

# PROTOCOLO DE COMUNICAÇÃO DNP3



Distributed  
Network  
Protocol

**ÍNDICE**

ÍNDICE.....	1
INTRODUÇÃO .....	2
DNP3.....	2
APLICAÇÕES DO PROTOCOLO.....	2
ESPECIFICAÇÕES E SUBCONJUNTURAS .....	3
RELATÓRIO DE DADOS.....	3
ESTRUTURA DE MENSAGEM .....	5
CÓDIGOS DE FUNÇÃO DE REQUISIÇÃO E RESPOSTA.....	8
REFERÊNCIA.....	8

## INTRODUÇÃO

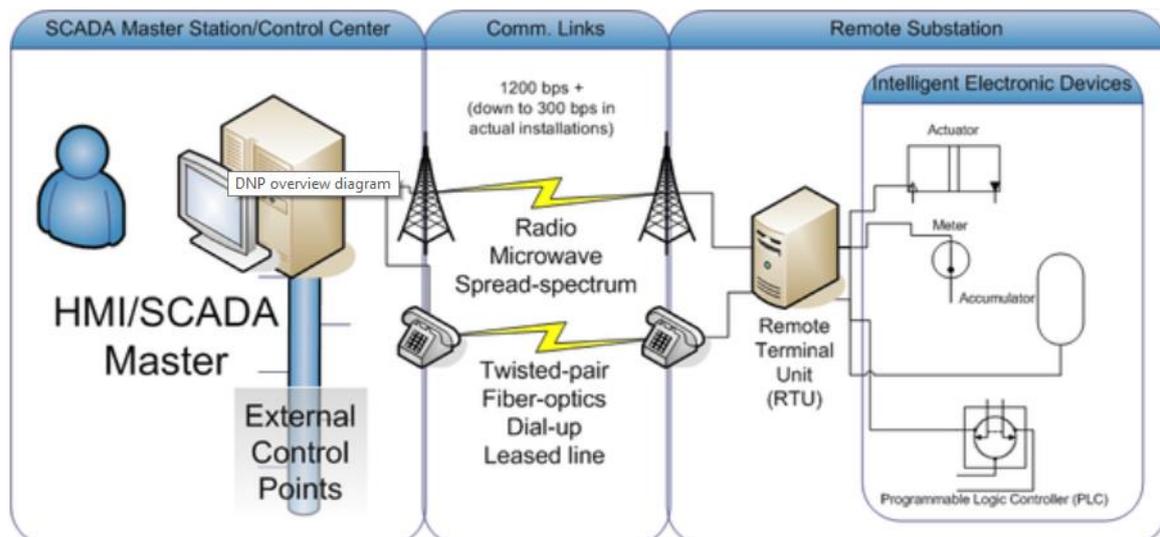
O objetivo deste artigo é introduzir conceitos básicos do Protocolo de Comunicação DNP3 (*Distributed Network Protocol – Protocolo de Rede Distribuída*) a fim de instruir o usuário com os principais aspectos de sua aplicação. Basicamente, o DNP3, similar ao Modbus, é um padrão de comunicação onde diferentes máquinas podem comunicar-se entre si. Permite o monitoramento e controle remoto.

## DNP3

Enquanto o Protocolo IEC 60870-5 estava em fase de desenvolvimento e ainda não havia sido normalizado, existia uma necessidade de criar-se um padrão que pudesse permitir a interoperabilidade entre vários componentes fornecedores de dados SCADA para a rede elétrica. Foi então, em 1993, que a GE-Harris Canada (antigamente conhecida como Westronic Inc) usou as especificações do ainda incompleto IEC-60870-5 como fundamento para um Protocolo aberto e com aplicabilidade imediata que atendia especificamente os requerimentos Norte-Americanos e estabeleceu seu comitê técnico em 1995. Hoje em dia o DNP3 já é amplamente difundido mundo a fora, como por exemplo, nas Américas do Sul e Norte, Reino Unido, Austrália, Nova Zelândia, África do Sul e em algumas partes da Ásia. O Protocolo de Comunicação DNP3 é projetado para permitir uma comunicação de dados confiável em ambientes com adversidades das quais um sistema de automação industrial está sujeito. Tais como distúrbios elétricos, vida-útil de componentes (Sua previsão de vida-útil pode ser prolongada em décadas) e baixa qualidade de transmissão de dados.

## APLICAÇÕES DO PROTOCOLO

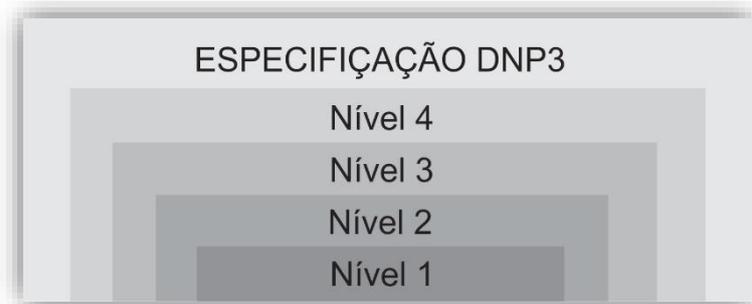
O objetivo primário do DNP3 foi de atender indústrias do segmento elétrico, posteriormente, indústrias de Óleo e Gás, Água. Os Protocolos DNP e IEC 90870-5-101 são especificados pela IEEE P1379 como prática recomendada para comunicação de dados entre IED's (*Intelligent Electronic Device – Dispositivos Eletrônicos Inteligentes*) e RTU (*Remote Terminal Unit – Unidade Terminal Remota*).



*Figura 1 – Protocolo de Comunicação DNP 3.0*

## ESPECIFICAÇÕES E SUBCONJUNTURAS

O entendimento das Especificações do DNP3 é bem intuitivo. Não seria prático tentar implementar cada uma das inúmeras possibilidades descritas em Especificações. Portanto, o grupo de usuários adotou 4 níveis de subconjunturas. Estas 4 subconjunturas quebram as especificações do DNP3 em “pedaços” mais administráveis. Cada nível representa um acordo de interoperabilidade entre os componentes da conjuntura especificada. Cada nível oferece uma maneira de identificar funções e objetos comumente usados e cada conjuntura inclui as funcionalidades da conjuntura anterior. Importante ressaltar que uma operação “Normal” com o DNP3 apenas inclui as funções de Nível 1



*Figura 2 – Níveis de Especificação*

- **Nível 1:** Esta conjuntura é projetada para pequenos IED's. Utiliza pesquisas simples, incluindo classes de pesquisas (class polls);
- **Nível 2:** Esta conjuntura é projetada para grandes IED's e pequenos RTU's. Adiciona Tipos de Objetos de Pesquisa (Object Type Polls), Formatos Extra de Dados (Extra Data Formats) e outras características úteis, tal como Congelamento de Contador (Counter Freeze);
- **Nível 3:** Esta conjuntura é projetada para IED's muito grandes, pode pesquisar quaisquer combinações de Objetos e Ranges e Reatribuições de Classes.
- **Nível 4:** Esta Conjuntura foi aprovada em uma reunião dos Users group em fevereiro de 2007. Inclui todos os tipos “básicos” de dados. Também inclui auto-endereçamento, formatos adicionais de dados (Object Group 0, Device Attributes, Double-bit inputs), variações com tempo para Frozen Counters, Frozen Counters Events e Events Analog Inputs, Variações de ponto de flutuação para saídas/entradas analógicas e entrada analógica que reporta Banda-morta;

## RELATÓRIO DE DADOS

Para um melhor entendimento de dados pode ser obtido analisando *Data Classes (Classe de Dados)*, *Data Acquisition Methods (Métodos de Aquisição de Dados)* e *Report by Exception (Reportado por exceção)*. Em Data Classes:

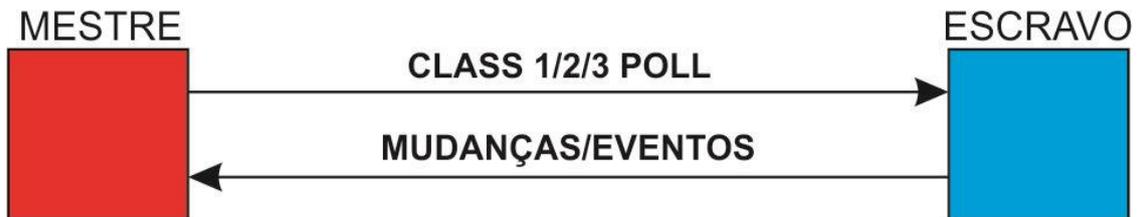
- **CLASS 0:** Onde todos os *Static Data (Dados Estáticos)* são atribuídos;
- **CLASS 1 / CLASS 2 / CLASS 3:** Onde *Event Data (Dados Eventuais)* são opcionalmente atribuídas a uma dessas classes;

O Termo **Static Data** em DNP3, aponta o valor atual. O comando Class 0 poll (*static data*) retorna o valor atual de todos pontos.



*Figura 3 – Exemplo de Class 0 Poll*

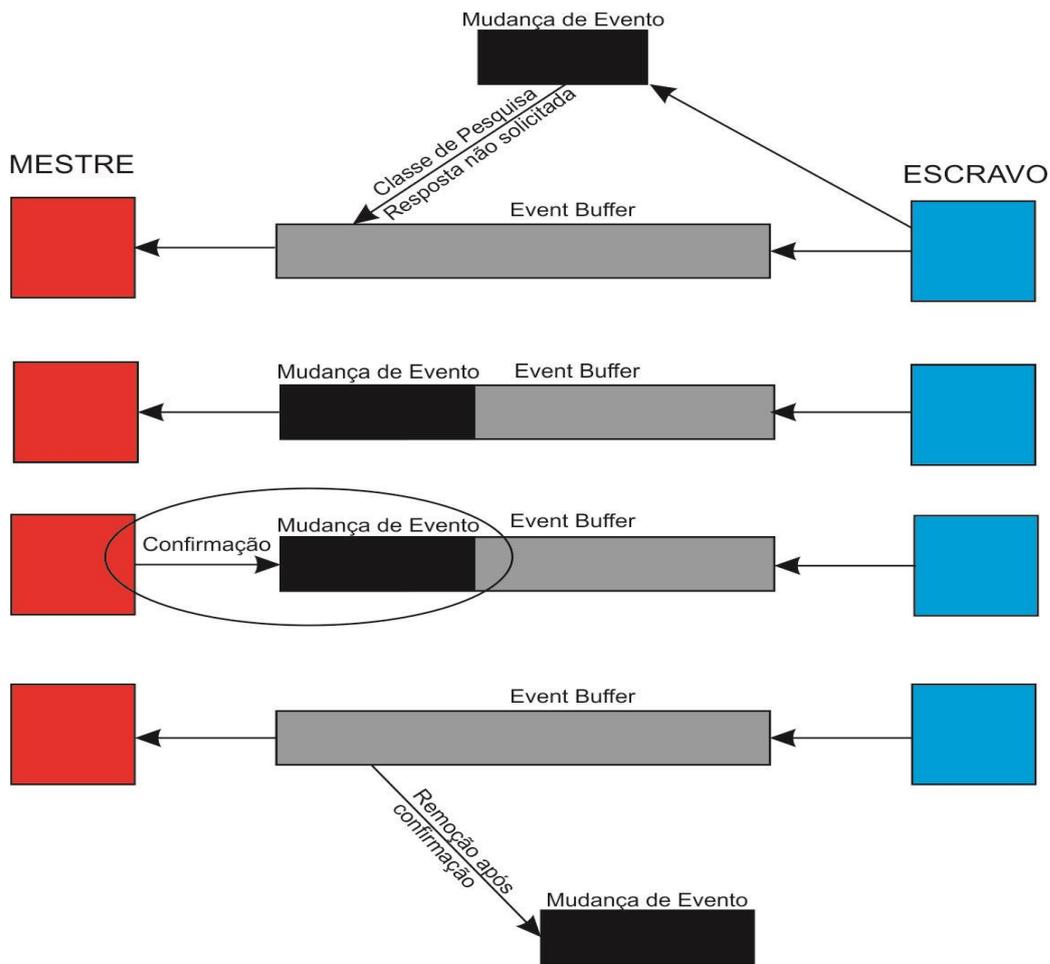
O Termo **Event Data** em DNP3, *Eventos* estão associados com algo de significativo está acontecendo. Como por exemplo, mudanças no valor de dados, quando valores cruzam o limiar, ou quando os dados excedem a banda morta. O Protocolo DNP3 oferece 3 classes de eventos, Classe 1, Classe 2 e Classe 3. Mas, as especificações do DNP3 não atribuem nenhuma significância para essas 3 classes. Algumas instalações atribuem a entrada de dados binários em Class 1 e entrada analógicas em Class 2, porém, os dados podem ser atribuídos as Classes de qualquer maneira que faça sentido para o sistema. As Classes podem ser atribuídas e mudadas via Protocolo.



*Figura 4 – Exemplo de Class 1/2/3 Poll*

O DNP3 também oferece **Report by Exception ou RBE**, com esse comando, os escravos apenas respondem mudanças de dados. RBE pode ser usado com Polling (pesquisas) ou com *Unsolicited responses*. E com RBE é configurável e o escravo pode adicionar todas ou apenas a recente mudança de evento em Event Bbuffer e o tipo de dados que for adicionado pode ser configurado por *datatype*. Os dados de *Event Buffer* podem ser enviados via *Unsolicited Response* ou como resposta de um *Class Poll*. A informação de mudança de evento adicionada pelo escravo permanece no buffer de evento até ser reconhecida pelo Mestre. *Events* ou mudança de dados, podem ser relatados como:

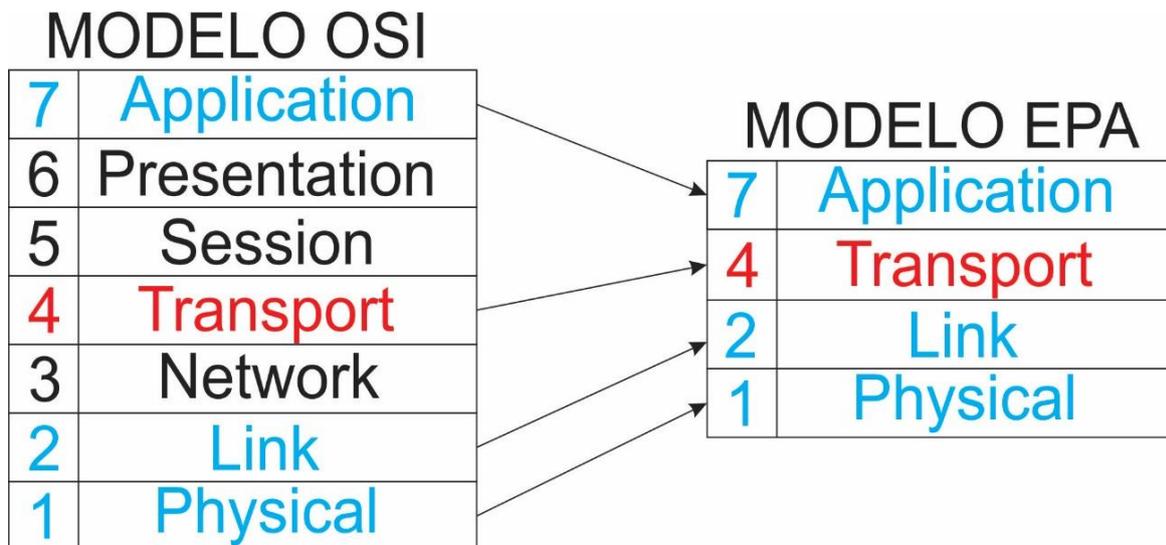
- **Sequence of Events (Sequência de Eventos):** Onde todos os Eventos são reportados, geralmente com *TimeStamps*. Entradas digitais, por exemplo, são reportadas através deste método.
- **Last Change (Última mudança):** Onde apenas a mudança mais recente é reportada, geralmente sem *TimeStamps*. Entradas analógicas, por exemplo, são reportadas através deste método.



*Figura 5 – Exemplo de Report by Exception*

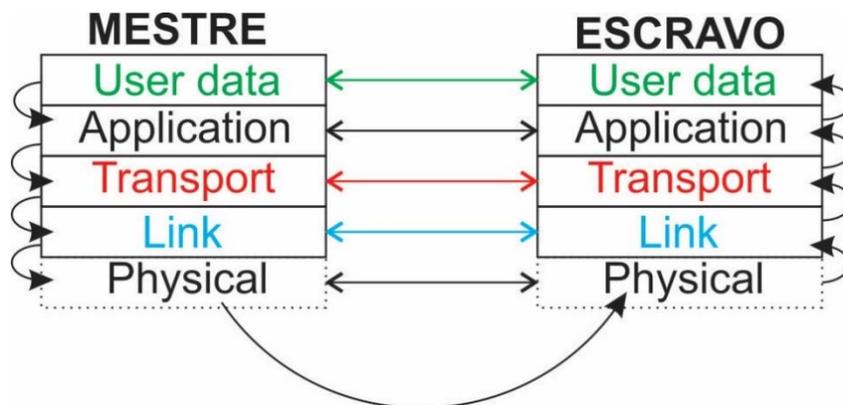
## ESTRUTURA DE MENSAGEM

A ISO (*International Standardization Organization - Organização Internacional de Padronização*) definiu a arquitetura de comunicação chamada de modelo de referência OSI (*Open System Interconnection – Interconexão de Sistemas Abertos*). Esse modelo separa a funcionalidade em 7 camadas diferentes. É costumeiro referenciar o modelo OSI em debates sobre Protocolos de Comunicação, embora poucos Protocolos de Comunicação implementem todas as 7 camadas em sua estrutura. O Protocolo de comunicação DNP3 utiliza uma versão simplificada de apenas 3 camadas do modelo OSI, chamada **EPA** (*Enhanced Performance Application – Melhora de Performance de Aplicação*). São elas, **Application** (*Aplicação*), **Data Link** (*Link de Dados*) e **Physical** (*Física*). O DNP3 também adiciona parcialmente a “função” **Transport** que fornece algumas e não todas as funcionalidades desta camada. Como permitir mensagens maiores do que o frame da camada **Data Link** permite.



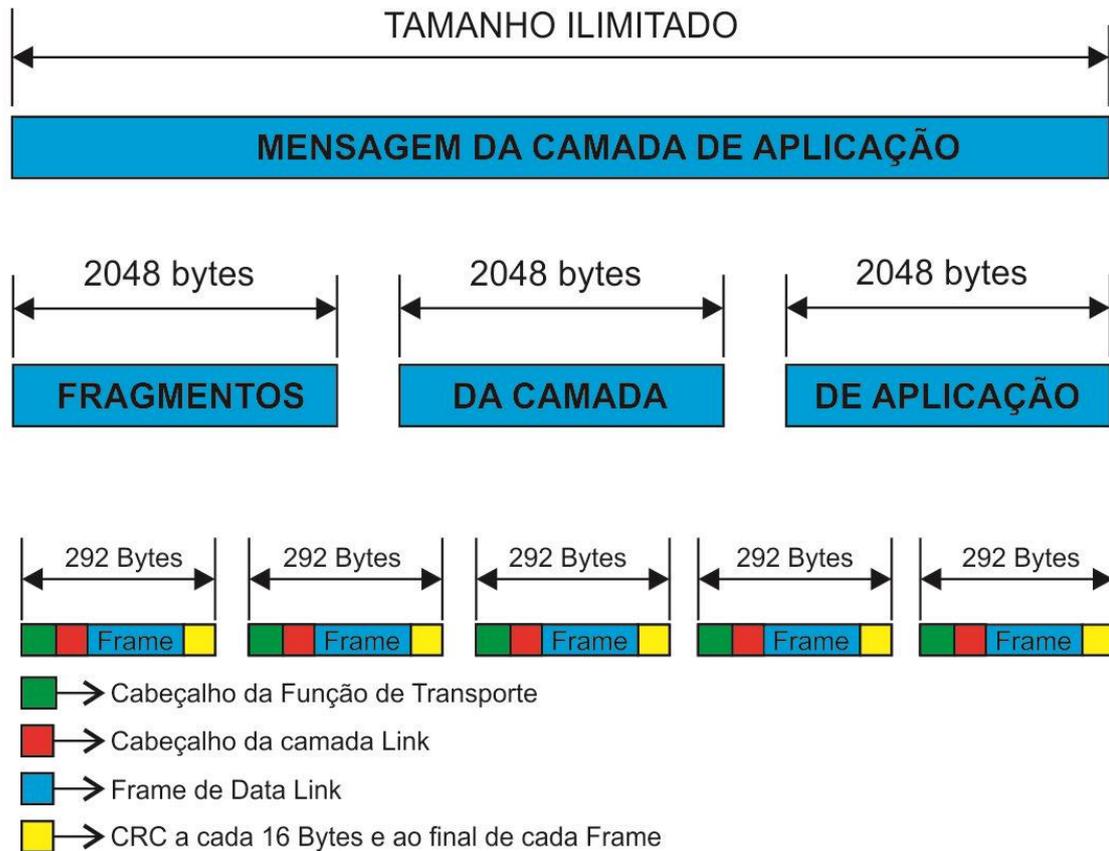
*Figura 6 – Comunicação Camada por Camada*

Em teoria, cada uma das camadas EPA estão configuradas para se comunicar com as mesmas camadas do outro dispositivo. Cada camada funciona independentemente das camadas acima e abaixo. Durante o envio de mensagem, cada camada do Dispositivo Mestre receberá informações providas das camadas superiores e cada camada adicionará mais informações no intuito de que a camada equivalente no Dispositivo Escravo processe apropriadamente a mensagem recebida. Durante o recebimento da mensagem, cada camada examina as informações nos dados recebidos para processar os dados corretamente de suas camadas equivalentes. A camada identifica o recebimento e envia para a camada acima que repete o procedimento até a camada do topo. Em suma, o processo de comunicação é de “Camada por Camada”.



*Figura 7 – Transporte de Mensagens*

A função da camada de Transporte do DNP3 “quebra” mensagens grandes em mensagens menores para que possam ser gerenciadas pela Camada Link durante o envio. Durante o recebimento a Camada Transporte “monta” a mensagem “quebrada” para enviar para camada de aplicação. O Protocolo DNP3 não restringe o tamanho de uma mensagem. Se uma mensagem é muito grande para camada LINK será fragmentada na camada de aplicação em porções de 2048 Bytes, usualmente. Cada fragmento é “quebrado” em frames menores, de tamanho máximo de 292 Bytes, onde os cabeçalhos das funções de Transport e de Data link são adicionados. O CRC é adicionado após os cabeçalhos, a cada 16 bytes, e ao final de cada frame.



*Figura 8 – Fragmentação de Mensagens*

**CÓDIGOS DE FUNÇÃO DE REQUISIÇÃO E RESPOSTA**

A tabela 1 a seguir detêm todos os códigos de função utilizados em mensagens de requisição:

<b>CÓDIGO (hex.)</b>	<b>DESCRIÇÃO</b>
0x00	Confirmação
0x01	Leitura
0x02	Escrita
0x03 - 0x06	Operações de Controle
0x07 - 0x0C	Operações de Congelamento
0x0D - 0x0E	Restart Frio/Morno
0x0F - 0x10	Inicializar DATA/APPLICATION
0x11 - 0x12	Iniciar/Parar Aplicação
0x013	Salvar Configuração
0x14 - 0x15	Habilitar/Desabilitar Mensagem Não-Solicitada
0x17	Medição de Atraso
0x18	Atribuir Classe
0x19 - 0x1E	Operações de Transferência de Arquivo
0x1F	Ativar Configuração
0x20 - 0x22	Autenticação

*Tabela 1 – Códigos de Função de Requisição*

A tabela 2 a seguir detêm os códigos de função utilizados em mensagens de respostas estão limitados

<b>CÓDIGO (hex)</b>	<b>DESCRIÇÃO</b>
0x81	Resposta
0x082	Mensagem Não-Solicitada
0x083 - 0x084	Autenticação

*Tabela 2 – Códigos de Função de Resposta*

**REFERÊNCIA**

<http://www.trianglemicroworks.com/SubVideo/DNP3%20Training%20Course/player.html>