco contendo um resumo de todos os eventos recebidos por **Reports** do IED informado na propriedade **Device**, com os Elementos **1**: Logical Device Path, **2**: Item Path, **3**: Report Path, **4**: Dataset Path, **5**: EntryID (**String**), **6**: ReasonCode (**DWord**, consulte o **texto** a seguir para mais informações) e **7**: Qualidade, Valor e Estampa de Tempo (**String**)

DEVICE	ITEM	OPERAÇÃO	SIGNIFICADO
ServerName:LDName	CompareDataDefinition	Somente escrita	Compara o conteúdo de um **Logical Node** obtido a partir de um arquivo de *cache* (LD) com o conteúdo corrente em um IED. Em caso de discordância, um novo arquivo de *cache* é gerado
ServerName:LDName	GetDataValues	Somente escrita	Solicita a leitura de um item, que pode ser **LN**, **DO** ou **DA**. Os valores são retornados nos respectivos Tags
ServerName:LDName	SetDataValues	Somente escrita	Solicita a escrita dos valores correntes do item informado, que pode ser **LN**, **DO** ou **DA**
ServerName:LDName	GetDataSetValues	Somente escrita	Solicita a leitura de todos os itens pertencentes a um **Dataset**. Os valores são retornados nos respectivos Tags
ServerName:LDName	GetAllDataValues	Somente escrita	Solicita a leitura de todos os itens pertencentes a um **Functional Constraint**. O valor de um Tag deve estar no formato **LN$FC**. Os valores são retornados nos respectivos Tags
ServerName:LDName	DeleteDataSet	Somente escrita	Solicita a exclusão de um **Dataset** informado em um Tag

O Elemento 6 (seis, **ReasonCode**) do Tag Bloco do item **ReportItemsUnified** usa os bits a seguir para informar o motivo da transmissão, começando com o bit 0 (zero) como menos significativo.

```
TrgOpt_DataChange = 1,
TrgOpt_QualityChange = 2,
TrgOpt_DataUpdate = 3,
TrgOpt_Integrity = 4,
TrgOpt_GI = 5,
```

Item ServerInitialBuild

No exemplo a seguir, supondo a existência de um Tag **IniBuild** com o parâmetro *Item* igual a **ServerInitialBuild**, um IED de índice 0 (zero) é habilitado e o parâmetro *TSel* é modificado.

```
Dim CommArr(1)
CommArr(0) = Array("IEC61850.Device[0].Disable", 0)
CommArr(1) = Array("IEC61850.Device[0].TSel", 1)
Write -1, 0, 0, 3, CommArr
Item("IniBuild").WriteEx(1)
```

Redundância

As formas deste Driver evitar o recebimento de eventos duplicados na inicialização são as seguintes:

- Através da definição do último **EntryID** de cada **Report** por script
- Através da definição do último **EntryID** através da geração automática de um arquivo com extensão .eid

Os tópicos a seguir descrevem cada uma destas formas.

Utilizando EntryID

Ao selecionar a opção **Check BRCB EntryID** na aba **Reports** sem a opção dos arquivos automáticos (EID), uma aplicação deve realizar os procedimentos a seguir para o correto funcionamento:

1. Crie nesta aplicação, ou importe via **Tag Browser**, os Tags **EntryID** que estão disponíveis em cada **Buffered Report** e em uso por este Driver. O valor deste Tag é do tipo **Texto** e representa uma sequência de bytes.
2. Caso não saiba se um **Buffered Report** está em uso ou não, consulte a lista de BRCBs conforme exposto no tópico **Aba Reports**.
3. Crie nesta aplicação um Tag Interno para cada **EntryID** a ser sincronizado. Este Tag Interno deve ter a propriedade **Retentive** configurada para Verdadeiro. **NOTA**: Também é possível realizar este procedimento através de Propriedades de XObjects, que podem ter a propriedade **Retentive** habilitada diretamente.
4. Cada Tag Interno, ou Propriedade de XObject, deve receber as mudanças de valor do **EntryID** correspondente via script ou Associação.
5. Na inicialização desta aplicação deve haver um script que escreva o valor de cada **EntryID** de acordo com o valor do respectivo Tag Interno, que deve estar atualizado de acordo com o valor enviado pela estação redundante. Ao final do processo, um Tag especial deve ser escrito indicando que o processo de escrita das propriedades **EntryID** chegou ao final e que este Driver pode prosseguir com a inicialização. Este Tag deve ser configurado com as propriedades a seguir:
   - **Name**: FinishedWriteEntryID
   - **Device**: ServerName
   - **Item**: FinishedWriteEntryID

Basta uma operação qualquer de escrita para que este processo seja aceito.

NOTA

Se a opção de verificação de **EntryID** está configurada, este Driver só prossegue com a inicialização após a escrita deste Tag.

Utilizando Arquivos EID

Ao selecionar a opção **R/W EntryID File (.EID)** na aba **Reports**, este Driver gera um arquivo para cada IED com a extensão .eid, contendo os últimos **EntryIDs** recebidos para cada **Report** em uso. Este arquivo é gerado em até dois segundos após a recepção de uma mensagem de **Report** e também antes de finalizar (*shutdown*) uma aplicação. Este arquivo é gerado no mesmo diretório dos arquivos de *cache* (LD).

Na inicialização deste Driver, uma aplicação espera pela disponibilidade deste arquivo para prosseguir, a fim de obter os valores de **EntryID** processados até o momento em que esta operação foi interrompida. Em um sistema redundante, uma aplicação deve ser responsável por sincronizar este arquivo entre os computadores principal e *standby*.

O exemplo a seguir, que deve ser executado ciclicamente, copia os arquivos de **EntryID** entre servidores redundantes. Como este Driver é carregado pelo módulo **IOServer**, que executa sob a conta **System**, o código de exemplo utiliza o programa **psExec** (**SysInternals**) para elevação dos privilégios do usuário do Windows, ao executar um arquivo *batch* (.bat) que especifica o usuário e a senha.

NOTA

Para mais informações sobre o programa **psExec**, consulte *este* artigo na **Microsoft Learn**.

```
Sub MyTimer_OnPreset()
  Set objWScript = CreateObject("WScript.Network")
  strServer = objWScript.ComputerName
  Set sw = CreateObject("WSCript.shell")
  Set fso = CreateObject("Scripting.FileSystemObject")
  Set folder = fso.GetFolder("C:\MyAppDir\MyEntryIdDir")
  'Carrega os arquivos .eid
  For Each file In folder.Files
    GetAnExtension = fso.GetExtensionName(file.path)
    If GetAnExtension = "EID" then
      'Copia os arquivos
      fso.CopyFile file.path, "C:\MyAppDir\MyEntryIdDir\EID\"
    End If
  Next
  If strServer = "Server2" Then
    sw.run "C:\MyAppDir\Bat\psExec.exe -i 0 -u UserAccount -p Domain@password cmd /c C:
\MyAppDir\Bat\toServer1.bat", 0
  End If
  If strServer = "Server1" Then
    sw.run "C:\MyAppDir\Bat\psExec.exe -i 0 -u UserAccount -p Domain@password cmd /c C:
\MyAppDir\Bat\toServer2.bat", 0
  End If
End Sub
```

Conteúdo do arquivo **toServer1.bat**:

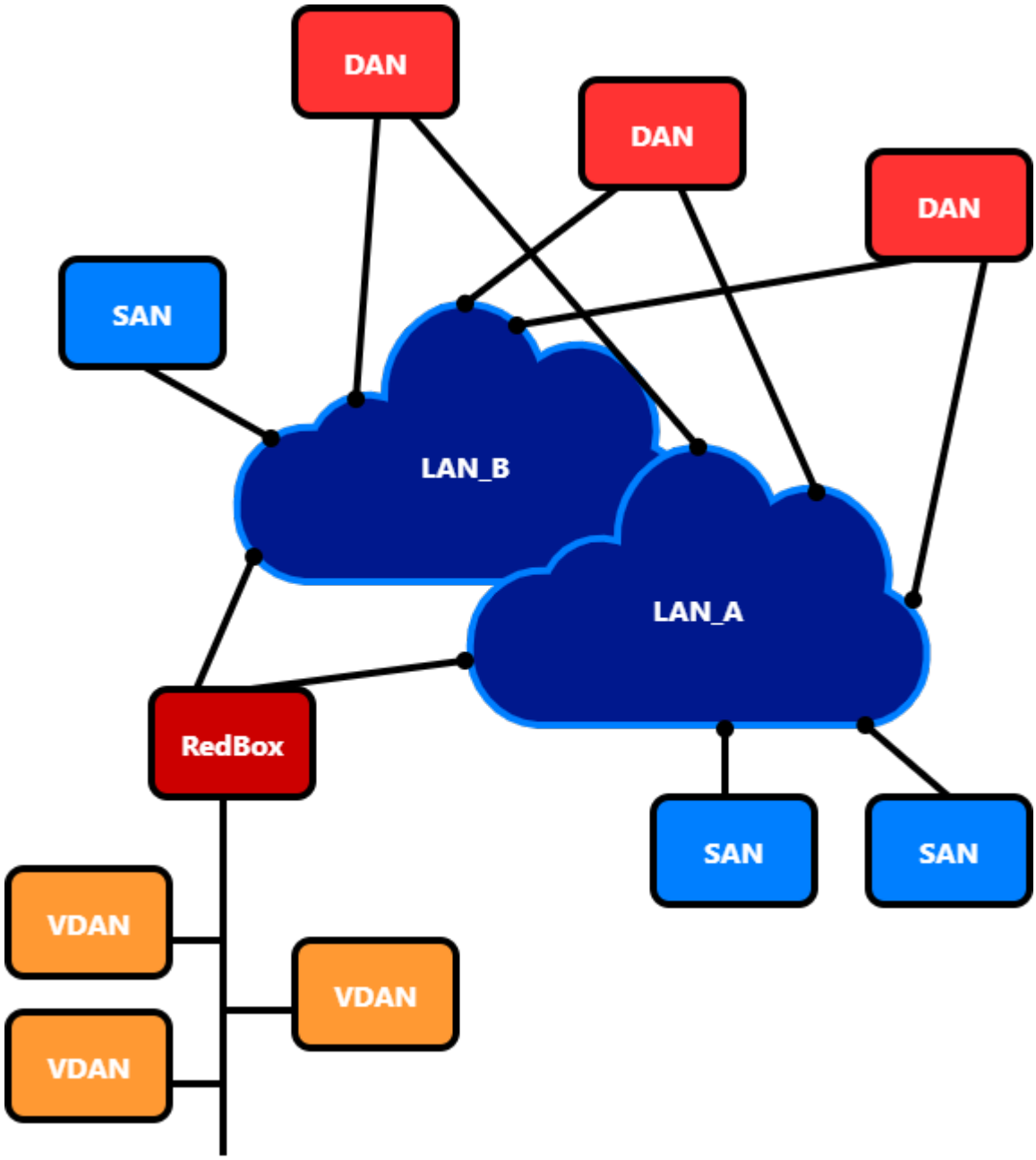
```
copy C:\MyAppDir\EntryId\EID, \\Server1\EntryID
```

Conteúdo do arquivo **toServer2.bat**:

```
copy C:\MyAppDir\EntryId\EID, \\Server2\EntryID
```

Redundância PRP

O **PRP** (*Parallel Redundancy Protocol*) é um padrão de protocolo para redes Ethernet que permite redundância ininterrupta contra qualquer falha simples de rede, em que a redundância é totalmente transparente para uma aplicação.



Configuração de uma rede PRP

No protocolo PRP devem existir duas redes paralelas, **LAN_A** e **LAN_B**, em que os dispositivos preferenciais e totalmente aderentes à redundância são chamados de **DANs** (*Dual Attached Nodes*), isto é, possuem duas interfaces, uma em cada rede. Dispositivos com uma única interface, chamados de **SAN** (*Single Attached Nodes*), também podem fazer parte desta arquitetura, mas não se beneficiam da redundância, bem como outros dispositivos que não suportam diretamente o protocolo PRP, chamados de **VDANs** (*Virtual Dual Attached Nodes*), podem estar conectados através de um **RedBox** (*Redundancy Box*).

Este Driver se comporta como um **DAN**, ou seja, um computador precisa ter duas placas de rede reais e uma placa de rede virtual, instaladas e configuradas. Os tópicos a seguir contêm os passos para esta configuração.

Biblioteca de Captura de Tráfego de Rede

O funcionamento do serviço PRP oferecido com este Driver é baseado na captura de pacotes que saem do adaptador virtual e são transferidos para os adaptadores reais **A** e **B**. Da mesma forma, pacotes recebidos nos adaptadores reais **A** ou **B** são transferidos para este adaptador virtual.

Para realizar a captura e injeção dos pacotes nos adaptadores, instale a biblioteca de captura de tráfego de rede **Npcap**, que é instalada com o *Wireshark* ou obtida diretamente no projeto *Nmap*.

Para o correto funcionamento, instale a biblioteca **Npcap** no modo de compatibilidade com a API (*Application Programming Interface*) **WinPcap**, que pode ser ativada na janela de instalação desta biblioteca.

Adaptador Virtual

Um **Adaptador Virtual** é uma placa de rede não física instalada no Windows com o objetivo de prover uma interface única para este Driver, ou outros protocolos, independente de qual rede física está ativa, **A**, **B** ou ambas.

Instalação

Para instalar um adaptador de rede virtual no **Windows 10**, siga estes procedimentos:

1. No menu **Iniciar** (*Start*), abra o **Gerenciador de Dispositivos** (*Device Manager*).
2. Selecione o item **Adaptadores de rede** (*Network adapters*) na árvore de dispositivos.
3. Selecione a opção **Adicionar hardware herdado** (*Add legacy hardware*) no menu **Ação** (*Action*).
4. Na janela **Adicionar Hardware** (*Add Hardware*), clique em **Avançar** (*Next*) e selecione a opção **Instalar o hardware que eu selecionar manualmente em uma lista (Avançado)** (*Install the hardware that I manually select from a list (Advanced)*) e clique em **Avançar** (*Next*).
5. Selecione o item **Adaptadores de rede** (*Network adapters*) e clique em **Avançar** (*Next*).
6. Selecione o item **Microsoft** na lista **Fabricante** (*Manufacturer*), selecione o adaptador **KM-TEST Loopback** na lista **Modelo** (*Network Adapter*) e clique em **Avançar** (*Next*) nas próximas telas.
7. Também é possível utilizar o adaptador virtual oferecido na instalação da biblioteca **Npcap**, porém este adaptador é um atalho para o adaptador **KM-Test Loopback**. Pode-se selecionar esta opção durante a instalação da biblioteca *Npcap*.

Configuração

Uma vez definido um adaptador virtual, habilite neste adaptador o uso da biblioteca de captura **Npcap** e defina um endereço IP fixo, conforme os procedimentos a seguir.

1. No menu **Iniciar** (*Start*), abra o **Painel de Controle** (*Control Panel*) e selecione a opção **Rede e Internet - Conexões de Rede** (*Network and Internet - Network Connections*).
2. Clique com o botão direito do mouse no adaptador virtual criado anteriormente e selecione a opção **Propriedades** (*Properties*).
3. Na janela de propriedades, habilite o menor número de opções possíveis para diminuir ao máximo o tráfego até os IEDs 61850 na lista **Esta conexão utiliza os seguintes itens** (*This connection uses the following items*). As principais opções são **Npcap Packet Driver** e **Internet Protocol Version 4 (TCP/IPv4)**.
4. Clique duas vezes na opção **Internet Protocol Version 4 (TCP/IPv4)**, selecione a opção **Usar o seguinte endereço IP** (*Use the following IP address*) na aba **Geral** (*General*), informe um endereço IP fixo e válido para acessar a rede de IEDs na opção **Endereço IP** (*IP address*) e clique em **OK** para aplicar as alterações.

Definição de MTU

Uma interface virtual, quando instalada, define por padrão um valor de MTU (*Maximum Transmission Unit*) praticamente ilimitado. Porém as placas de rede físicas geralmente utilizam um valor de MTU de 1500 bytes. Na versão atual do serviço **PRP** da **Elipse Software**, o valor de MTU é ajustado dinamicamente para 1500 bytes, de forma que um *frame* gerado em um adaptador virtual possa ser transmitido para os adaptadores físicos. Para verificar ou ajustar o valor de MTU para algum adaptador de rede de forma permanente, que é o processo recomendável, execute os passos a seguir.

1. Abra um Prompt de Comando no modo **Administrador** (*Administrator*). Para isto, clique com o botão direito do mouse no ícone do Prompt de Comando (*Command Prompt*) no menu **Iniciar** (*Start*) e selecione a opção **Executar como administrador** (*Run as administrator*).

2. Para verificar o índice e o valor de MTU das interfaces existentes, digite o comando a seguir.

```
> netsh interface ipv4 show interfaces
```

3. Para alterar o valor de MTU de uma interface, digite o comando a seguir. O parâmetro <*Idx*> corresponde ao índice desta interface e deve ser substituído pelo valor da coluna **Ind** (*Idx*) mostrado no resultado do comando anterior.

```
> netsh interface ipv4 set subinterface <Idx> mtu=1500 store=persistent
```

Associação de uma Placa Virtual a este Driver

Para garantir que as mensagens MMS (*Manufacturing Message Specification*) do protocolo 61850 sejam direcionadas apenas para o adaptador virtual e não para outros adaptadores, é necessário informar a este Driver esta configuração. Pode-se executar esta configuração informando manualmente qual adaptador deve ser usado por este Driver ou informando, na opção **ServerName/Interface List** da aba **PRP**, uma lista com os nomes dos servidores que executam este Driver e os respectivos endereços IP.

Definindo Manualmente uma Interface de Rede

1. Abra a janela de propriedades deste Driver e, na aba **Setup**, defina a opção **Physical Layer** com o valor **Ethernet**.

2. Na aba **Ethernet**, selecione a opção **Interface** e, na lista de endereços, selecione o endereço IP definido para o adaptador virtual. Esta definição é obrigatória para que o serviço seja iniciado.

3. Retorne à aba **Setup**, defina a opção **Physical Layer** com o valor **None** e clique em **OK** para salvar as configurações deste Driver.

Definindo Dinamicamente uma Interface de Rede

1. Na aba **PRP**, informe na opção **ServerName/Interface List** uma lista de nomes de servidores (DNS) e endereços IP no formato **ServerA:IPA;ServerB:IPB** para que um mesmo arquivo de projeto contendo este Driver possa ser usado sem alterações em ao menos dois servidores. Quando este Driver inicia, obtém o nome do computador local e verifica se há nesta lista um endereço IP correspondente a este nome. Em caso positivo, este endereço IP é usado como o endereço IP da interface virtual.

NOTAS

- No caso de outros computadores com o serviço **PRP** da **Elipse Software** na mesma rede, é necessário definir um endereço MAC (*Media Access Control*) diferente para cada adaptador virtual em cada computador. Isto porque um adaptador virtual é instalado sempre com um endereço MAC padrão, idêntico entre todos os computadores onde está instalado, como por exemplo dois servidores **Elipse Power** redundantes. Para modificar um endereço MAC, abra a janela de propriedades deste adaptador de rede, clique em **Configurar** (*Configure*), selecione a aba **Avançado** (*Advanced*), clique em **Network Address** na lista **Propriedade** (*Property*) e defina um endereço MAC na opção **Valor** (*Value*). Este valor deve estar no padrão de 12 dígitos hexadecimais, como por exemplo "02004C4F4F52".
- Devido à uma característica dos adaptadores de rede virtuais, ao desabilitar um adaptador no Windows durante a execução do serviço **PRP** e habilitar este adaptador novamente, é preciso reiniciar este serviço para que a comunicação volte a funcionar.

Adaptadores Físicos

O funcionamento esperado do protocolo **PRP** utiliza dois adaptadores de rede físicos, um conectado à rede **A** e outro conectado à rede **B**. Entretanto, este serviço pode ser iniciado e operar somente com uma placa de rede, de forma temporária, desde que tenha sido configurado corretamente ao menos uma vez. A configuração destes adaptadores físicos deve respeitar as seguintes regras:

- Não devem possuir um endereço IP habilitado, bem como todos os demais protocolos de rede
- Devem possuir o mesmo endereço MAC (*Media Access Control*)

Para realizar esta configuração, execute os procedimentos a seguir.

1. No menu **Iniciar** (*Start*), abra o **Painel de Controle** (*Control Panel*) e selecione a opção **Rede e Internet - Conexões de Rede** (*Network and Internet - Network Connections*).
2. Clique com o botão direito do mouse no adaptador Ethernet que recebe a rede **A** e selecione a opção **Propriedades** (*Properties*).
3. Na janela de propriedades, desabilite todos os protocolos de rede, inclusive a opção **Internet Protocol Version 4 (TCP/IPv4)**, na lista **Esta conexão utiliza os seguintes itens** (*This connection uses the following items*), deixe habilitada somente a opção **Npcap Packet Driver** e clique em **OK** para salvar as alterações.
4. Abra novamente a janela de propriedades deste adaptador de rede e clique em **Configurar** (*Configure*).
5. Selecione a aba **Avançado** (*Advanced*), clique em **Network Address** na lista **Propriedade** (*Property*) e defina um endereço MAC na opção **Valor** (*Value*). Este valor deve estar no padrão de 12 dígitos hexadecimais, como por exemplo "02004C4F4F52".
6. Repita estes procedimentos para o adaptador que recebe a rede **B**.

NOTAS

- É importante que ambos os adaptadores de rede tenham o mesmo endereço MAC. Para isto, configure o mesmo endereço MAC do primeiro adaptador de rede no segundo adaptador de rede ou vice-versa, ou ainda defina um novo endereço MAC para ambos os adaptadores de rede.
- Para saber o endereço MAC atual de um adaptador de rede, digite o comando **getmac** em um Prompt de Comando (*Command Prompt*) do Windows.

Instalação, Configuração e Execução do Serviço PRP

Juntamente com este Driver, a partir da versão **3.0.42**, são fornecidos os seguintes programas:

- **ElipsePRPSvc.exe**: Serviço que implementa o protocolo **PRP**
- **ElipsePRPTray.exe**: Programa que permite configurar e monitorar o serviço **ElipsePRPSvc**

Estes programas devem ser copiados para algum diretório de livre escolha para a utilização em uma rede PRP. Estes programas podem estar localizados no mesmo diretório deste Driver ou em um diretório diferente. Copie também para este mesmo diretório o arquivo et32b.dll, localizado na pasta **Bin** ou **Bin32** da instalação do **Elipse Power**.

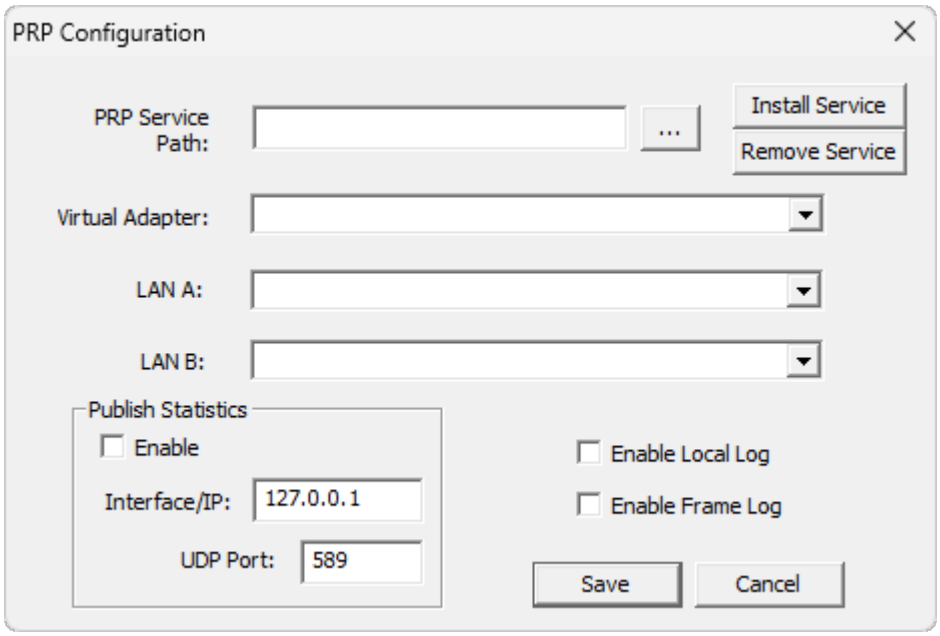
Para que este serviço funcione corretamente, deve-se instalá-lo no computador e também deve existir no mesmo diretório um arquivo de inicialização, chamado ElipsePRPSvc.ini. Ambas as configurações são realizadas através do programa **ElipsePRPTray**.

ElipsePRPTray

O programa **ElipsePRPTray** deve ser executado como administrador para poder instalar e configurar o serviço. Para isto, clique com o botão direito do mouse no arquivo executável e selecione a opção **Executar como administrador** (*Run as administrator*). Ao abrir, este programa exibe, na Área de Notificações do Windows, um ícone indicando que o serviço não está instalado, um ícone indicando que o serviço está instalado mas parado ou um ícone indicando que o serviço está em execução. As opções disponíveis no menu contextual do programa **ElipsePRPTray** estão descritas na tabela a seguir.

Opções do menu contextual do ElipsePRPTray

OPÇÃO	DESCRIÇÃO
About Elipse PRP Tray	Abre uma caixa de diálogo com informações sobre este programa
Configuration	Permite instalar e configurar o serviço na janela PRP Configuration, mostrada na figura a seguir
Run/Stop	Com o serviço instalado, esta opção permite parar ou executar o serviço
Exit	Fecha o programa sem alterar o estado do serviço



Janela PRP Configuration

As opções disponíveis nesta janela estão descritas na tabela a seguir.

Opções disponíveis na janela PRP Configuration

OPÇÃO	DESCRIÇÃO
PRP Service Path	Indica o caminho completo do arquivo ElipsePRPSvc.exe
Virtual Adapter	Define um adaptador virtual de rede
LAN A	Define o primeiro adaptador físico de rede
LAN B	Define o segundo adaptador físico de rede
Install Service	Instala o serviço **Elipse HSR/PRP Service**
Remove Service	Remove o serviço **Elipse HSR/PRP Service**

OPÇÃO	DESCRIÇÃO
Publish Statistics - Enable	Define se o serviço Elipse HSR/PRP Service publica estatísticas de pacotes para este Driver
Publish Statistics - Interface/IP	Define o endereço IP para a publicação de estatísticas de pacotes para este Driver
Publish Statistics - UDP Port	Define a porta UDP/IP para a publicação de estatísticas de pacotes para este Driver
Enable Local Log	Habilita a geração de logs básicos do serviço **Elipse HSR/PRP Service**
Enable Frame Log	Habilita a geração de logs de pacotes
Save	Gera um arquivo ElipsePRPSvc.ini no diretório de instalação do serviço **Elipse HSR/PRP Service**

O serviço **Elipse HSR/PRP Service** é instalado por padrão com a opção de inicialização automática. Para alterar esta configuração, abra o programa **Serviços** (*Services*) no menu **Iniciar** (*Start*), clique duas vezes neste serviço e selecione o item **Automático** (*Automatic*) da opção **Tipo de inicialização** (*Startup*) na aba **Geral** (*General*).

Ambiente de Produção

Para uma instalação em ambiente de produção, recomenda-se que este serviço seja configurado com a opção de inicialização automática e que seja automaticamente reiniciado em caso de falha. A partir da versão **1.0.0.2**, se alguma das interfaces de rede não possa ser acessada por mais de 1 (um) minuto, este serviço força uma saída com um código de erro 57 (*Adapter Hardware Error*). Para configurar este serviço para reiniciar automaticamente, selecione a aba **Discovery** e realize as seguintes alterações:

- Selecione o item **Restart the Service** na opção **First Failure**
- Selecione o item **Restart the Service** na opção **First Failure**
- Selecione o item **Restart the Service** na opção **Subsequent Failure**
- Selecione a opção **Enable actions for stops with errors**
- Defina o tempo de espera para reiniciar este serviço na opção **Restar service after**

Além disto, desabilite as opções **Enable Local Log** e **Enable Frame Log** na janela PRP Configuration para evitar a geração de excessiva de logs.

NOTAS

- O funcionamento deste serviço é totalmente independente deste Driver e de qualquer outra aplicação da **Elipse Software** estar em execução ou não.
- Para utilizar o programa **ElipsePRPTray** sempre que o Windows inicia, adicione um atalho no menu **Iniciar** (*Start*) do Windows. Para isto, clique com o botão direito do mouse no ícone do programa e selecione a opção **Adicionar ao Startup** (*Pin to Start*).
- No caso de trocar este serviço de diretório depois de executado uma vez, remova este serviço do Windows manualmente. Para isto, selecione este serviço no programa **Serviços** (*Services*) e selecione a opção **Parar** (*Stop*). Para remover, abra um Prompt de Comando (*Command Prompt*) do Windows como administrador e execute o comando **sc delete "ElipsePRP"**.

Monitoramento e Estatísticas

As informações de início, como por exemplo erros de configuração, são registradas pelo serviço em um arquivo ElipsePRPSvc.log gerado no mesmo diretório do serviço, caso esta opção esteja habilitada. Após iniciar, e se o serviço de monitoramento está habilitado, o serviço passa a enviar logs e estatísticas para a porta UDP/IP selecionada, que são

recebidos por este Driver. Estas mensagens de logs são redirecionadas para o log deste Driver com o prefixo "PRPService: XXXXXXXX". Estes Tags devem ser criados na aplicação e estão descritos na tabela a seguir.

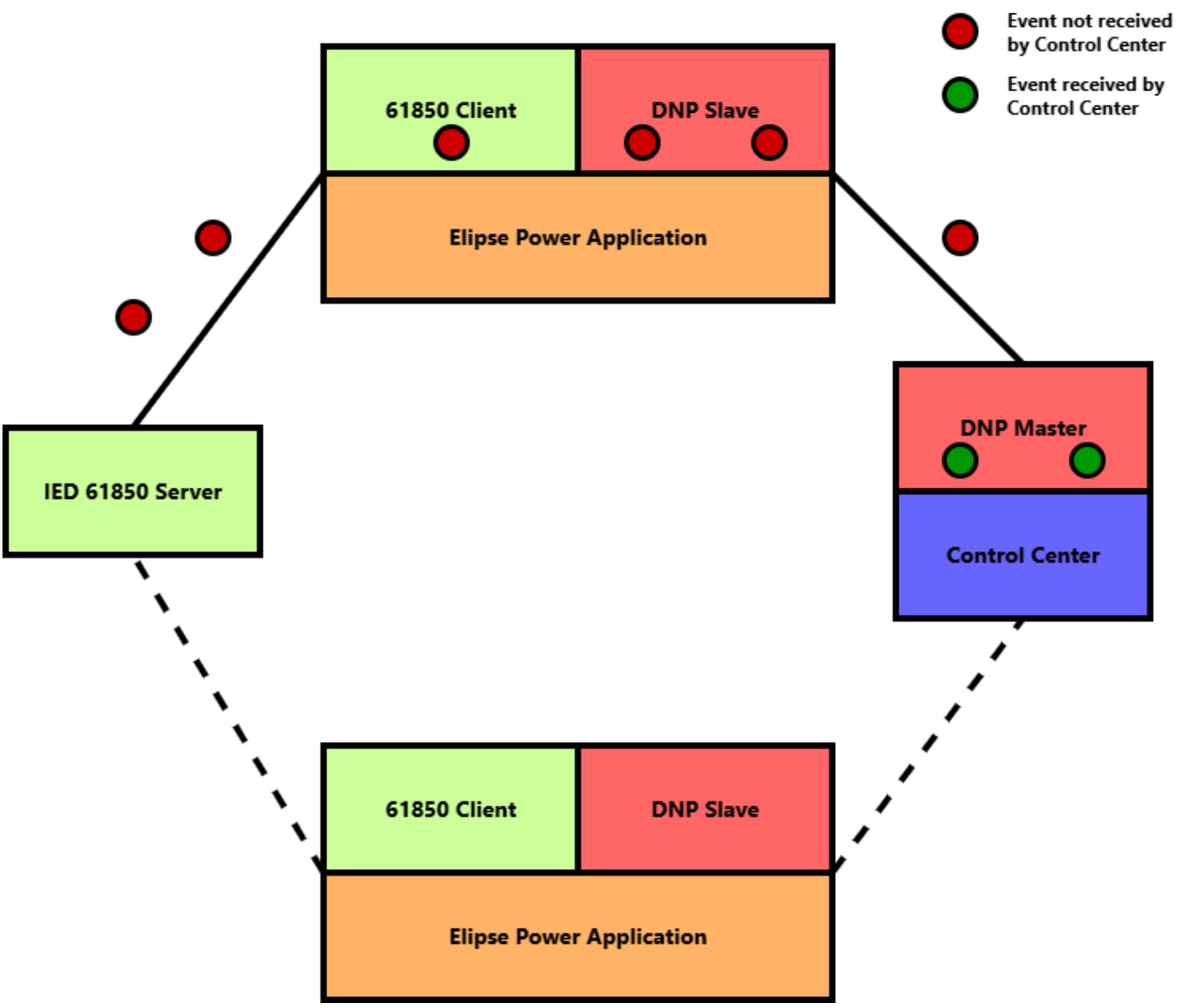
Tags disponíveis para estatísticas

NOME	DESCRIÇÃO	DEVICE	ITEM
LANA_Active	Indica se o serviço **PRP** está logicamente ativo no adaptador de rede **LAN A**. Adaptadores de rede invertidos, por exemplo, não permitem que nenhum destes adaptadores de rede permaneça ativo	ElipseClientInfo	PRPStat_LANA_Active
LANB_Active	Indica se o serviço **PRP** está logicamente ativo no adaptador de rede **LAN B**. Adaptadores de rede invertidos, por exemplo, não permitem que nenhum destes adaptadores de rede permaneça ativo	ElipseClientInfo	PRPStat_LANB_Active
LANA_TotalSent	Indica o número total de pacotes enviados pelo adaptador de rede **LAN A**	ElipseClientInfo	PRPStat_LANA_TotalSent
LANB_TotalSent	Indica o número total de pacotes enviados pelo adaptador de rede **LAN B**	ElipseClientInfo	PRPStat_LANB_TotalSent
LANA_TotalReceived	Indica o número total de pacotes recebidos pelo adaptador de rede **LAN A**	ElipseClientInfo	PRPStat_LANA_TotalReceived
LANB_TotalReceived	Indica o número total de pacotes recebidos pelo adaptador de rede **LAN B**	ElipseClientInfo	PRPStat_LANB_TotalReceived

Defina um intervalo de varredura confortável na propriedade **Scan** destes Tags, de forma que não poluam demais os logs deste Driver. Como sugestão, configure um valor entre 5000 e 10000 ms, ou seja, entre 5 (cinco) e 10 segundos.

Configuração de Gateway

Para aplicações de *gateway*, em que é necessário o envio dos dados deste Driver para outros protocolos, tais como DNP 3.0 ou IEC 104, podem ocorrer situações em que, em caso de reinício de uma aplicação ou chaveamento para uma aplicação redundante, ou em **Hot-Standby**, há uma perda ou duplicação de eventos. Isto pode ocorrer quando um evento foi recebido por este Driver e repassado, mas ainda não foi recebido pelo **Mestre** remoto do outro protocolo, conforme o exemplo da figura a seguir.



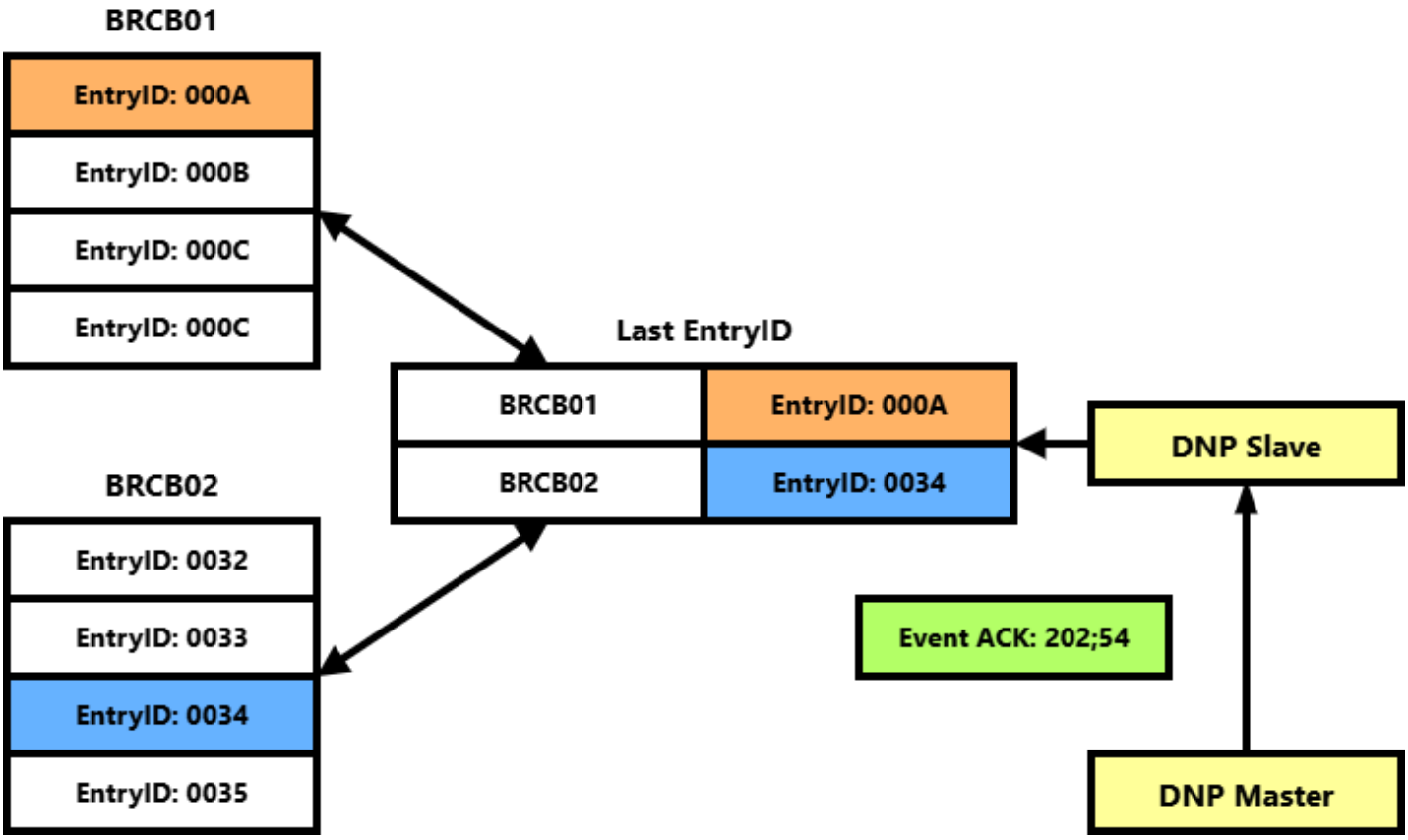
Configuração de Gateway

A imagem anterior mostra que podem existir eventos transferidos e pendentes de transmissão nos diversos componentes, tais como IEC 61850 Server, IEC 61850 Client, DNP Slave ou DNP Master. Esta fila de eventos pendentes pode ser maior ou menor dependendo das configurações destes componentes e da arquitetura de rede, dentre outros fatores.

Quando ocorre um chaveamento do **Gateway A** para o **Gateway B**, por exemplo, os eventos que estavam pendentes de transmissão podem tanto serem perdidos ou duplicados, dependendo da configuração de uma aplicação. Isto porque o protocolo 61850 permite que sejam recebidas novamente as mensagens de **Buffered Reports**, a partir da programação do campo **EntryID** de cada **Report**, que é um identificador único de cada mensagem enviada.

Este Driver já possui um recurso de geração de arquivos com o último **EntryID** de cada **Report**, que é utilizado para habilitação dos **Reports** quando este Driver inicia novamente. Entretanto, estes **EntryIDs** são atualizados com base no instante de recepção das mensagens por este Driver, e não pela recepção de fato de um dado em um elemento externo. No caso do exemplo anterior, no Centro de Controle (*Control Center*).

Desta forma, ao habilitar o recurso de *Gateway* na opção **Wait for Gtw Ref on Startup**, este Driver espera que uma aplicação **Elipse Power** informe uma **String** que representa o endereço de um Tag em um outro Driver, como por exemplo DNP ou IEC 104 Escravo. Quando este outro Driver recebe a confirmação que um evento foi entregue ao Centro de Controle, DNP ou IEC 104 Mestre, que pode informar este Driver sobre este evento, e desta forma este Driver pode posicionar os controles de **EntryID** à frente para que, caso haja um novo chaveamento, o fluxo de eventos continue a partir deste último ponto, evitando assim ao máximo a perda ou repetição de eventos.



Gerenciamento de EntryIDs

Os passos para a configuração deste recurso são os seguintes:

1. Habilite as opções **Check BRCB Entry ID**, **R/W Entry ID File (.EID)** e **Wait for Gtw Ref on Startup** na aba **Reports**. Além disto, caso um IED utilize os mesmos **Reports** para os dois *gateways*, deve-se avaliar o uso de uma lista fixa de **Reports** na opção **User Defined Report List**, evitando assim que uma aplicação redundante tente habilitar outros **Reports** que não são os esperados.

2. Neste Driver, crie um Tag chamado "FinishedGtwRef" e dois Tags Bloco chamados "GatewayRef" e "GatewayEventOK. Estes Tags estão descritos no tópico **Referência de Tags - Tags Internos**.

3. Tomando o Driver DNP Escravo como exemplo, crie um Tag Bloco com o parâmetro *N2* igual a -22, que recebe no evento **OnRead** uma lista com todos os eventos confirmados por um DNP Mestre. Consulte a documentação do Driver DNP *Escravo* e *Mestre* para mais informações.

4. Os Tags no Driver DNP Escravo devem receber Associações dos Tags deste Driver. Por exemplo, um Tag para um objeto DNP 3205 com índice 0 (zero) é alimentado por uma corrente de fase A vinda este Driver.

5. Uma aplicação, ao iniciar, deve inicializar as referências passando um texto que representa o endereço DNP para cada Tag deste Driver necessário para realizar o controle de **EntryIDs**. Porém esta escrita só pode ser realizada no momento em que este Driver está com os **Logical Devices** criados, caso contrário esta escrita falha. Para solucionar este problema, o exemplo a seguir mostra um Tag Interno que tenta enviar a configuração a cada 5 (cinco) segundos. A partir do momento que esta escrita é bem-sucedida, não há mais nenhuma tentativa. Ao final deste processo, o Tag **FinishedGtwRef** é escrito.

6. Também neste exemplo, a **String** de referência foi obtida através da união do objeto e variação do Tag DNP mais o índice. No Driver DNP Escravo, estes valores são obtidos nos parâmetros *N3* e *N4*, respectivamente.

   - **Valor inicial do Tag InitGateway**: True (Boolean)
   - **Evento WhileNotOK**: Expressão: Value Repetir o Evento a cada 5000 ms

```
Sub InitGateway_WhileNotOK()
  Set DNPSlave = Application.GetObject("DNPSlave")
  Set Ref = Application.GetObject("IEC61850").Item("GatewayRef")
  bFirst = FALSE
  For Each tag in DNPSlave
    If TypeName(tag) = "IOTag" Then
      If tag.N2 = 1 Then
        Source = tag.Links.Item("Value").Source
        Set s = Application.GetObject(Source)
        arr = Split(s.ParamDevice,":")
        Ref.Item("IEDName").Value = arr(0)
        Ref.Item("LDName").Value = arr(1)
        Ref.Item("Object").Value = s.ParamItem
        Ref.Item("Ref").Value = CStr(tag.N3) + ";" + CStr(tag.N4)
        If NOT bFirst Then
          bFirst = Ref.Write(1) 'sync
          If NOT bFirst Then Exit Sub
        Else
          Ref.Write 2 'async unconfirmed
        End If
      End If
    End If
  Next
  res = Application.GetObject("IEC61850").Item("FinishedGtwRef").WriteEx(2) 'async unconfirmed
  If res Then Value = FALSE
End Sub
```

7. Quando o Tag Bloco **EventOK** é lido no Driver DNP Escravo, isto significa que um novo evento foi recebido pelo Driver DNP Mestre. O script a seguir obtém as informações deste evento e repassa a este Driver. No caso específico do Driver DNP Escravo, este Tag Bloco retorna algumas informações sobre este evento, como objeto e variação, índice, classe DNP, valor, estampa de tempo válida e estampa de tempo. Todos estes parâmetros são enviados para este Driver via escrita no Tag Bloco **GatewayEventOK**, exceto a classe DNP, que não é relevante.

```
Sub EventOK_OnRead()
  Set IECBlock = Application.GetObject("IEC61850.GatewayEventOK")
  IECBlock.Item("Ref").Value = CStr(Item("ObjVar").Value) + ";" + CStr(Item("Index").Value)
  IECBlock.Item("Value").Value = Item("Value").Value
  IECBlock.Item("TSValid").Value = Item("TSValid").Value
  IECBlock.Item("Time").Value = Item("Time").Value
  IECBlock.Write 2 'async
End Sub
```

8. Neste momento, este Driver posiciona os **EntryIDs** para o próximo evento pendente. Para sincronizar os **EntryIDs** com o computador redundante, é necessário seguir os passos indicados no tópico **Redundância - Utilizando Arquivos EID**.

9. Para posicionar os **EntryIDs**, este Driver utiliza a seguinte metodologia:

   - Quando um novo **Report** é recebido, verifica se algum objeto teve uma variação de valor ou qualidade. Caso positivo, um evento é adicionado à uma lista interna individual por objeto.

   - Quando um evento é quitado por um Driver externo e recebido pelo Tag Bloco **GatewayEventOK**, a lista interna é comparada com o evento removido da seguinte forma:

     - Se a estampa de tempo não é válida, ou seja, um Tag nos Drivers DNP ou IEC 104 não está usando estampa de tempo, todos os eventos são removidos desta lista

     - Se a estampa de tempo é válida, os eventos são removidos se são iguais ou mais antigos que o evento recebido

Para evitar processamentos desnecessários, recomenda-se que apenas sejam adicionadas referências de Tags que não podem ter mudanças perdidas ou duplicadas, tipicamente Tags digitais ou de estados simples ou duplos com estampa de tempo. Também recomenda-se que, neste caso, os **Datasets** associados aos **Reports** sejam separados por tipo de Tag, como por exemplo não colocar no mesmo **Dataset** Tags Digitais e Analógicos, de forma a evitar a recepção de valores analógicos antigos sem necessidade.

Configuração Dinâmica

Este Driver permite que alguns parâmetros de configuração sejam informados em tempo de execução por script, usando o comando de escrita de parâmetros da biblioteca **IOKit**. Estes parâmetros estão descritos na tabela a seguir.

Parâmetros de configuração

PROPRIEDADE	PARÂMETRO	TIPO DE DADOS
Total Device Number	IEC61850.DeviceCount	DWORD
Server	IEC61850.Device[%u].Name	STRING
IP	IEC61850.Device[%u].IP	STRING
Backup IP	IEC61850.Device[%u].IPBackup	STRING
PSEL	IEC61850.Device[%u].PSel	DWORD
SSEL	IEC61850.Device[%u].SSel	DWORD
TSEL	IEC61850.Device[%u].TSel	DWORD
Rem AE Qual	IEC61850.Device[%u].AEQualifier	DWORD
Rem AP ID	IEC61850.Device[%u].AppID	STRING
Disable	IEC61850.Device[%u].Disable	BYTE
Use Backup IP	IEC61850.Device[%u].UseIPBackup	BYTE
Save Comtrade (Per Device)	IEC61850.Device[%u].SaveComtrade	BYTE
Delete Comtrade Files (Per Device)	IEC61850.Device[%u].DeleteComtrade	BYTE
Browse Root Folder Only (Per Device)	IEC61850.Device[%u].BrowseRootFolderOnly	BYTE
Comtrade Path (Per Device)	IEC61850.Device[%u].ComtradePath	STRING
Use Fixed Comtrade Path On Device (Per Device)	IEC61850.Device[%u].UseFixedPathOnDevice	BYTE
Name+Index File Transfer	IEC61850.Device[%u].NameAndIndexFileTransfer	BYTE
File Root Name	IEC61850.Device[%u].FileTransferRootName	STRING
Prefer Buffered Report Control Blocks	IEC61850.UseReports	BYTE
User-Defined Report List	IEC61850.UserReportList	BYTE
Poll Tags not found in any report	IEC61850.PollTags	BYTE
Local P Selector	IEC61850.LocalPSel	DWORD
Local S Selector	IEC61850.LocalSSel	DWORD
Local T Selector	IEC61850.LocalTSel	DWORD
Local AE Qualifier	IEC61850.LocalAEQualifier	DWORD
Local App ID	IEC61850.LocalAppID	STRING
RFC 1006 Source TSAP	IEC61850.SourceTSAP	DWORD
App Category	IEC61850.AppCategory	STRING
Polled Intg Rpt	IEC61850.RptGI	DWORD

PROPRIEDADE	PARÂMETRO	TIPO DE DADOS
Auto Intg Prt	IEC61850.IntgPd	DWORD
Use Quality change trigger	IEC61850.UseQChgTrgOps	BYTE
Conformance Blocks (internal use)	IEC61850.CBB	STRING
Services (internal use)	IEC61850.Services	STRING
LD File Path	IEC61850.LDPath	STRING
Comtrade Path	IEC61850.ComtradePath	STRING
Comtrade Directory on Device (internal use)	IEC61850.ComtradeDir	STRING
Msg Timeout	IEC61850.Timeout	DWORD
Full Log Details	IEC61850.DetailedLog	BYTE
Save Comtrade Files	IEC61850.SaveComtrade	BYTE
Save Comtrade Per IED	IEC61850.SaveComtradePerIED	BYTE
Delete Files after upload	IEC61850.DeleteComtrade	BYTE
Comtrade Directory Check Interval	IEC61850.CheckComtrade	BYTE
Comtrade Upload Interval	IEC61850.UploadComtrade	DWORD
Check Report Time of Entry	IEC61850.RedundantBRCB	BYTE
Status Check	IEC61850.StatusCheck	DWORD
Asynchronous Write	IEC61850.AsyncWrite	BYTE
Use Single Tag Command Alias	IEC61850.SingleTagCmdAlias	BYTE
Apply Local Time Offset to Timestamps	IEC61850.ApplyLocalTime	BYTE
Use Cmd Tag N1 as check condition	IEC61850.SingleTagCmdN1	BYTE
No LD Database Scan	IEC61850.NoLDScan	BYTE
Browse Root Folder Only	IEC61850.ComtradeBrowseRootFolder Only	BYTE
Use Exclusive URCB	IEC61850.ReserveURCB	BYTE

Um exemplo de configuração COMTRADE dinâmica está descrito a seguir:

1. Este Driver deve ter todos os IEDs já declarados, e devem estar com a opção **Disable** configurada com o valor 1 (um). Isto evita que a comunicação inicie com os parâmetros incorretos.

2. Este Driver deve também estar com a opção **Define Parameters Per IED** configurada.

3. No evento **AfterStart** deve haver um script que altera os parâmetros de cada IED. Estes parâmetros devem ser escritos via biblioteca **IOKit**. Os parâmetros estão indicados no script a seguir, em que **%u** deve ser trocado pelo índice do IED neste Driver, começando com 0 (zero). Estes parâmetros estão disponíveis no log deste Driver.

```
"IEC61850.Device[%u].SaveComtrade"
"IEC61850.Device[%u].DeleteComtrade"
"IEC61850.Device[%u].BrowseRootFolderOnly"
"IEC61850.Device[%u].ComtradePath"
"IEC61850.Device[%u].UseFixedPathOnDevice"
"IEC61850.Device[%u].NameAndIndexFileTransfer"
"IEC61850.Device[%u].FileTransferRootName"
```

4. Considere que os parâmetros **ComtradePath** e **FileTransferRootName** são do tipo **String** e que, se o valor individual por IED não está configurado, considera-se o valor padrão definido na janela de propriedades. O valor padrão para as demais propriedades é 0 (zero, Falso).

5. Após escrever os parâmetros desejados, habilite cada IED escrevendo o valor 0 (zero) no parâmetro "IEC61850.Device[%u].Disable".

6. Inicie a comunicação com este Driver escrevendo o valor 1 (um) no Tag especial **ServerInitialBuild**. Este Tag cria os *Hosts* que estão habilitados e que ainda não foram criados.

```
' Exemplo considerando um array de 8 (oito) Elementos
' Os índices vazios são ignorados por este Driver
Dim Arr(8)
Arr(1) = Array("IEC61850.Device[0].SaveComtrade",1) ' Este IED busca Comtrades
Arr(2) = Array("IEC61850.Device[1].SaveComtrade",0) ' Este IED não busca Comtrades
Arr(3) = Array("IEC61850.Device[2].SaveComtrade",1) ' Este IED busca Comtrades
Arr(4) = Array("IEC61850.Device[2].NameAndIndexFileTransfer",1) ' Este IED busca Comtrades usando
RootName + Index
Arr(5) = Array("IEC61850.Device[0].Disable",0) ' Habilita IED
Arr(6) = Array("IEC61850.Device[1].Disable",0) ' Habilita IED
Arr(7) = Array("IEC61850.Device[2].Disable",0) ' Habilita IED
Driver.Write -1,0,0,3 Arr ' Escreve parâmetros
Driver.Item("ServerInitialBuild").WriteEx 1 'Supondo a existencia deste Tag com o parâmetro Item =
ServerInitialBuild,
'ao escrever 1 (um) inicia a comunicação com todos os IEDs habilitados
```

7. Ao parar este Driver, estes parâmetros não são salvos, por isso o evento **AfterStart** deve continuar existindo para que, ao iniciar este Driver, os IEDs sejam configurados corretamente.

Limitações deste Driver

O Driver IEC 61850 (25 Conexões) é fornecido com os seguintes tipos de licenças pela **Elipse Software**:

- **IEC61850.dll**: Permite a comunicação com até 25 relés (IEDs), de forma a garantir o desempenho da comunicação. Para mais relés, utilize uma nova instância deste Driver

- **Power_IEC61850.dll**: Permite a comunicação com apenas um relé (IED), utilizando uma licença padrão que permite a substituição por outro Driver de sistema elétrico. Para a lista completa de Drivers que compartilham esta licença, consulte a **Elipse Software**

Documentação das Interfaces de Comunicação

Esta seção contém a documentação das Interfaces de Comunicação referentes ao Driver **IEC61850**.

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.

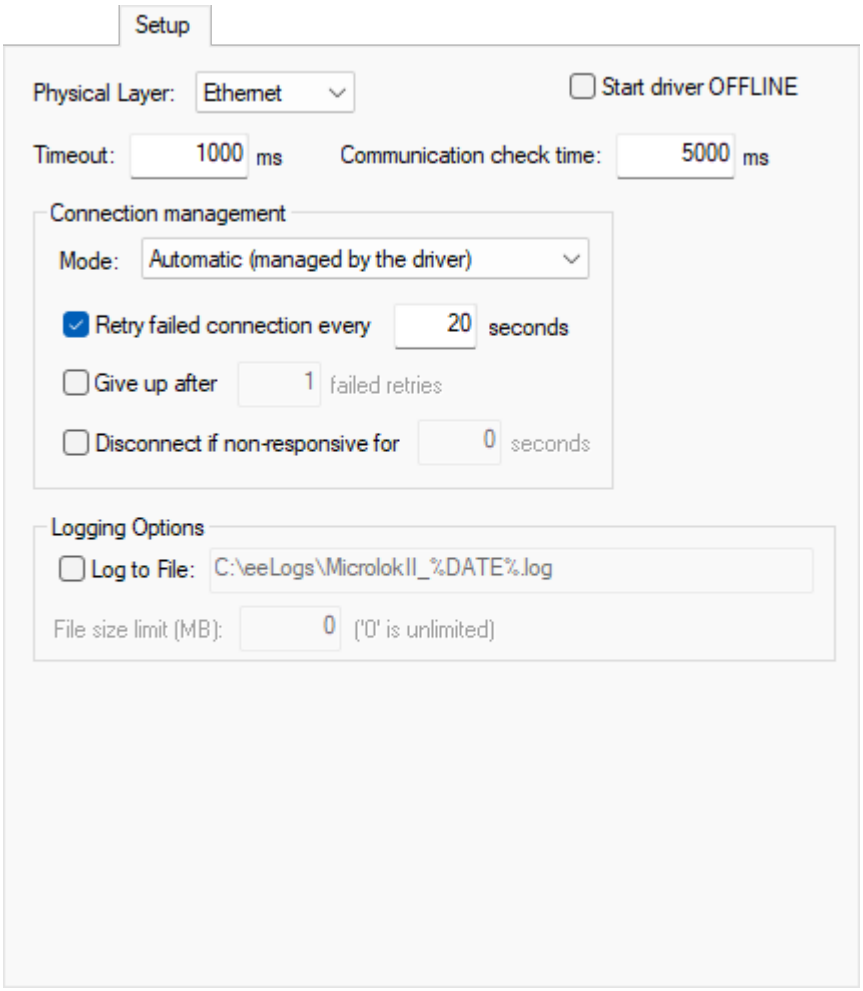
Caixa de Diálogo de Configuração

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



Aba Setup

Opções gerais da aba Setup

OPÇÃO	DESCRIÇÃO
Physical Layer	Selecione a interface física em uma lista. As opções disponíveis são **Serial**, **Ethernet**, **Modem** e **RAS**. A interface selecionada deve ser configurada na aba específica
Timeout	Configure o *time-out*, 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 *buffer* de recepção
Communication check time	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 **IO.CommunicationStatus**
Start driver OFFLINE	Selecione esta opção para que um Driver inicie em modo **Offline** ou parado. Isto significa que a interface de I/O não é criada até que se configure um Driver em modo **Online** 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

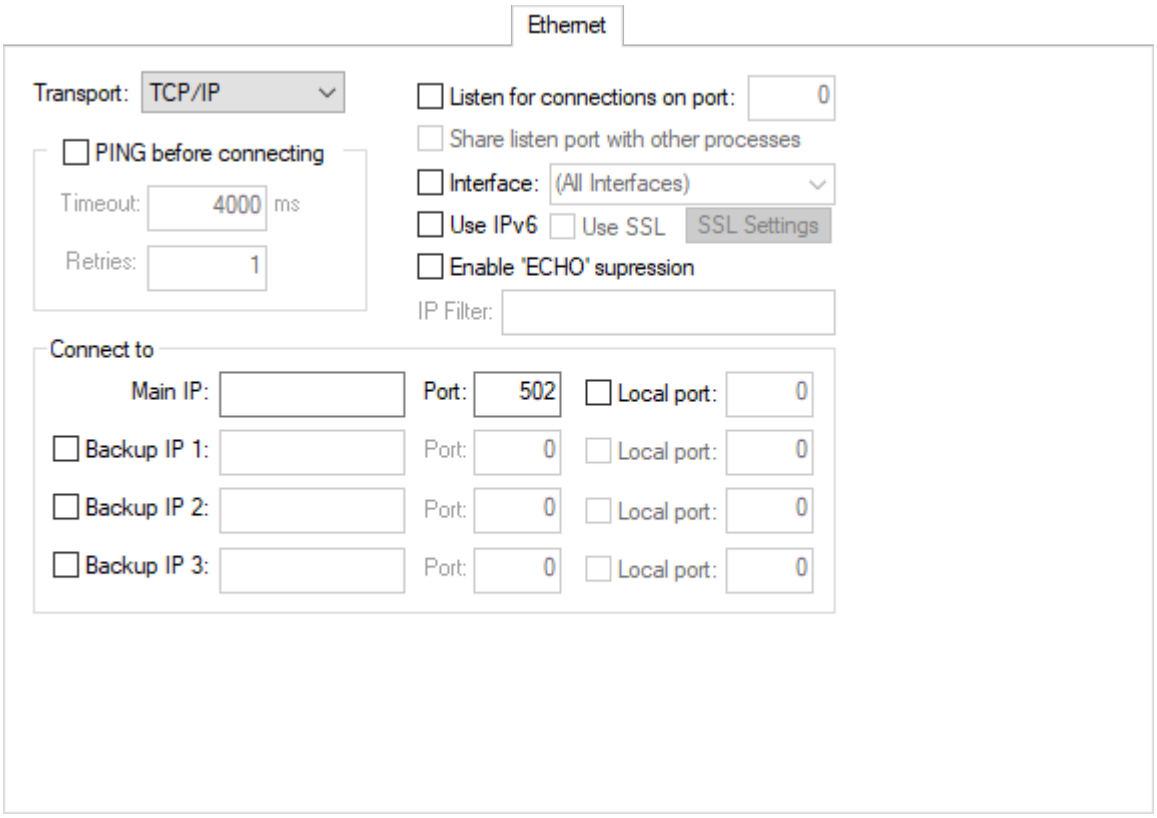
OPÇÃO	DESCRIÇÃO
Mode	Seleciona o modo de gerenciamento de conexão. Selecionar a opção **Automatic** permite que um Driver gerencie a conexão automaticamente, como especificado nas opções seguintes. Selecionar a opção **Manual** permite que uma aplicação gerencie a conexão completamente
Retry failed connection every ... seconds	Selecione esta opção para habilitar a retentativa de conexão de um Driver em um determinado intervalo, em segundos. Se a opção **Give up after failed retries** não está selecionada, este Driver continua retentando até que a conexão seja efetuada, ou que a aplicação seja parada
Give up after ... failed retries	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 **Offline**, 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
Disconnect if non-responsive for ... seconds	Habilite esta opção para forçar um Driver a se desconectar se nenhum byte chegou à interface de I/O no *time-out* especificado, em segundos. Este *time-out* deve ser maior que o *time-out* configurado na opção **Timeout**

Opções para o grupo Logging Options

OPÇÃO	DESCRIÇÃO
Log to File	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 **%PROCESS%** 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 **Elipse E3**, 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 **0FDAh**. Pode-se também utilizar a macro **%DATE%** no nome do arquivo. Neste caso é gerado um arquivo de log por dia, no formato **aaaa_mm_dd**. 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 **%DATE_HOUR%** gera um arquivo de log por hora, no formato **aaaa_mm_dd_hh**
File size limit (MB)	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

Aba Ethernet

Utilize esta aba para configurar os parâmetros da Interface **Ethernet**. Estes parâmetros, exceto as configurações de porta, devem ser também configurados para uso na Interface **RAS**.



Aba Ethernet

Opções disponíveis na aba Ethernet

OPÇÃO	DESCRIÇÃO
Transport	Selecione o valor **TCP/IP** para um *socket* TCP (*stream*) ou selecione o valor **UDP/IP** para utilizar um *socket* UDP (*connectionless datagram*)
Listen for connections on port	Utilize esta opção para aguardar por novas conexões em uma porta IP específica, comum em Drivers Escravos. Caso esta opção permaneça desmarcada, um Driver se conecta ao endereço e porta especificados no grupo **Connect to**
Share listen port with other processes	Selecione esta opção para compartilhar a porta de escuta com outros Drivers e processos
Interface	Selecione a interface de rede local, identificada pelo endereço IP, que um Driver utiliza para efetuar e receber conexões, ou selecione o valor **(All Interfaces)** para permitir conexões em qualquer interface de rede
Use IPv6	Selecione esta opção para forçar um Driver a utilizar endereços no formato **IPv6** em todas as conexões Ethernet. Deixe esta opção desmarcada para utilizar o formato **IPv4**
Enable 'ECHO' suppression	Habilite esta opção para eliminar o *eco* dos dados recebidos. O *eco* é uma cópia dos dados enviados, que pode ser retornada antes da mensagem de resposta
IP Filter	Lista de endereços IP restringidos ou permitidos de onde um Driver aceita conexões (*Firewall*). Consulte a propriedade **IO.Ethernet.IPFilter** para mais informações
PING before connecting	Habilite esta opção para executar um comando **ping**, ou seja, para verificar se um dispositivo pode ser encontrado na rede, em um dispositivo antes de tentar uma conexão

OPÇÃO	DESCRIÇÃO
	com o *socket*. Esta é uma maneira rápida de determinar uma conexão bem sucedida antes de tentar abrir um *socket* com um dispositivo. O *time-out* de uma conexão com um *socket* pode ser bem alto. As opções disponíveis são: • **Timeout**: Especifique o número de milissegundos de espera por uma resposta de um comando **ping**. Deve-se usar um comando **ping** para verificar o tempo normal de resposta, configurando esta opção para um valor acima desta média. Normalmente pode-se configurar um valor entre 1000 e 4000 milissegundos, ou seja entre 1 (um) e 4 (quatro) segundos • **Retries**: Número de retentativas de um comando **ping**, sem contar a tentativa inicial. Se todas as tentativas falharem, então a conexão com o *socket* é abortada

Opções disponíveis no grupo Connect to

OPÇÃO	DESCRIÇÃO
Main IP	Digite o endereço IP de um dispositivo remoto. Pode-se usar tanto o endereço IP separado por pontos quanto uma URL. No caso de uma URL, um Driver usa o serviço de DNS disponível para mapear a URL para um endereço IP, como por exemplo "192.168.0.13" ou "Server1"
Port	Digite a porta IP de um dispositivo remoto, entre 0 (zero) e 65535
Local port	Selecione esta opção para utilizar uma porta IP local fixa ao conectar a um dispositivo remoto
Backup IP 1, 2 e 3	Indique o endereço IP, a porta IP e a porta IP local fixa de até 3 (três) endereços de *backup* de um dispositivo remoto

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

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	6 (seis)
Configuração por String	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

Tipo de Tag	Tag Bloco
Tipo de Acesso	Somente Leitura
Parâmetro B1	-1 (menos um)
Parâmetro B2	0 (zero)
Parâmetro B3	0 (zero)
Parâmetro B4	1 (um)
Propriedade Size	4 (quatro)
Propriedade ParamItem	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

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	2 (dois)
Configuração por String	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

Tipo de Tag	Tag Bloco
Tipo de Acesso	Somente Leitura
Parâmetro B1	-1 (menos um)
Parâmetro B2	0 (zero)
Parâmetro B3	0 (zero)
Parâmetro B4	3 (três)
Propriedade Size	2 (dois)
Propriedade ParamItem	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

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Leitura ou Escrita
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	4 (quatro)
Configuração por String	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

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

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

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

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

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

IO.Type

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

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

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

IO.Stats.Partial.BytesSent

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

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

IO.Stats.Partial.TimeConnectedSeconds

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1102
Configuração por String	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

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1103
Configuração por String	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

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

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

IO.Stats.Total.BytesSent

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

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

IO.Stats.Total.ConnectionCount

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1004
Configuração por String	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

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

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

IO.Stats.Total.TimeDisconnectedSeconds

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1003
Configuração por String	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.

Configuração da Interface Ethernet

Esta seção contém informações sobre a configuração dos **Tags de Comunicação** e das **Propriedades** da Interface **Ethernet**.

Tags de Comunicação

Tags da Interface Ethernet (N2/B2 = 4)

Os Tags descritos a seguir permitem controlar e identificar a Interface **Ethernet** em tempo de execução e também são válidos quando a Interface **RAS** está selecionada.

IMPORTANTE
Estes Tags estão disponíveis **SOMENTE** enquanto um Driver está em modo **Online**.

IO.Ethernet.IPSelect

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

Indica o endereço IP ativo. Os valores possíveis são **0**: O endereço principal de IP está selecionado, **1**: O primeiro endereço IP alternativo ou de *backup* está selecionado, **2**: O segundo endereço IP alternativo ou de *backup* está selecionado ou **3**: O terceiro endereço IP alternativo ou de *backup* está selecionado.

Se a Interface **Ethernet** ou **RAS** está conectada, este Tag indica qual dos quatro endereços IP configurados está em uso. Se a Interface está desconectada, este Tag indica qual endereço IP é usado primeiro na próxima tentativa de conexão.

Durante o processo de conexão, se o endereço IP ativo não está disponível, a Interface de Comunicação tenta conectar-se usando o outro endereço IP. Se a conexão com o endereço IP alternativo funcionar, este é configurado como o endereço IP ativo (*switchover* automático).

Para forçar um *switchover* manual, escreva valores de 0 (zero) a três (3) neste Tag. Isto força a reconexão com o endereço IP especificado (**0**: Endereço principal, **1, 2, 3**: Endereços alternativos) se um Driver está atualmente conectado. Se um Driver está desconectado, este Tag configura o endereço IP ativo para a próxima tentativa de conexão.

IO.Ethernet.IPSwitch

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Escrita
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	4 (quatro)
Parâmetro N4	1 (um)
Configuração por String	IO.Ethernet.IPSwitch

Qualquer valor escrito neste Tag força um *switchover* manual. Se o endereço principal de IP está ativo, então o primeiro endereço IP alternativo ou de *backup* é ativado, e assim por diante passando por todos os endereços IP alternativos e voltando para o endereço principal até estabelecer uma conexão.

Se um Driver está desconectado, este Tag configura o endereço IP ativo para a próxima tentativa de conexão.

IO.Ethernet.SocketState

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	4 (quatro)
Parâmetro N4	2 (dois)
Configuração por String	IO.Ethernet.SocketState

A propriedade **Value** deste Tag corresponde a estados do *socket* em um mapa de bits:

- **Bit 0**: 0 (zero, não está em escuta) ou 1 (um, em escuta)

- **Bit 1**: 0 (zero, desconectado) ou 1 (um, conectado)

Propriedades

Estas propriedades controlam a configuração da Interface **Ethernet**.

NOTA

A Interface **Ethernet** também é usada pela Interface **RAS**.

IO.Ethernet.AcceptConnection

◩ Configure em Falso se um Driver não deve aceitar conexões externas, ou seja, se um Driver se comporta como mestre, ou configure em Verdadeiro para habilitar a recepção de conexões, ou seja, se um Driver se comporta como escravo.

IO.Ethernet.BackupEnable[2,3]

◩ Configure em Verdadeiro para habilitar o endereço IP alternativo ou de *backup*. Se a tentativa de reconectar com o endereço IP principal falhar, um Driver tenta utilizar um endereço IP alternativo ou de *backup*. Configure em Falso para desabilitar a utilização.

IO.Ethernet.BackupIP[2,3]

A Endereço IP alternativo ou de *backup* de um equipamento remoto. Pode-se utilizar tanto o endereço numérico como o nome de *host* de um equipamento, como por exemplo "192.168.0.7" ou "SERVER2".

IO.Ethernet.BackupLocalPort[2,3]

9 Número da porta local a ser utilizada na conexão ao endereço IP alternativo ou de *backup* de um equipamento remoto. Usado apenas se a propriedade **IO.Ethernet.BackupLocalPortEnable** está configurada para Verdadeiro.

IO.Ethernet.BackupLocalPortEnable[2,3]

◩ Configure em Verdadeiro para forçar o uso de uma porta local específica ao conectar ao endereço IP alternativo ou de *backup* ou configure em Falso para utilizar qualquer porta local disponível.

IO.Ethernet.BackupPort[2,3]

9 Número da porta do endereço IP alternativo ou de *backup* de um equipamento remoto, usado juntamente com a propriedade **IO.Ethernet.BackupIP**.

IO.Ethernet.IPFilter

A Lista de endereços IPv4 ou IPv6 separados por vírgula, que define de quais endereços um Driver aceita ou bloqueia conexões. Pode-se utilizar asteriscos, como por exemplo "192.168.*.*", ou intervalos, como por exemplo "192.168.0.41-50", em qualquer parte dos endereços IP. Para bloquear um endereço IP ou um intervalos de endereços IP, use o caractere til ("~") no início do endereço, conforme os exemplos a seguir:

- **192.168.0.24**: Aceita apenas conexões do endereço IPv4 192.168.0.24
- **192.168.0.41-50**: Aceita conexões dos endereços IPv4 no intervalo entre 192.168.0.41 e 192.168.0.50
- **192.168.0.***: Aceita conexões dos endereços IPv4 no intervalo entre 192.168.0.0 e 192.168.0.255

- **fe80:3bf:877::*:* (expande para fe80:03bf:0877:0000:0000:0000:*:*)**: Aceita conexões de endereços IPv6 no intervalo entre fe80:03bf:0877:0000:0000:0000:0000:0000 e fe80:03bf:0877:0000:0000:0000:ffff:ffff
- **192.168.0.10, 192.168.0.15, 192.168.0.20**: Aceita conexões dos endereços IPv4 192.168.0.10, 192.168.0.15 e 192.168.0.20
- **~192.168.0.95, 192.168.0.***: Aceita conexões dos endereços IPv4 no intervalo entre 192.168.0.0 e 192.168.0.255, exceto o endereço IPv4 192.168.0.95

Quando um Driver recebe uma tentativa de conexão, a lista de filtros é percorrida sequencialmente da esquerda para a direita, procurando por uma autorização ou bloqueio específico para o endereço IP de onde veio a conexão. Se nenhum elemento da lista corresponde ao endereço IP, a autorização ou bloqueio são ditados pelo último elemento da lista:

- Se o último elemento da lista é uma autorização, como por exemplo "192.168.0.24", então todos os endereços IP não encontrados na lista são bloqueados
- Se o último elemento da lista é um bloqueio, como por exemplo "~192.168.0.24", então todos os endereços IP não encontrados na lista são autorizados

Se um endereço IP aparece em mais de um filtro da lista, o filtro mais à esquerda tem precedência. Por exemplo, no caso de "~192.168.0.95, 192.168.0.*", o endereço IP 192.168.0.95 se encaixa nas duas regras, mas a regra que vale é a mais à esquerda, "~192.168.0.95", e portanto o endereço IP é bloqueado.

Quando o **IOKit** bloqueia uma conexão, a mensagem "Blocked incoming socket connection from {IP}!" é logada.

No caso de conexões UDP em modo escuta em *broadcast*, em que um Driver pode receber pacotes de diferentes endereços IP, o bloqueio ou permissão é realizado a cada pacote recebido. Se um pacote é recebido de um endereço IP bloqueado, a mensagem "Blocked incoming packet from {IP} (discarding {N} bytes)!" é logada.

IO.Ethernet.ListenIP

A Endereço IP da interface local de rede por onde um Driver efetua e aceita conexões. Deixe esta propriedade vazia para efetuar e aceitar conexões por qualquer interface local de rede.

IO.Ethernet.ListenPort

9 Número da porta IP utilizada por um Driver para escutar conexões.

IO.Ethernet.MainIP

A Endereço IP de um equipamento remoto. Pode-se utilizar tanto o endereço numérico como o nome de *host* de um equipamento, como por exemplo "192.168.0.7" ou "SERVER2".

IO.Ethernet.MainLocalPort

9 Número da porta local a ser utilizada na conexão ao endereço IP principal de um equipamento remoto. Este valor é usado apenas se a propriedade **IO.Ethernet.MainLocalPortEnable** é igual a Verdadeiro.

IO.Ethernet.MainLocalPortEnable

◩ Configure em Verdadeiro para forçar o uso de uma porta local específica ao conectar ao endereço IP principal ou configure em Falso para utilizar qualquer porta local disponível.

IO.Ethernet.MainPort

9 Número da porta IP em um equipamento remoto, usado em conjunto com a propriedade **IO.Ethernet.MainIP**.

IO.Ethernet.PingEnable

◩ Configure em Verdadeiro para habilitar o envio de um comando **ping** para o endereço IP de um equipamento remoto, antes de tentar conectar-se ao *socket*. O *time-out* de conexão do *socket* não pode ser controlado, por isto o envio de um comando **ping** antes de conectar-se é uma maneira rápida de detectar se a conexão vai falhar. Configure em Falso para desabilitar o comando **ping**.

IO.Ethernet.PingTimeoutMs

9 Tempo de espera por uma resposta de um comando **ping**, em milissegundos.

IO.Ethernet.PingTries

9 Número máximo de tentativas de comandos **ping**. O valor mínimo é 1 (um), incluindo o primeiro comando **ping**.

IO.Ethernet.ShareListenPort

◩ Configure em Verdadeiro para compartilhar a porta de escuta com outros Drivers e processos ou Falso para abrir a porta de escuta em modo exclusivo. Para compartilhar uma porta de escuta com sucesso, todos os Drivers e processos envolvidos devem abrir esta porta em modo compartilhado. Quando uma porta de escuta é compartilhada, cada nova conexão é distribuída para um dos processos que estão escutando. Desta forma, se um Driver Escravo só suporta uma conexão por vez, pode-se utilizar várias instâncias deste Driver escutando na mesma porta, portanto simulando um Driver com suporte a múltiplas conexões.

IO.Ethernet.SupressEcho

◩ Configure em Verdadeiro para eliminar o eco presente em uma comunicação. O eco é a recepção indesejada de uma cópia exata de todos os pacotes de dados que um Driver enviou para um equipamento.

IO.Ethernet.Transport

A Define o protocolo de transporte. Os valores possíveis são **T ou TCP**: Utiliza o protocolo TCP/IP ou **U ou UDP**: Utiliza o protocolo UDP/IP.

IO.Ethernet.UseIPv6

◩ Configure em Verdadeiro para utilizar endereços IPv6 em todas as conexões Ethernet ou configure em Falso para utilizar endereços IPv4 (padrão).

Histórico de Revisões do Driver

VERSÃO	DATA	AUTOR	COMENTÁRIOS
3.0.59	03/11/2026	M. Ludwig	• Corrigido um congelamento de envio para todos os IEDs, com desconexões falsas e perda de respostas quando um IED permanecia mais tempo do que o *time-out* de recepção da biblioteca **IOKit**

VERSÃO	DATA	AUTOR	COMENTÁRIOS
			(**IO.TimeoutMs**) sem enviar o restante de um pacote. Agora este Driver reconecta somente o IED afetado e valida os pacotes recebidos antes deste processamento (*Case 41155*).
3.0.58	15/06/2026	M. Ludwig	• Driver atualizado para a biblioteca **IOKit** versão **3.0** e Visual Studio 2022 (*Case 37980*).
3.0.56	20/04/2026	M. Salvador	• Melhorias no tratamento da camada XMPP e em mensagens de log (*Case 30839*). • Adicionado suporte ao protocolo PRP usando um serviço do Windows e configuração pela janela de propriedades deste Driver (*Case 27270*). • Agora é possível informar uma opção **None** para um dos adaptadores de rede na configuração do protocolo PRP (*Case 31538*). • Adicionada informação de versão ao protocolo PRP (*Case 31995*). • Agora os parâmetros como nome dos adaptadores de rede, endereço IP e porta UDP de estatísticas são lidos do arquivo ElipsePRPSvc.ini, e não mais de um projeto de aplicação. Criada uma propriedade com a lista de nomes de computadores e endereços IP para que este Driver identifique o computador e use a porta IP correspondente, permitindo que um mesmo projeto de aplicação possa ser usado em servidores redundantes sem alterações, ou seja, as diferenças permanecem no arquivo ElipsePRPSvc.ini (*Case 32144*). • Melhorias na configuração e implementação do protocolo PRP (*Case 34350*).

VERSÃO	DATA	AUTOR	COMENTÁRIOS
			• Realizado o mapeamento da propriedade **TimeQuality** para o Tag **Quality** com os valores de estampa de tempo inválido igual a 216 (*Non-specific local timestamp*, sub-status igual a 6, que até esta versão indicava 192), **Clock Failure** igual a 220 (*Local Override Local Timestamp*, sub-status igual a 7), **Clock not Synchronized** igual a 224 (introduzido) e sub-status igual a 8 (*Case 35905*). • Agora é possível realizar o *download* de arquivos quaisquer a partir de um IED, não somente de arquivos no formato **COMTRADE** (*Case 35507*). • Criada uma opção para identificação da revisão 2.1 do protocolo e revisado o procedimento de habilitação de **Reports** (*Case 37072*). • Adaptado o processo de desabilitar um **Report** para posteriormente apagar um **Dataset**, em conformidade com a revisão 2.1 do protocolo (*Case 30486*). • Ao habilitar um **Report**, o parâmetro *RptID* agora é comparado com o valor presente para definir se é necessária uma escrita, para evitar que um IED entenda este procedimento como uma mudança de configuração suficiente para apagar o *buffer* de eventos (*PurgeBuf*) (*Case 30721*). • Corrigida a possibilidade da conexão COTP não respeitar o *time-out* (*Case 30798*). • Agora os **Reports** são lidos novamente antes deste Driver tentar habilitar após o retorno de comunicação, evitando possíveis situações

VERSÃO	DATA	AUTOR	COMENTÁRIOS
			de erro em que o valor que está na memória deste Driver está diferente do IED, que pode ter sido alterado durante o período de uma queda de comunicação (*Case 30820*). • Este Driver agora habilita corretamente **Reports** trancados como se estivessem habilitados em um IED por algum tempo (*Case 30858*). • Arquivos no formato **COMTRADE** contendo caracteres especiais \ (barra invertida) ou / (barra) agora são salvos corretamente (*Case 30859*). • Se um **Dataset** é configurado incorretamente em um IED, ou seja, sem o parâmetro *t*, agora este Driver verifica se a estampa de tempo foi realmente lida antes de reportar o valor (*Case 31000*). • Revisado o procedimento de reconexão TCP/IP e MMS após uma desconexão (*Case 31133*). • Se uma mensagem de *polling* não é respondida por *time-out*, agora os Tags presentes nesta mensagem indicam uma qualidade ruim (*Case 31537*). • Correções no serviço PRP (*Case 31717*). • Os Tags **ComtradeInfo** e **ComtradeTransfer** não mostravam o estado corretamente quando um IED não possui arquivos (*Case 31877*). • A habilitação automática de **Reports** podia ter resultados inconsistentes devido à existência de itens com nomes parciais em **Datasets** ou **Reports** já habilitados (*Case 32417*).

VERSÃO	DATA	AUTOR	COMENTÁRIOS
			• Alterado o procedimento de logs para evitar uma falha em ambientes VMWare e corrigidos os filtros de recepção de pacotes em adaptadores físicos (*Case 32345*). • Este Driver agora está utiliza o SDK da biblioteca **Npcap** em substituição à biblioteca **WinPcap** (*Case 33489*). • Este Driver foi adaptado para baixar arquivos no formato **COMTRADE** se um IED tem muitos arquivos no diretório raiz e adicionado suporte a transferências baseadas em nome mais índice ao invés de comparação de datas (*Case 33741*). • Corrigido o procedimento de reconexão após uma queda da comunicação (*Case 33887*). • Adicionado suporte a Arrays em **Datasets** (*Case 34243*). • Corrigido um comportamento de falta de resposta de um método **GetFileDirectory** depois de uma primeira resposta com um parâmetro *MoreFollows* igual a Verdadeiro (*Case 34463*). • Melhorada a robustez no tratamento e recepção de comandos para evitar falhas ao receber mensagens mal formadas de servidores (*Case 35638*). • Após habilitar um **Report**, foi inserida uma retentativa de executar um comando de **General Interrogation** caso a primeira tentativa falhe. Além disto, se o usuário especifica a escrita da propriedade **Dataset** e esta escrita falha, foi corrigido um problema que

VERSÃO	DATA	AUTOR	COMENTÁRIOS
			escrevia um valor vazio nesta propriedade duas vezes, quando deveria escrever um valor vazio na primeira vez e depois escrever o valor correto do **Dataset** (*Case 36008*). • Corrigido um *loop* deste Driver, que não conseguia se reconectar a um IED desconectado após uma falha de inicialização causada pelo envio de um *Release Request* impróprio (*Case 36439*). • Corrigido um possível vazamento de memória em casos de transações MMS sem resposta (*Case 36492*). • Modificado o serviço PRP para utilizar operações em bloco (*Case 36720*). • Corrigido um erro de *Unsupported Operation* ao abrir a janela de propriedades de um Driver que foi salvo em versões anteriores (*Case 37224*). • Criados os Tags **DeviceConnectionStatus**, **WorkOnline** e **DeleteAllDevices** para monitorar e controlar a comunicação com IEDs (*Case 37238*). • Criado o Tag **GIAllReports** para forçar um comando de **General Interrogation** de todos os **Reports** habilitados por este Driver (*Case 37631*). • Criado o Tag Bloco **ReportItemsUnified** para receber todos os eventos de **Reports** por IED (*Case 37633*). • Removidas as referências ao **Elipse E3** na documentação deste Driver (*Case 36973*). • Corrigido um comportamento de não tratar corretamente *frames*

VERSÃO	DATA	AUTOR	COMENTÁRIOS
			maiores que 1500 bytes, aproximadamente (*Case 37512*). • Agora ao tentar reabilitar **Reports** depois de uma queda de comunicação e verificar que um **Report** está com o parâmetro *RptEna* igual a Verdadeiro, este Driver não tenta habilitar somente se o parâmetro *ResvTms* é igual a zero (*Case 37632*). • Revisado o funcionamento da camada de transporte XMPP e também da documentação (*Case 37788*).
3.0.17	21/12/2020	M. Salvador	• Adicionado suporte a *gateways*. • Arquivos de horário do último COMTRADE.
3.0.9	11/12/2019	M. Salvador	• Suporte ao protocolo PRP.
3.0.6	01/09/2019	M. Ludwig	• Driver portado para o Visual Studio 2017 (*Case 27095*).
3.0.1	04/06/2019	M. Salvador	• Implementada a Certificação IEC 61850 ED2.
2.0.23	08/02/2019	M. Salvador	• Implementada uma opção de espera na escrita do Tag **LastComtradeFileTime**, configurável na aba **Comtrade / Events** (*Case 25806*).
2.0.20	19/11/2018	M. Salvador	• Corrigido um vazamento de memória quando um servidor não implementa ou não responde um pedido de **GetVariableAccessAttributes** (*Case 25546*).
2.0.19	17/09/2018	M. Salvador	• Agora um IED que não suporta a utilização de **DataSets** dinâmicos pode ser usado junto com outros IEDs que suportam esta operação no mesmo Driver (*Case 25072*).
2.0.16	16/11/2017	M. Salvador	• Criar um Driver define como padrão o uso de um Tag para cada comando na

VERSÃO	DATA	AUTOR	COMENTÁRIOS
			opção **Single Tag Cmd Alias** (*Case 23693*). • Realizadas correções ao utilizar **Datasets** dinâmicos, incluindo o tratamento de reconexões e reinicialização de IEDs (*Case 23644*).
2.0.3	11/02/2014	M. Salvador	• Implementada a Certificação KEMA.
1.1.27	03/12/2012	M. Salvador	• Correções no processo de conexão e desconexão. • Corrigido o tratamento de **Reports** com **BitInclusion** incoerente com o tamanho de um **Dataset**. • Correções no processo de conexão e desconexão, criando uma nova *thread* de verificação do estado das conexões. • Modificado o tratamento de exceções de leituras. • Mensagem de **Reject** estava sendo tratada como **ConfirmedError** em alguns casos. • Correção de erros de tratamento de listas. • Tratamento de números reais inválidos (QNAN). • Correções na verificação de status da conexão. • Corrigida a informação de status de execução de um comando. • Opções adicionais para busca de arquivos no formato **COMTRADE**. • Correção de um possível *deadlock* entre semáforos na reconexão. • Bloco de status de COMTRADE. • Suporte a **EntryID** na inicialização de **Reports**. • Criação dinâmica de *devices*. • Propagação de eventos da biblioteca **IOKit**.

VERSÃO	DATA	AUTOR	COMENTÁRIOS
			• Correção na coleta de oscilografia para relés Areva e Alstom. • Suporte a tipos de dados **Double** em **mms_float**. • **DNSNames** com especificação de porta no endereço IP de um IED. • Correção na importação ICD SEL, proteção adicional para evitar escritas e *polling* quando em reconexão. • Suporte a *user session requirements* no Presentation CPA PPDU. • Correção de um vazamento de memória (VLD). • Correção de um vazamento de memória quando desconectado (COTP) e teste de semáforo SEL. • Log do fuso horário ao iniciar este Driver. • Considera *TimeZone Unknown* como válido para o cálculo do *offset* local. • Mudança no modo de reabilitação dos **Reports** depois da reconexão. • Não é mais permitido criar LDs, a não ser na leitura de arquivos LD ou após a execução de um comando **GetServerDirectory** (*Case 11378*). • Eliminados os zeros à esquerda dos endereços IP (*Case 11379*). • Eliminar zeros à esquerda não considerava o uso de porta depois do endereço IP. • Reprogramados os **Reports** completamente após uma reconexão. • Criados os Tags **IPSelect** e **IPSwitch** e removida a propriedade **TimeofEntry** do *polling*, se habilitada.

VERSÃO	DATA	AUTOR	COMENTÁRIOS
			• Implementada uma *thread* de verificação de status dos *hosts*. • Limite de 20 Tags para leitura por *scan* e de atualização do parâmetro *RptEna* quando há Tags no limbo. • Problema de recepção errada aleatória de pacotes nas camadas *Session* e *Presentation*. • Modificado o parâmetro *Nesting Level Requested* de cinco para seis (INGETEAM). • O comando **GetVariableAccessAtributes** é realizado um diretório abaixo caso falhe (INGETEAM). **EntryID** e **OptFlds** agora não são solicitados nos **OptFlds** quando um **Report** é do tipo **Unbuffered**. • Corrigida uma potencial falha dentro de **MMS::SendRequest**, que pode ocorrer quando há uma queda de conexão. • Melhoria de performance dos Tags de status. • Pedido de desconexão TCP/IP não estava sendo realizado quando o pedido de **MMS::Status** falhava ou não cobria todas as situações. • Modificada a forma de realizar uma operação de **ServerRebuild**, que agora é assíncrona. • Melhorada a performance de inicialização do sistema, principalmente na leitura dos arquivos LD.
1.0.1	31/03/2010	M. Salvador M. Bihre	• Corrigido o envio de números inteiros para comandos na propriedade **ctlNum** (*Case 11097*). • O parâmetro *WriteStatus* das operações de escrita pelo método **WriteEx** do

VERSÃO	DATA	AUTOR	COMENTÁRIOS
			Elipse Power não era informado em alguns casos (*Case 11104*). • Adicionado suporte a *download* de oscilografias. • Adicionado suporte a comandos assíncronos. • Adicionado suporte a Tags individuais para comandos. • Corrigido suporte a arquivos ICD e SCD. • Versão inicial deste Driver.

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

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

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

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

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

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

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

www.elipse.com.br
kb.elipse.com.br
forum.elipse.com.br
www.youtube.com/elipsesoftware
elipse@elipse.com.br

Gartner 2014 Cool Vendor

OPC FOUNDATION MEMBER

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)

Copyright © 2008-2026 Elipse Software Ltda. Todos os direitos reservados.