

Driver JCI Metasys

Nome do Arquivo	Metasys.dll
Fabricante	Johnson Controls International
Equipamentos	Servidores Metasys v10 ou superior
Protocolo	Metasys API v3/v4
Versão	1.0.14
Última Atualização	10/08/2026
Plataforma	Win32
Dependências	IOKIT 2.0
Leitura com Superblocos	Não utiliza superblocos
Nível	31314

Introdução

Este Driver implementa a comunicação com Servidores Metasys v10 ou superior da Johnson Controls International através da Metasys API v3/v4.

Configuração do Driver

A configuração do driver deve ser realizada através da caixa de diálogo de propriedades.

Adicionalmente, devem ser definidos manualmente os arquivos **MetasysState0.txt** e **MetasysState1.txt** contendo diversos nomes de estados que correspondem ao valor 0 e 1 respectivamente, para que o driver possa traduzir esses estados para valores numéricos 0 e 1.

Os arquivos devem conter uma lista de palavras com os diversos estados, uma palavra em cada linha, e uma palavra que estiver no arquivo para o estado 0, não pode estar no arquivo de estado 1, e vice-versa.

Todas as palavras são convertidas para minúsculas, portanto as palavras não são case-sensitive.

Linhas que comecem com o símbolo # serão consideradas comentários.

Exemplo:

MetasysState0.txt

Parado

Desligado

Local

Fechar

Inativo

Inactive

FALSO

Desativar

Deactivate

Normal

MetasysState1.txt

Início

Start

Reset

Ligado

Remoto

Abrir

Active

Ativo

VERDADEIRO

Ativar

Activate

Alarme

Também é possível informar um arquivo chamado **MetasysReadAhead.txt** que informa uma parte do nome de um comando que está associado a um respectivo tag de estado, para ser lido em sequencia após a espera de um tempo configurável na propriedade "Read Ahead after Cmd - Wait Time".

O arquivo será lido na partida do driver, com o seguinte conteúdo :

#comentários começam com #

#coloque aqui as palavras chave para associar um comando a um estado que deve ser lido o mais rapido possivel

#na seguinte forma

#keyword_comando;keyword_estado

#os objetos devem pertencer à mesma NAE

QDL-S;QDL-C

SF-COM;SF-STS

SYSENA;SF-STS

SF-CVF;SF-SVF

O procedimento de busca se dá verificando se o comando que foi executado (com sucesso) contém a primeira keyword, substitui-se essa keyword pela segunda e verifica se existe algum tag com esse nome, se existir então coloca esse tag numa fila especial, esperando o tempo “Read Ahead After Cmd – Wait Time” para ser lido.

Configuração das Propriedades

Os parâmetros de configuração para efetuar comunicação com o sistema Metasys, estão disponíveis na Janela de Propriedades do Driver.

PROPRIEDADE	DESCRIÇÃO
Metasys Server	Nome DNS ou endereço IP do Servidor Metasys ou concentrador NAE
User/Password	Nome do usuário e senha para Login
User2/Password2	Nome do usuário e senha para Login alternativo, em caso de falha do primeiro usuário (depois de 2 tentativas de login)
Cache File Path	Nome de diretório onde serão salvos e lidos os arquivos de cache e dos significados das Enums. Ao total são 3 tipos de arquivos: - Arquivo de GUID's (MetasysGUID.txt) contém os códigos internos de cada objeto Metasys, que são armazenados em arquivo cache pelo driver para acelerar o processo de inicialização - Arquivos de Cache de Comandos e Estados

PROPRIEDADE	DESCRIÇÃO
	(DeviceName_MetasysStateCache.txt) contém a descrição do significado de estados e comandos dos objetos marcados com N1>0, que são armazenados em um arquivo de cache por controladora NAE, para acelerar o processo de inicialização e tradução dos significados para valores numéricos 0 e 1. - Arquivos de nomes de estados (MetasysState0.txt e MetasysState1.txt) contém os nomes dos diversos estados que correspondem ao valor 0 e ao valor 1 respectivamente. Se não for informado um diretório, os arquivos serão salvos na mesma pasta onde está a dll do driver.
Force New Login After (s)	Força um novo login independente da data de expiração do token. Deixe em 0 para desabilitar.
Protocol Version	V3: Utiliza apenas leitura por polling V4: Permite a leitura por polling e por eventos (assinaturas)
Read Method	Polling: realiza a leitura dos tags por polling, pode ser usado na V3 e V4; Events: realiza a leitura dos tags por assinaturas, pode ser usado apenas na V4; ByTag(N2): permite escolher a cada tag, no parametro N2, se vai ser lido por polling ou por eventos. Pode ser usado apenas na V4, sendo que N2=0 (polling) e N2=1 (eventos).
Subscription Renew (s)	Quando usado o método de leitura de um tag via eventos, caso o valor do tag não mude dentro deste intervalo máximo em segundos, será feita uma nova assinatura, o que força a atualização do tag. Caso o valor recebido seja identico ao existente, o tag não será atualizado, exceto se a propriedade EnableDeadBand estiver em FALSE.
Polling Options	Opções relacionadas ao funcionamento via polling
Items per Read Request	Indica quantos itens podem ser solicitados no máximo a cada pedido de comunicação. O valor mínimo é 1. Esta opção também é usada nas assinaturas de eventos para definir quantos itens podem ser assinados na mesma mensagem.
Inter-Request Delay (ms)	Informa um tempo padrão em milisegundos de atraso entre pedidos de leitura para todas as NAEs. Será usado caso uma NAE não tenha seu proprio delay informado no tag InterReqDelayMS.
Force New Login After Consec Errors	Força um novo login depois de um numero de erros de leitura consecutivos. Deixe em 0 para desabilitar.

PROPRIEDADE	DESCRIÇÃO
Force New Login with Read Error Rate > %	Força um novo login se a taxa média de erros (em porcentagem) for maior do que a informada neste campo. Deixe em 0 para desabilitar.
Read Ahead After Cmd - Wait Time (s)	Tempo de espera para realizar a leitura de um tag de status associado a um comando. Esta associação é feita através do arquivo MetasysReadAhead.txt
Full Logs	Informa se os logs do driver (quando habilitados) serão gerados com informações detalhadas da comunicação.

Além da Janela de Propriedades, estas configurações também podem ser definidas em tempo de execução nas aplicações **NaN** ou **Elipse Power**. Para isto, inicialize o Driver em modo **Offline**, ou seja, execute a aplicação com a opção **Start driver OFFLINE** habilitada, configurável na aba **Setup** da Janela de Propriedades. As opções de configuração deste Driver estão descritas na tabela a seguir.

Opções de configuração do Driver JCI Metasys

PARÂMETRO	STRING DA PROPRIEDADE OFFLINE	TIPO DE DADO	DESCRIÇÃO
Metasys Server	Metasys.UseChecksum	Texto	Nome do Servidor Metasys
User	Metasys.User	Texto	Usuário para Login
Password	Metasys.Password	Texto	Senha para Login
Cache File Path	Metasys.GUIDPath	Texto	Diretório de Cache
Items per Read Request	Metasys.MaxItemsRead	Texto	Numero máximo de itens por pedido de leitura
Protocol Version	Metasys.APIVersion	DWord	3 ou 4
Read Method	Metasys.ReadConfig	DWord	0 = polling, 1 =Eventos, 2 = by Tag
Inter Request Delay	Metasys.InterReqDelay	Dword	Delay entre leituras
Force New Login after (s)	Metasys.ForceLoginTime	Dword	Forçar novo login após x segundos

PARÂMETRO	STRING DA PROPRIEDADE OFFLINE	TIPO DE DADO	DESCRIÇÃO
Force New Login after consec errors	Metasys.ForceLoginConsecErrors	Dword	Forçar novo login após x erros consecutivos
Force New Login with read error rate	Metasys.ForceLoginReadErrorRate	Dword	Forçar novo login se taxa de falha for superior a x
Read Ahead After Cmd - Wait Time	Metasys.ReadAheadWaitTime	Dword	Tempo de espera para leitura de status após comando
Full Logs	Metasys.FullLogs	Int	Habilita Logs completos

Estas propriedades podem ser escritas através de um tag com os parâmetros *N1* igual a -1 (menos um), *N2* igual a 0 (zero), *N3* igual a 0 (zero) e *N4* igual a 3 (três).

Referência de Tags

Devem ser utilizados IOTags (Tags individuais) para a comunicação.

Os tags são endereçados através dos parâmetros **Device**, **Item**, **N1**, **N2** e **N3** no seguinte formato:

Propriedade	Significado
Device	Nome_do_Servidor_Metasys:Concentrador_NAE
Item	ObjetoMetasys;propriedade{Extra} Veja mais detalhes abaixo em Endereçamento :
N1	Tipo de Objeto 0 = Padrão 1 = Enum (será convertido para numero) 2 = Analogico com limites de alarmes
N2	Tipo de Leitura, usado caso a propriedade Read Method = By Tag (N2) 0 = Polling 1 = Eventos
N3	Retentativas para leitura de limites de alarmes (N1 = 2)

Endereçamento

No parametro Item, **ObjetoMetasys** é o nome de algum objeto no sistema, e **propriedade** é um parâmetro opcional, que informa a propriedade a ser lida do objeto. Se não for informada, a propriedade a ser lida será **presentValue**.

Caso seja informada a propriedade especial **PVALM**, serão lidas as propriedades presentValue e alarmState, sendo que a primeira será mapeada para a propriedade Value do tag e a segunda para a propriedade Quality. Se o valor de alarmState for HighAlarm ou Alarm, isto será mapeado para o campo OPC_LIMIT_HIGH (bit 1 do Quality) e se for LowAlarm será mapeado para o campo OPC_LIMIT_LOW (bit 0 do Quality), resultando respectivamente nos valores 194 (OPC_GOOD + OPC_LIMIT_HIGH) e 193 (OPC_GOOD + OPC_LIMIT_LOW).

Extra é um texto no formato JSON que pode ser uma lista de estados discretos (States) ou o nome de um Alarme (Alarm).

States é um texto JSON onde podemos declarar o significado dos estados para este tag em específico, sendo que o driver dará preferência a tradução de estados declarados no tag, caso não sejam encontrados, aí sim utiliza os textos que foram declarados nos arquivos de estado 0 e 1.

O formato esperado para o States é o seguinte:

```
{"states" : [ {"Nome1":Valor1}, {"Nome2":Valor2} , {"NomeN":ValorN}]}
```

Alarm é um texto JSON onde declaramos, somente para tags analógicos, o nome de um alarme, no formato:

```
{"alarmName":"Name"}
```

Este objeto de alarme terá seus limites (Baixo e Alto, Warning e Alarm) buscados e salvos em um arquivo, através do tag especial "GetAlarmLimits".

Alguns exemplos de tags:

Device: WIN-CGVCS5PQ7TV:NAE4561 Item: Programming.AV1 => será lido o presentValue

Device: WIN-CGVCS5PQ7TV:NAE4561 Item: Programming.AV1;presentValue => será lido o presentValue

Device: WIN-CGVCS5PQ7TV:NAE4561 Item: Programming.AV1;PVALM => será lido o presentValue e o alarmState conforme explicação acima

Device: WIN-CGVCS5PQ7TV:NAE4561 Item: Programming.AV1{"alarmName":"Alarm1"} => será lido o presentValue com o nome do objeto de Alarme como sendo "Alarm1"

Device: WIN-CGVCS5PQ7TV:NAE4561 Item: FieldBus1.4oPav.FX34.BinaryInputs;status => será lido a propriedade status do objeto

Device: WIN-CGVCS5PQ7TV:NAE4561 Item: Programming.AV1{"states":[{"Local":0}, {"Remoto":1}, {"Automático":2}]} => será lido o presentValue, interpretando seu conteúdo conforme os estados Local, Remoto e Automático.

Nota: Caso o valor obtido da propriedade seja um Enum e N1=1, o driver tentará traduzir o texto da Enum para um valor correspondente, através do mapeamento de comandos e de declaração dos nomes de estados.

Para a escrita, o driver obtém o nome do comando na propriedade e envia o que for recebido no tag, fazendo a tradução para o Enum correspondente caso seja necessário.

Tags Especiais

Nome	Device	Item	Read/Write
ClearStateCache	Nome_do_Servidor_Metasys:Concentrador_NAE	ClearStateCache	R/W

FlushGUID	Endereço de um NAE válido qualquer	FlushGUID	W	
GetGUID	Nome_do_Servidor_Metasys:Concentrador_NAE	GetGUID:ObjectName	R	
GetDeviceEnumCount	Nome_do_Servidor_Metasys:Concentrador_NAE	GetDeviceEnumCount	R	
GetAlarmLimits	Nome_do_Servidor_Metasys:Concentrador_NAE	GetAlarmLimits	R/W	
InterReqDelayMS	Nome_do_Servidor_Metasys:Concentrador_NAE	InterReqDelayMS	W	
LoginStatus	Nome_do_Servidor_Metasys:Concentrador_NAE	LoginStatus	R	
MaxItemsRead	Metasys_Server_Name:NAE_Device	MaxItemsRead	W	
CallStats	Se vazio: Obtem totais de todo o driver, ou para obter de uma NAE: Metasys_Server_Name:NAE_Device	CallStats5 (5 minutos) CallStats60 (1 hora)	R	

SubscriptionStats	Se vazio: Obtem totais de todo o driver, ou para obter de uma NAE: Metasys_Server_Name:NAE_Device	SubscriptionStats5 (5 minutos) SubscriptionStats60 (1 hora)	R	

Histórico de Revisões do Driver

VERSÃO	DATA	AUTOR	COMENTÁRIOS
1.0.14	10/08/2026	M. Salvador	• [BUG] Corrigida situação que gerava logins excessivos em caso de falha de comunicação. [36510] { surgiu no Metasys v1.00 Build 8 (aproximado), corrigido no Metasys v1.00 Build 12 } • [BUG] Implementado bloqueio de tags configurados com erros de sintaxe. [36946] { corrigido no Metasys v1.00 Build 13 } • [BUG] - Agora novos devices podem ser adicionados com o driver em execução. [38214] { corrigido no Metasys v1.00 Build 13 }
1.0.13	18/06/2025	M. Ludwig	Driver atualizado para a biblioteca IOKit versão 3.0 e Visual Studio 2022 (Case 37987).
1.0.9	21/03/2024	M. Salvador	Suporte à versão 4 da API.
1.0.8	20/09/2022	M. Salvador / M.Ludwig	Versão inicial do driver.