GE Energy

DNP Protocol V3.00

NOTICE OF COPYRIGHT & PROPRIETARY RIGHTS

TRADEMARK NOTICES

Table of Contents

Course Overview

Objectives:

1. Develop an understanding of DNP 3.0 on a protocol level.
2. Analyze typical DNP messages.

Prerequisite:

Basic understanding of SCADA concepts, network protocols and data communication.

Reference Materials:

- DNP DPA Configuration Guide (B021_0CG)
- DNP DPA Functional Specification (B021_0FS)
- DNP DCA Configuration Guide (B023_0CG)
- DNP DCA Functional Specification (B023_0FS)
- DNP Basic 4 Documentation (944-0007)

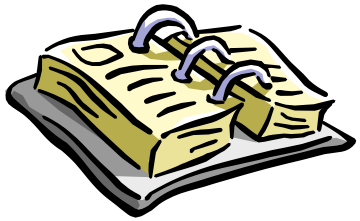
Recommended Schedule

Day 1

- Introduction to DNP V3.0
- DNP vs. IEC 870-5
- Using DNP / The GE Implementation
- Configuring DNP
- Introduction to DNP Message Structure

Day 2

- Conclusion of DNP Message Structure
- DNP Message Labs



1 Introduction to DNP

Section 1 – Introduction to DNP

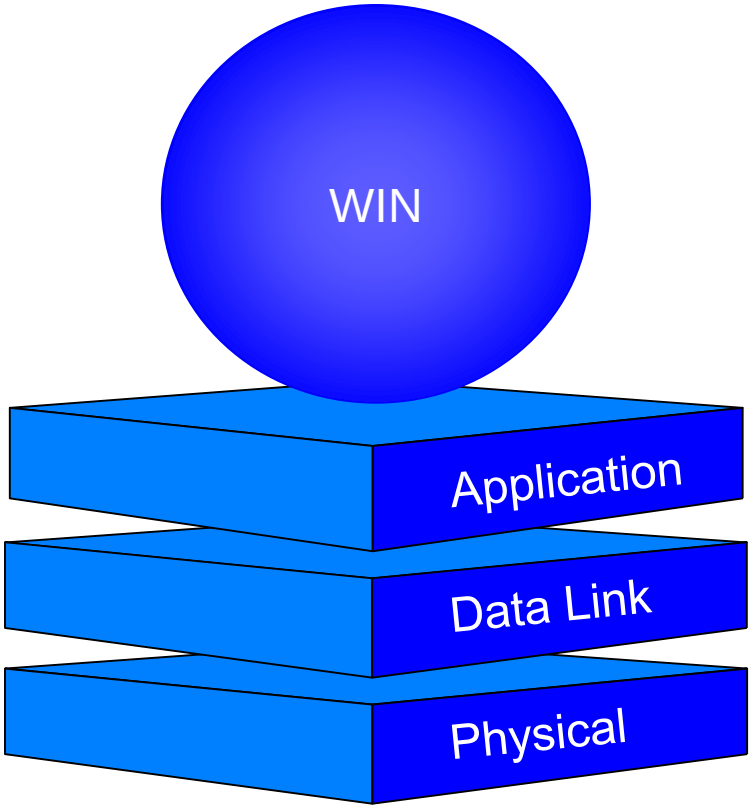
Objectives:

1. State the origin of the DNP protocol.
2. Explain the basic application of DNP protocol.

Section 1 – Introduction to DNP – cont.

Highlights

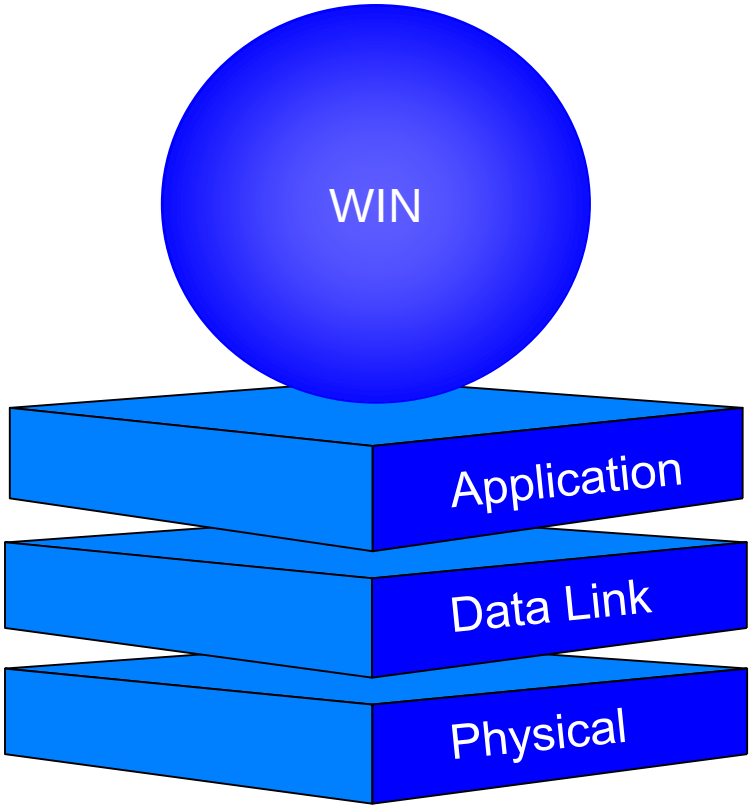
- Distributed Network Protocol (DNP)
- Open Systems Protocol Stack
- Recommended by IEEE for RTU to IED messages
- Based on IEC 3-layer version of 7-layer OSI model
- Developed by Harris Energy Control Systems (now GE Energy)
- Controlled by DNP Users Group



Section 1 – Introduction to DNP – cont.

DNP Features

- Asynchronous, code transparent
- Addresses over 65,000 devices
- Addresses over 4,000,000,000 data points of each data type per device
- Broadcast Messages
- Configuration/File Transfer
- Time of Day and Date Synchronization
- Time-Stamped Event Data
- Data Priority Levels (classes)



Section 1 – Introduction to DNP – cont.

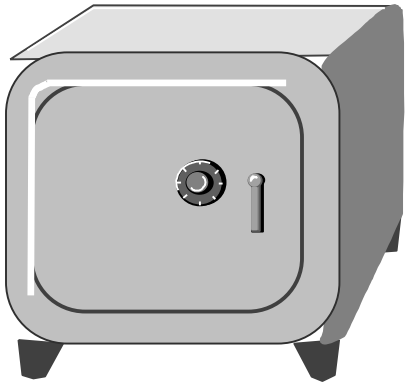
DNP is Interoperable

- Based on IEC 870-5 standards
- Implemented by many vendors; list growing daily
- Specified by many customers worldwide
- Not owned by any one company
- Users have choice of vendors

Section 1 – Introduction to DNP – cont.

DNP is Robust

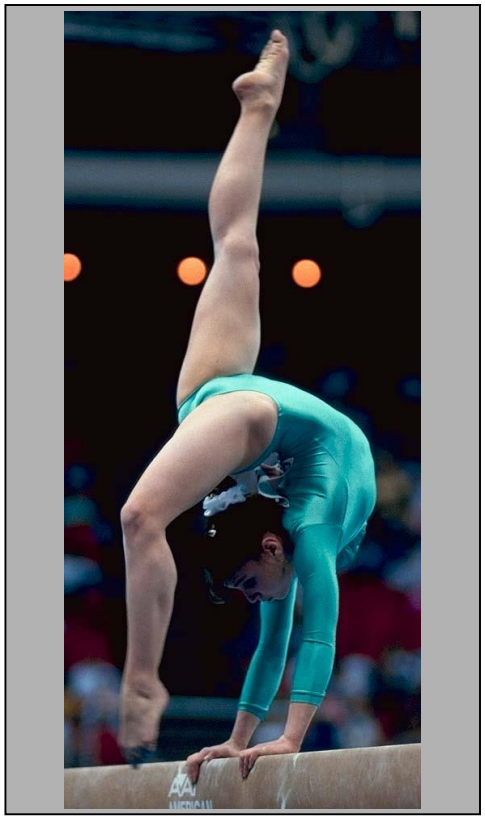
- Intended for low-to-medium speed media
- 16-Bit data link CRC every 16 octets
- Hamming distance of 6
- Optional data link confirmation
- Optional application confirmation
- Very efficient for a layered protocol



Section 1 – Introduction to DNP – cont.

DNP is flexible:

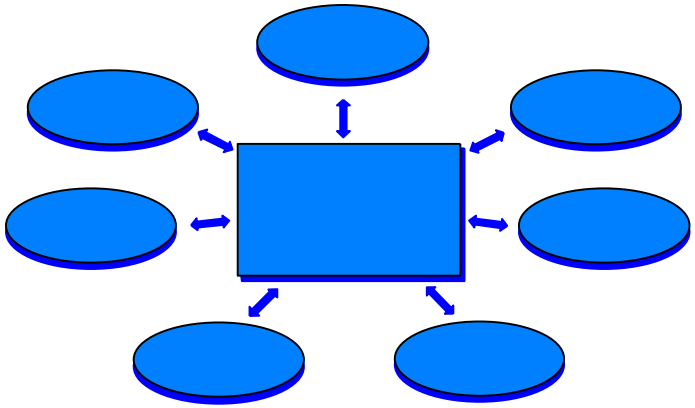
- Object-based application layer
- Objects provided for most generic data formats
- Add new objects via DNP Users Group
- Devices can choose which objects to support
- Standard implementation subsets defined



Section 1 – Introduction to DNP – cont.

Fits a Variety of Networks

- Multiple operating modes
  - Poll-Response
  - Polled Report-by-Exception
  - Unsolicited Response
  - Peer-to-Peer
- Layered protocol allows mix-and-match
- Suited to large distributed systems
- Simple implementations for IEDs
- Complex implementations for data concentrators
- Permits Multiple Masters
- Encourages Distributed Intelligence



Section 1 – Introduction to DNP – cont.

DNP Users Group

- Controls DNP as of November 1993
- Members include individuals from utilities and vendors
- Meet twice per year
- Evaluates modifications to the protocol
- Evaluates modifications to documentation

Section 1 – Introduction to DNP – cont. Review Questions

1. What IEC standard is DNP based on?
2. Name 5 features of DNP?
3. Who controls DNP?
4. How many devices can DNP address?
5. How many data points can DNP address?

2

DNP vs. 870-5

Section 2 – DNP vs. 870-5

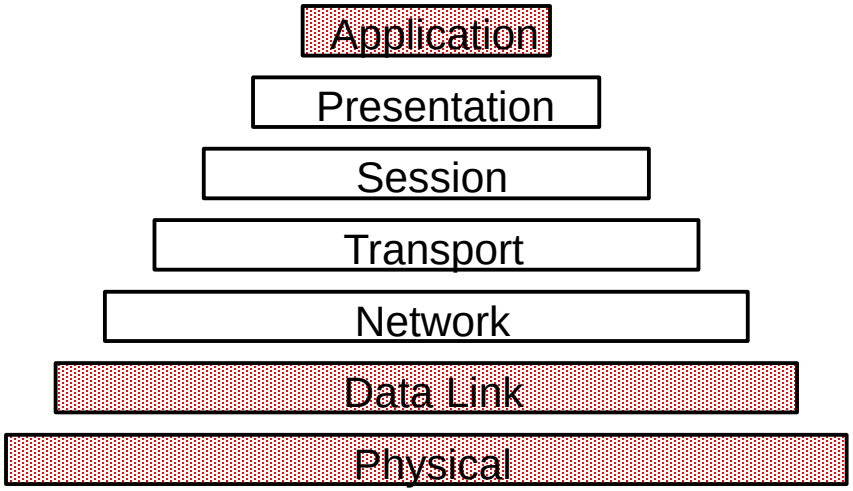
Objectives:

1. Describe the features of the IEC 870-5 standard.
2. State the differences between DNP and the IEC standard.

Section 2 – DNP vs. 870-5 – cont.

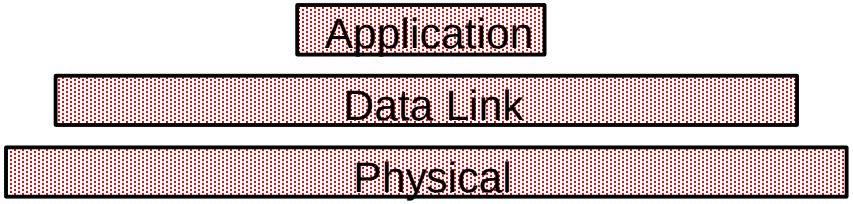
The IEC 870-5 Standard:

- Specifically designed for telecontrol applications
- Uses the 3-layer EPA model based on the 7-layer OSI model
- Has short reaction times when used in noisy environments at low communication speeds



Seven Layer vs. Three Layer:

- Only the Application, Datalink and Physical Layers are used
- Reduces communication overhead providing a more efficient protocol for low-to-medium speed media



Section 2 – DNP vs. 870-5 – cont.

Application Layer

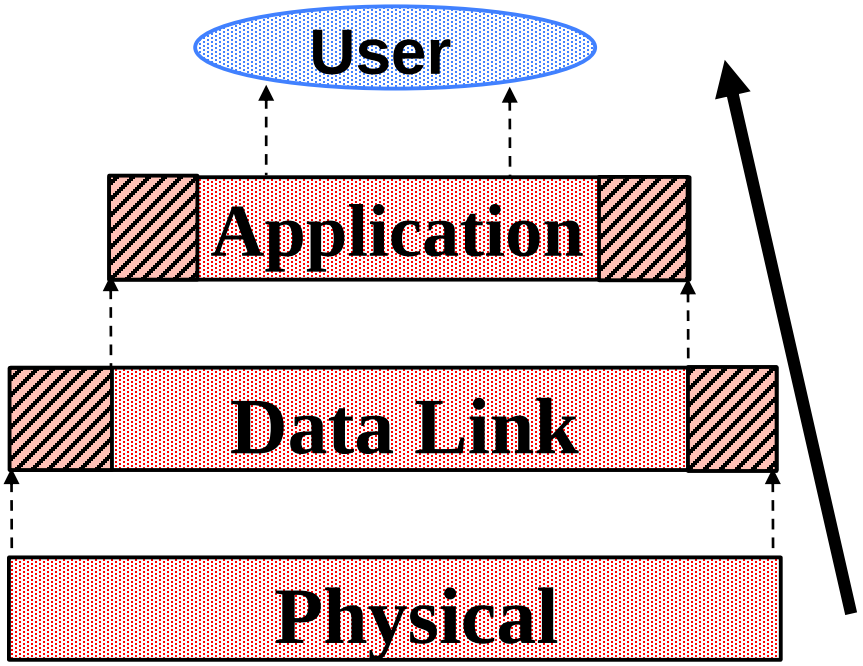
- Determines the format and characteristics of the data

Datalink Layer

- Confirms data integrity and extracts user data from the frames

Physical Layer

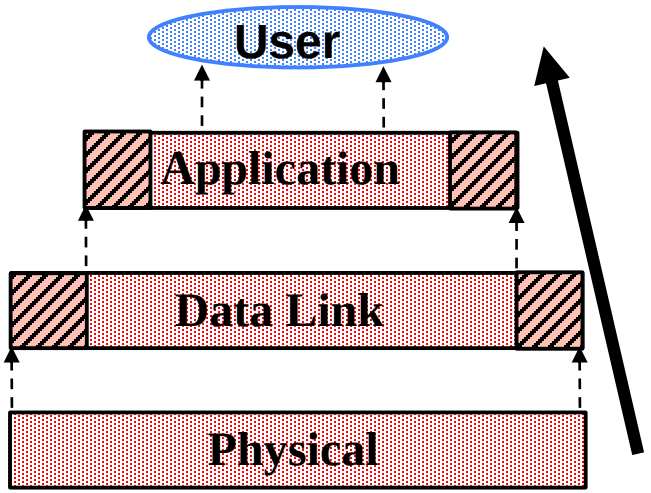
- Provides a method of transport for the data frames



Section 2 – DNP vs. 870-5 – cont.

Physical Layer:

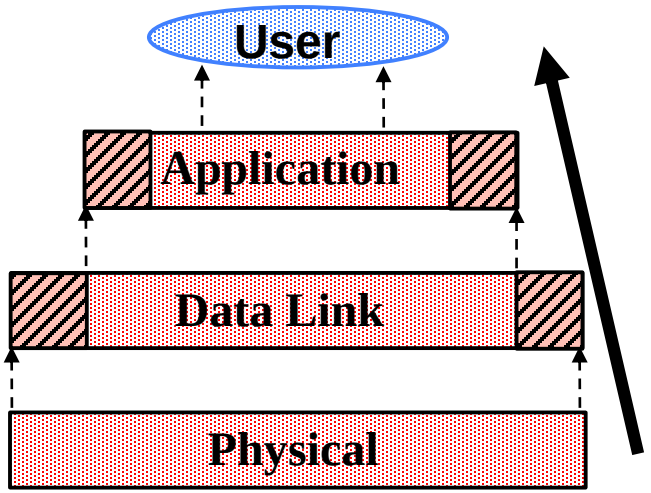
- Converts media signaling
- Provides isolation from media
- Synchronizes bits
- Adds / removes frame synch
- Detects media states
- Octet oriented (Byte)
- Uses FT3 Format
  - 1 Start
  - 1 Stop
  - 8 Data
  - No parity



Section 2 – DNP vs. 870-5 – cont.

Data Link Layer:

- Controls access to the physical layer
- Assembles/disassembles frames
- Sends/receives application data transparently
- Detects frame errors
- Provides error recovery procedures
- Recognizes addresses
- Handles different size frames
- Performs physical layer switching functions
  - start/stop switched network connections
  - switch over for failure recovery

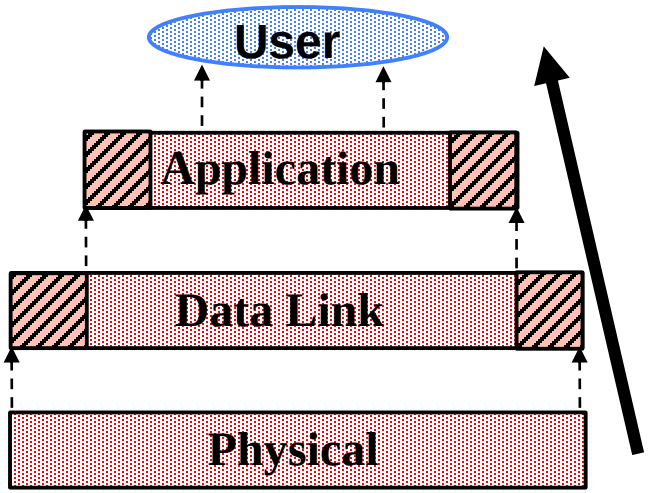


Section 2 – DNP vs. 870-5 – cont.

There are 3 service classes:

- Service Class S1 - SEND/NO REPLY
  - transmit message; neither ack nor answer is required
  - used for simplex (one-way) data transmission
  - data with errors detected is discarded

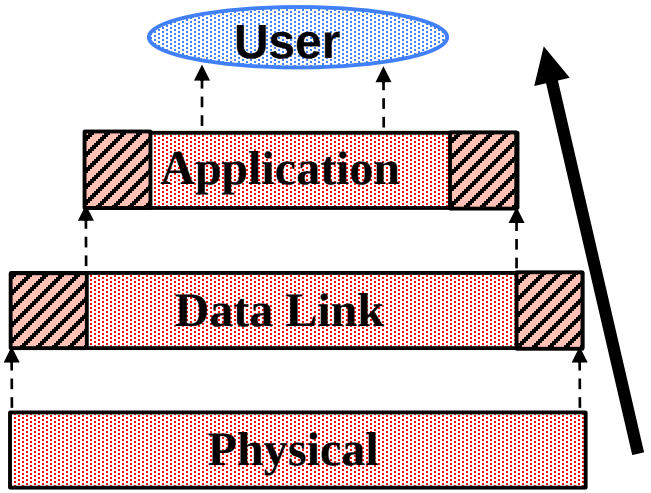
- Service Class S2 - SEND/CONFIRM
  - transmit message; acknowledgment requested
  - used for spontaneous data transmissions
  - good data is acknowledged
  - if receiver is unavailable, a NAK will be returned
  - data with errors detected is discarded



Section 2 – DNP vs. 870-5 – cont.

Three Service Classes – cont'd

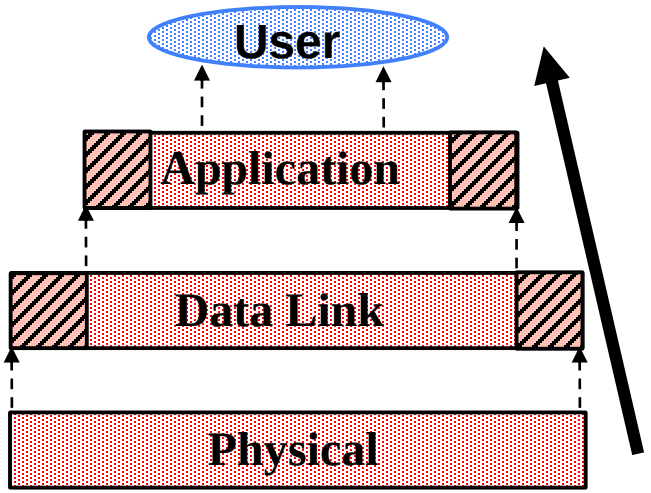
- Service Class S3 - REQUEST/RESPOND
  - used for "read" operations
  - receiving station responds with requested data if available
  - if no data available, a NAK will be sent
  - requests with errors detected are discarded



Section 2 – DNP vs. 870-5 – cont.

3 main types of Frame Formats:

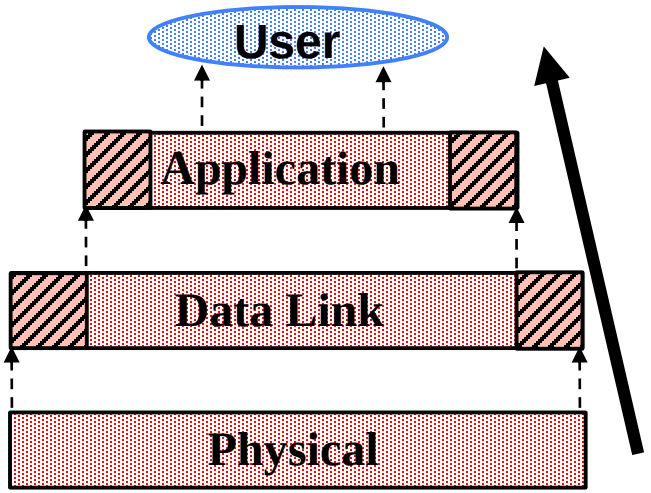
- FT1 - broken into two sub types: FT1.1 and FT1.2
  - adds a start, stop and parity bit to each octet, including the header octets
  - FT1.2 includes a single 8 bit CRC
- FT2
  - frame begins with a fixed length header starting with one of two 8 bit start characters, sequence of data blocks containing up to 16 data octets
  - each data block, including header has 8 bit CRCs



Section 2 – DNP vs. 870-5 – cont.

3 main types of Frame Formats - Cont'd

- FT3
  - each frame begins with a fixed length header starting with one of two 16 bit start characters, at the beginning of a sequence of data blocks containing up to 16 data octets
  - each data block, including header, has 16 bit CRCs



Section 2 – DNP vs. 870-5 – cont.

Un-balanced mode:

- Typically used for Master-Slave systems, i.e. polling (multi-drop) environments:
  - the Master (always primary) generates <requests> and receives <confirmations>
  - the Slave (always secondary) receives <indications> and responds with <responses>
  - requests which can be sent to the Data Link (from the Application Layer):
    - open a communication channel
    - close a communication channel
    - send a message
    - receive a message, a transmit status indication or a time-out indication
    - perform a control request

Section 2 – DNP vs. 870-5 – cont.

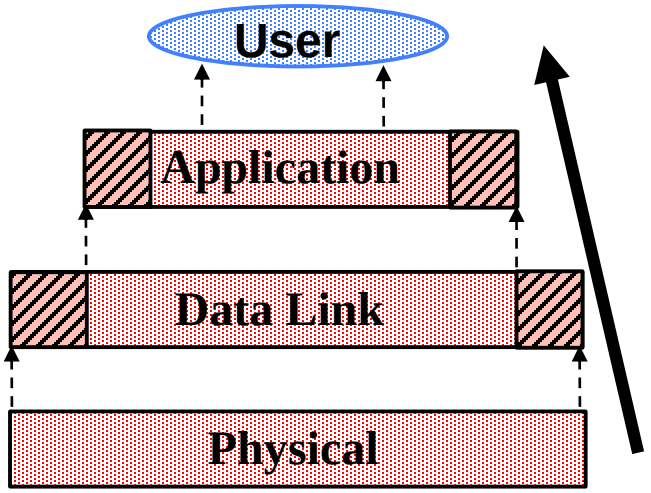
Balanced Mode:

- Used by combined systems, i.e.., where either station may initiate data transfers:
  - is best suited for point-to-point, full duplex systems
  - the Master/Slave relationship is relatively unimportant, as either station can assume the primary role at any time.
  - the station which initiates the transfer (primary) generates <requests> and receives <confirmations>
  - the station which responds (secondary) receives <indications> and responds with <responses>

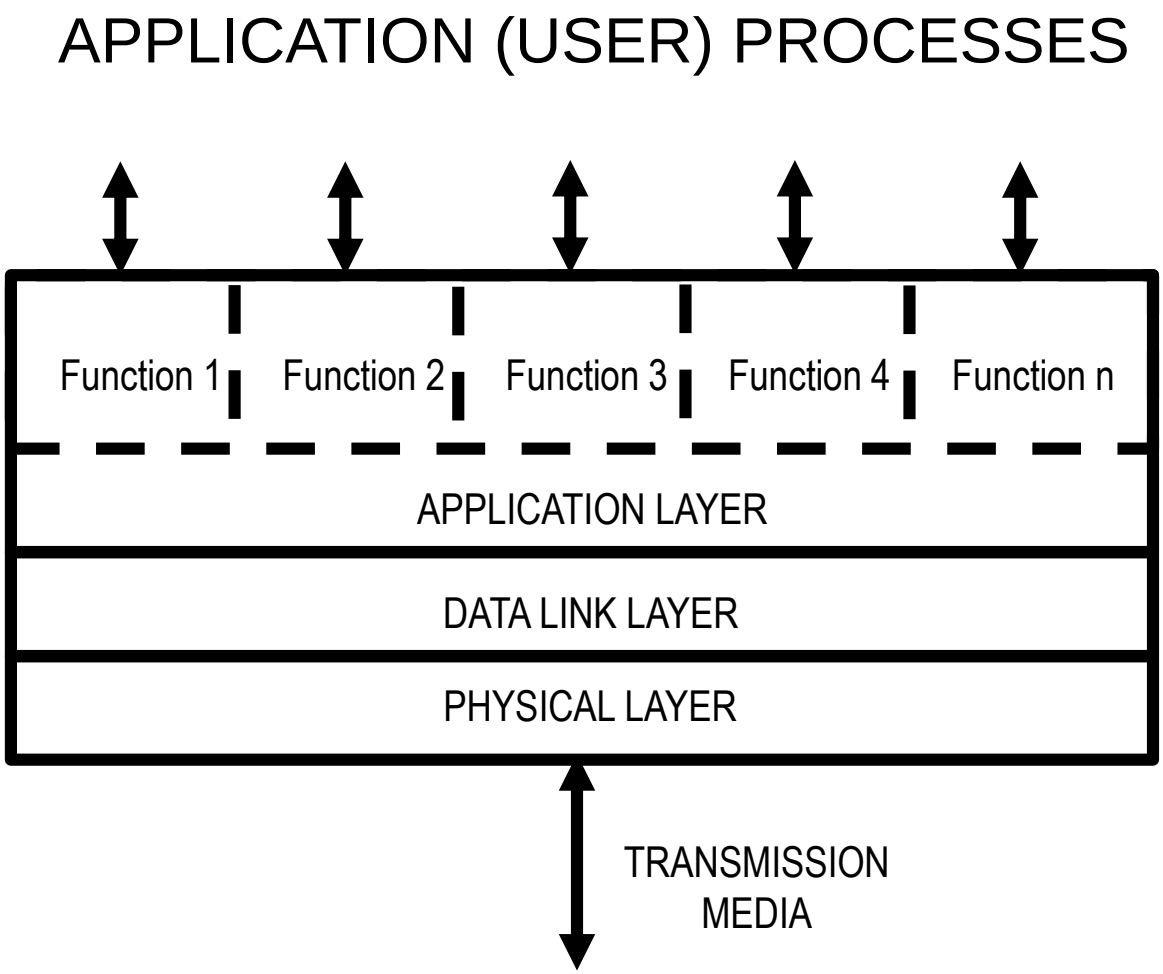
Section 2 – DNP vs. 870-5 – cont.

Application Layer:

- The user processes communicate with the application layer through Application Functions
- Several Application Functions may be communicating at one time
- Application Processes can have a Primary Application Function or a Secondary Application Function
- Application Functions use Service Primitives to pass requests and responses to/from the communication services (data link layer)



Section 2 – DNP vs. 870-5 – cont.

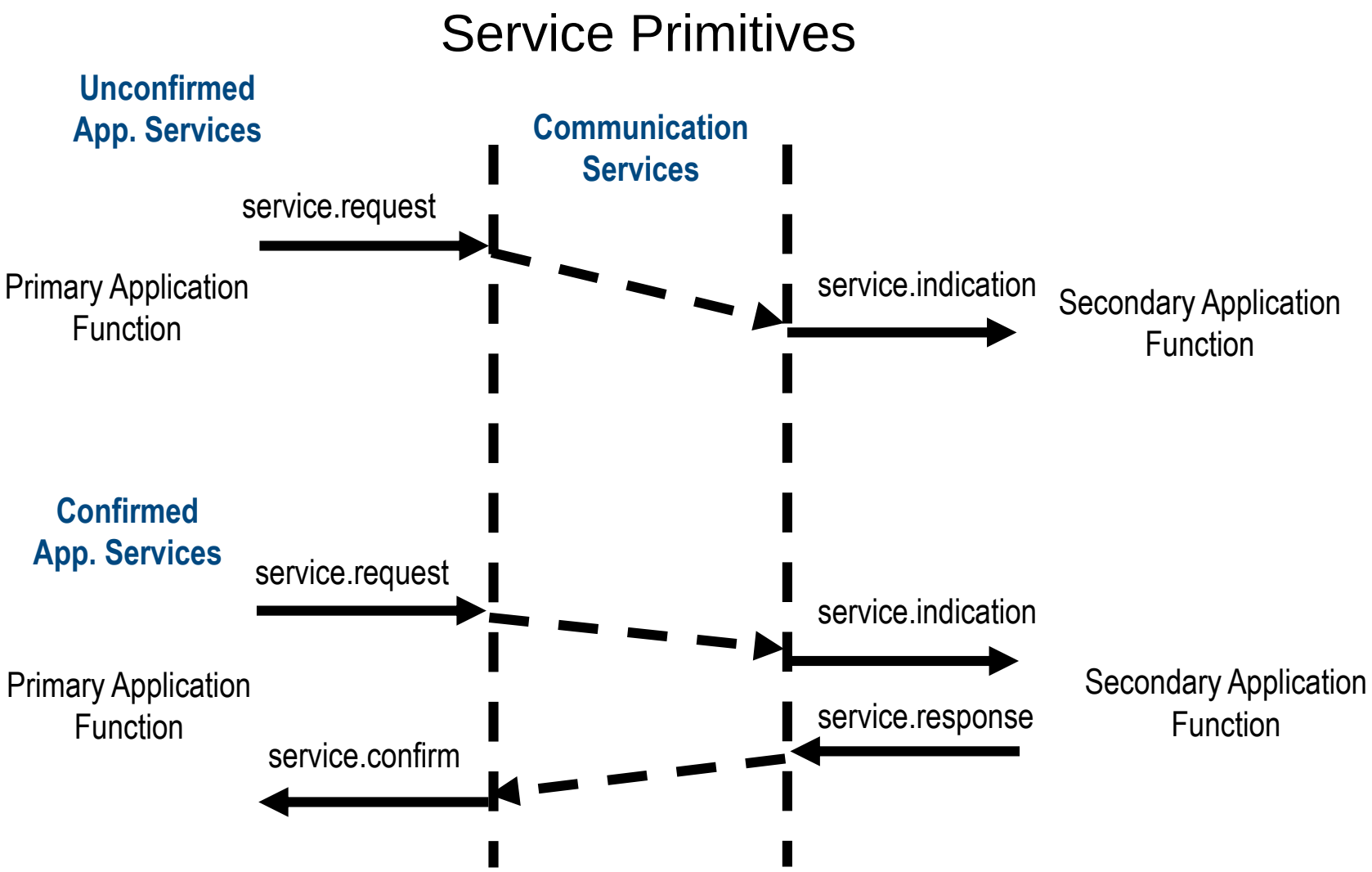


Section 2 – DNP vs. 870-5 – cont.

Four types of Service Primitives:

- <service>.request
  - the primary application function initiates a request via the communication services (data link layer)
- <service>.indication
  - comm. services deliver the indication to the secondary application function
- <service>.response
  - the secondary application function responds to an indication from the comm. services
- <service>.confirm
  - comm. services deliver the secondary's response to the primary application function

Section 2 – DNP vs. 870-5 – cont.

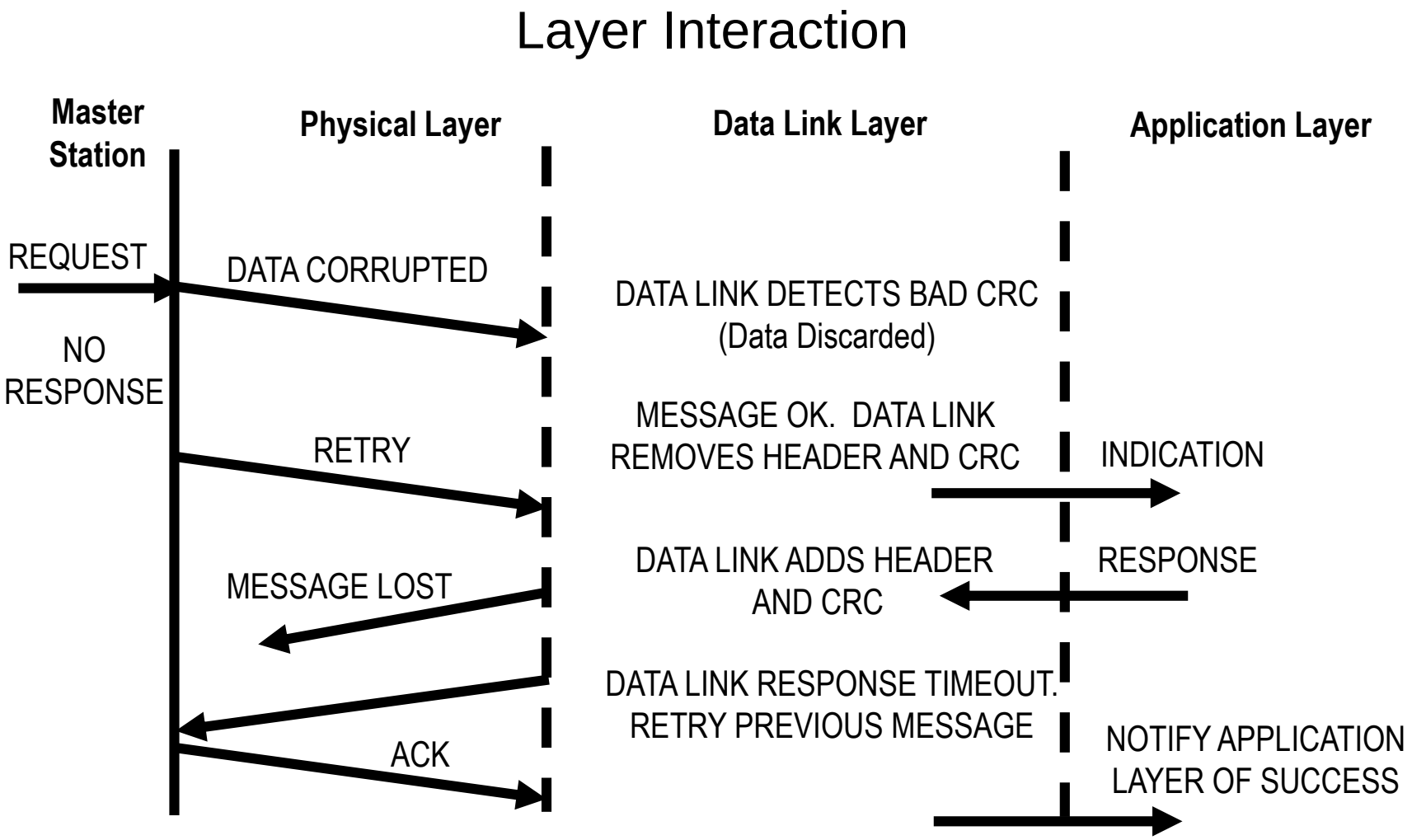


Section 2 – DNP vs. 870-5 – cont.

Application Functions defined in IEC 870-5-5

- Initialization
- Data Acquisition by Polling
- Cyclic Data Transmission
- Acquisition of Events
- General Interrogation
- Clock Synchronization and Delay Measurement
- Control Commands
- Transmission of Integrated Totals
- Parameter Loading (configuration file transfer)
- Specific messages defined in IEC 870-5-101

Section 2 – DNP vs. 870-5 – cont.



Section 2 – DNP vs. 870-5 – cont.

Physical layer:

- Same as IEC EPA, plus RS-485
- Real implementations often over radio systems

Data Link layer:

- Address field includes both source and destination
- Uses frame format FT3 ONLY
- Variable length frame type ONLY
- Balanced procedures ONLY
- Collision avoidance at physical layer for unsolicited
- Max. frame length 255 octets
- Various confirmation services
- Balanced procedures don't use REQUEST/RESPOND very much
- Asynchronous only

Section 2 – DNP vs. 870-5 – cont.

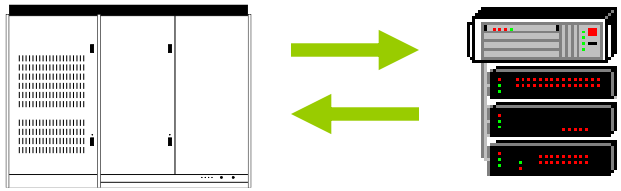
Application layer:

- Uses the same general concepts:
  - layered protocol
  - application primitives: request, respond, indicate, confirm
- Includes most of the same (SCADA) functions:
  - initialization
  - polling/interrogation
  - acquisition of events
  - clock synchronization
  - freezing data
  - control outputs
  - file/configuration transfer
- Actual message format is quite different

Section 2 – DNP vs. 870-5 – cont.

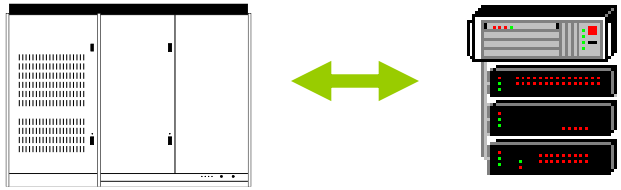
Collision Avoidance – DNP:

- DNP V3.00 uses IEC 870-5 Balanced Procedures
- Any device can theoretically transmit at any time



4-Wire

- 4-Wire Direct Connection
  - No Collisions are possible. One pair for TX and one for RX
- 2-Wire Direct Connection
  - Collisions are possible. One pair shared for TX and RX

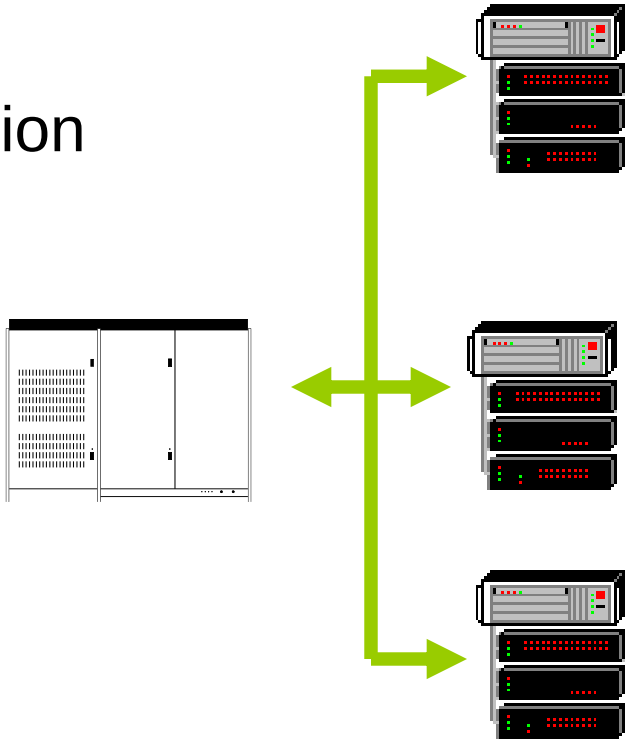


2-Wire

Section 2 – DNP vs. 870-5 – cont.

Collision Avoidance – DNP Implementation
Half Duplex 2-Wire

- All stations use RTS to turn on carrier of modem when they wish to transmit
- No station transmits while DCD is asserted
- Slaves wait variable time delays to transmit after DCD is clear
- Master has control of the circuit by always having the smallest delay

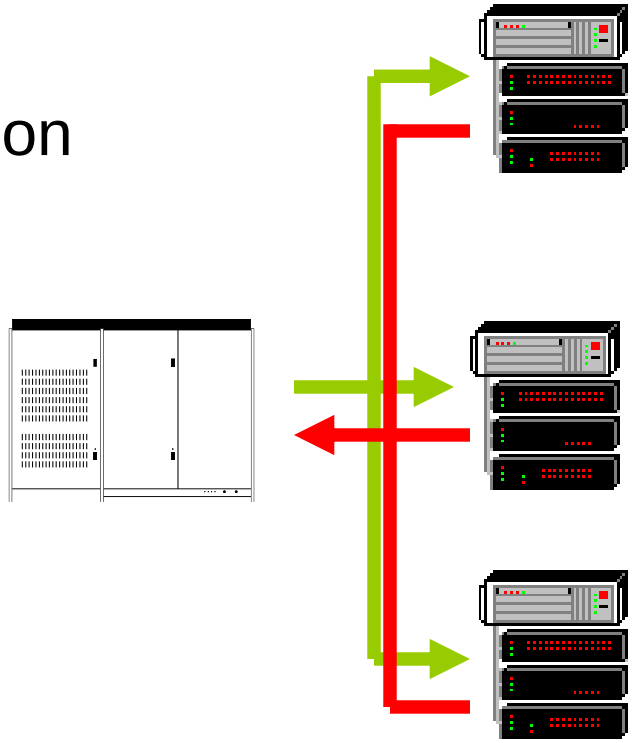


2-Wire Half Duplex
Multi Drop

Section 2 – DNP vs. 870-5 – cont.

Collision Avoidance – DNP Implementation
Half Duplex 4-Wire

- This format does not support collision avoidance because slaves cannot see each other's carriers.
- Collision Avoidance Algorithm will not work
- Users have been able to make systems work by using extended time delays to reduce probability of collision

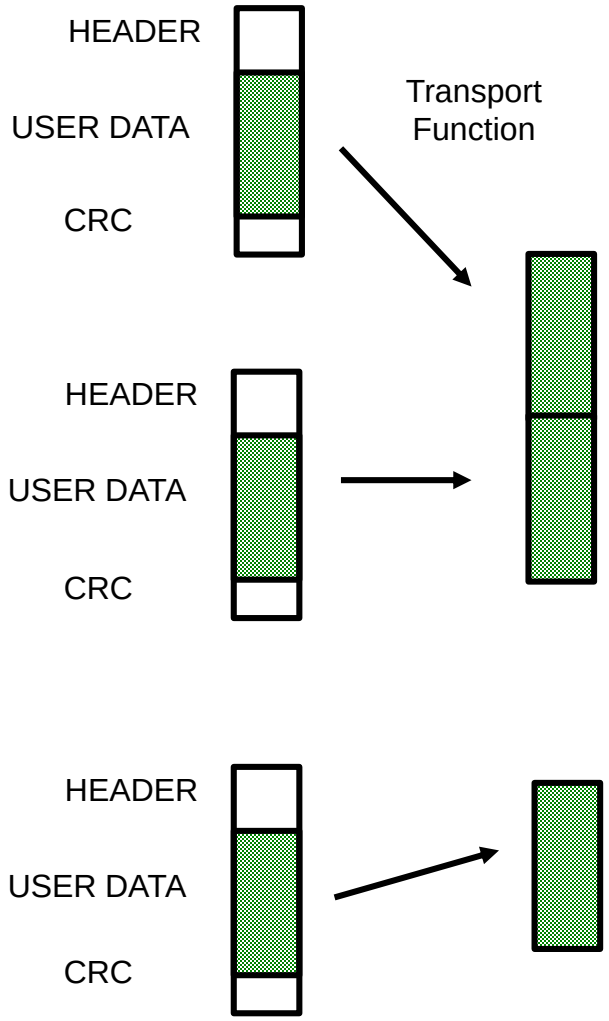


2-Wire Half Duplex
Multi Drop

Section 2 – DNP vs. 870-5 – cont.

Transport Functions – DNP Implementation

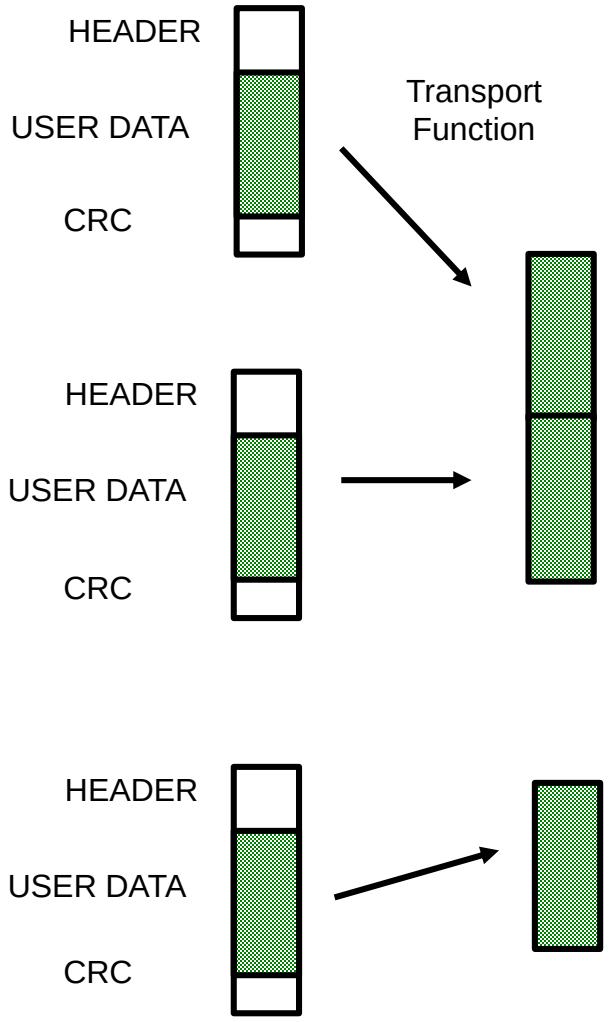
- When Application Messages are too large for a single frame a Pseudo Transport Layer provides segmentation services
- One Transport Service Data Unit (TSDU) is broken into several sequenced Transport Protocol Data Units (TPDUs or segments), before it is sent to the Data Link Layer for transmission
- Multiple are TPDUs from the data link and reassembled into one TSDU
- A one-byte header is added between application and data link headers to do this.



Section 2 – DNP vs. 870-5 – cont.

“Real” Transport Layer vs. DNP:

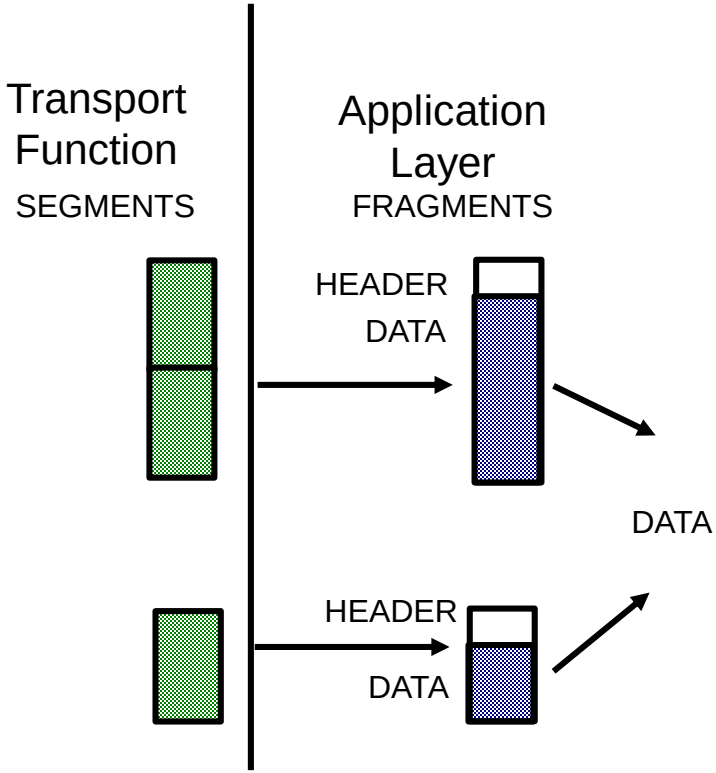
- A "real" transport layer:
  - is terminated only at each end of the message path
  - provides its own reliability/confirmation
  - has its own transport layer addresses
- The DNP Transport Function:
  - is terminated at the end of each data link
  - uses the data link layer addresses and reliability
  - however, not data link layer - first byte of user data



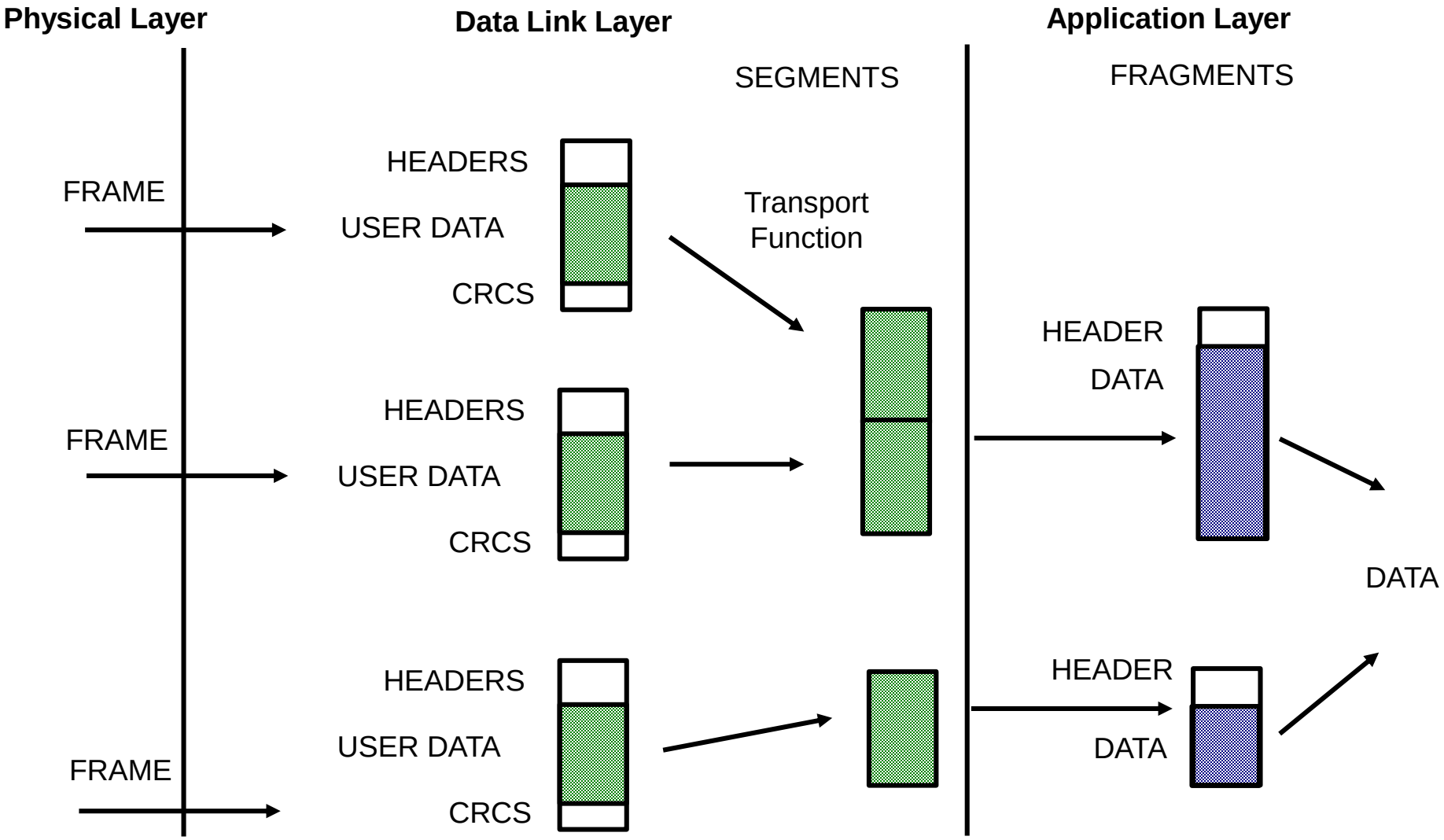
Section 2 – DNP vs. 870-5 – cont.

Fragmentation

- Performed at Application Layer
- Allows multiple responses to a single request
- Sets upper limit on size of buffer needed on receive side
- Data can't be broken just anywhere - each fragment must be separately parseable
- Performed much more rarely
- Typical limit is 2048 bytes



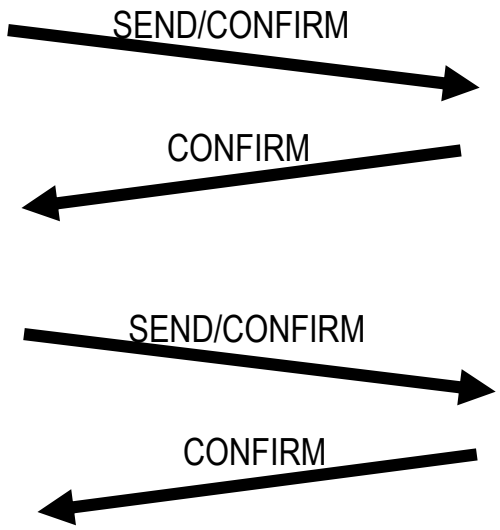
Section 2 – DNP vs. 870-5 – cont.



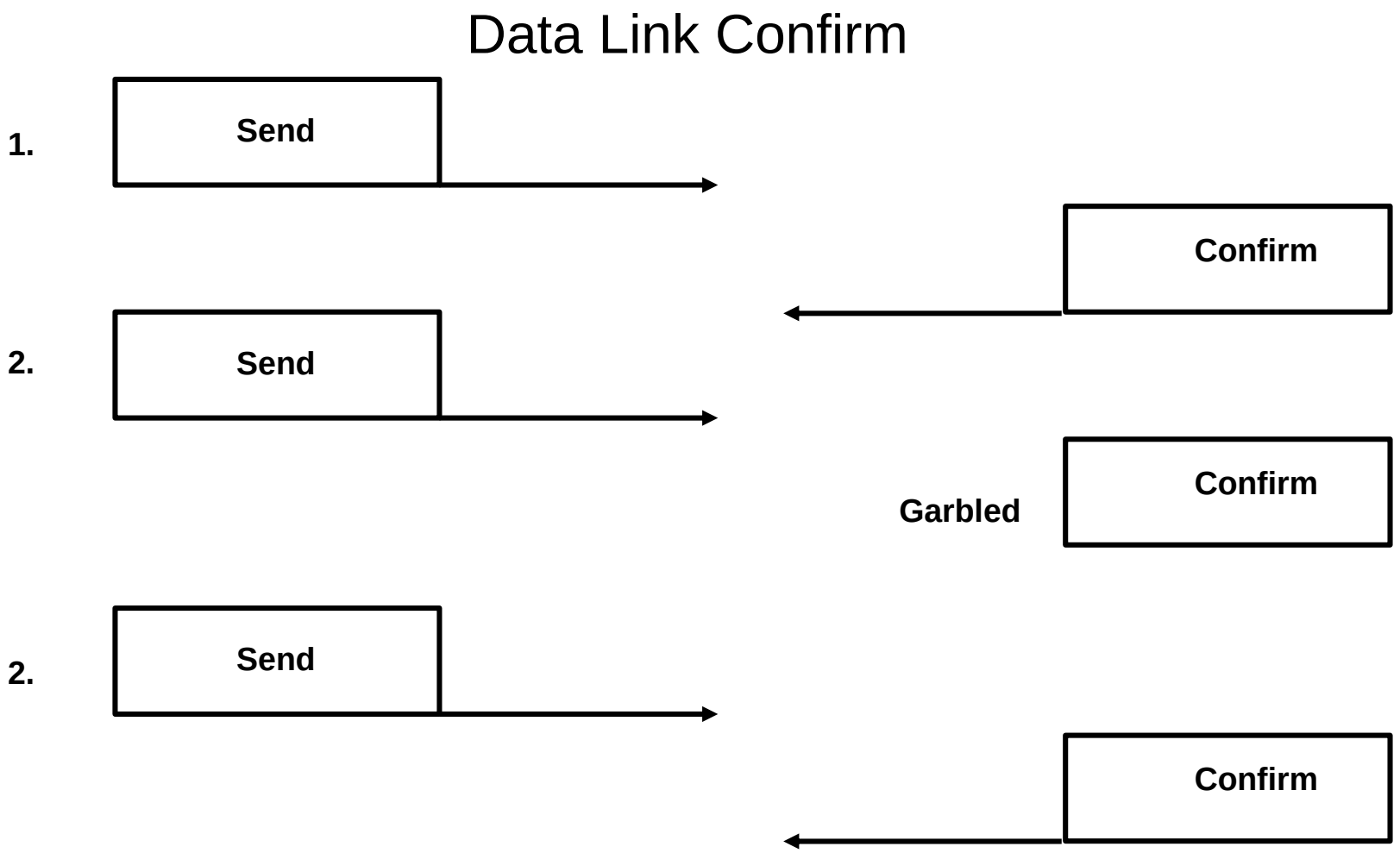
Section 2 – DNP vs. 870-5 – cont.

Confirmation Services – DNP Implementation:

- Data Link Confirmation:
  - confirms frame transmission
  - used to confirm that a single frame was received
  - used to provide reliability on noisy links
  - sending data link layer may choose to retry
  - single-bit sequence number ensures no duplicates
  - retries are efficient because frames are small



Section 2 – DNP vs. 870-5 – cont.

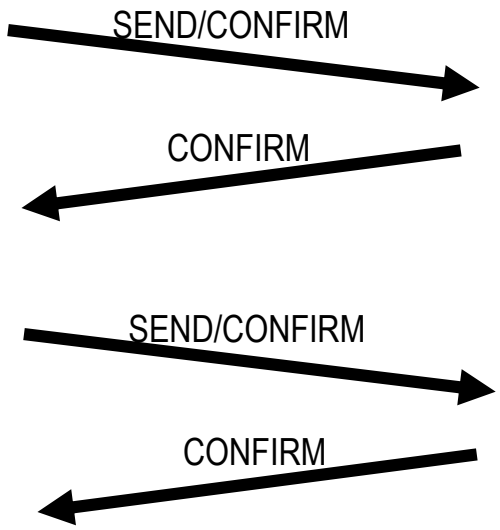


Send two frames. The second frame confirmation is lost.

Section 2 – DNP vs. 870-5 – cont.

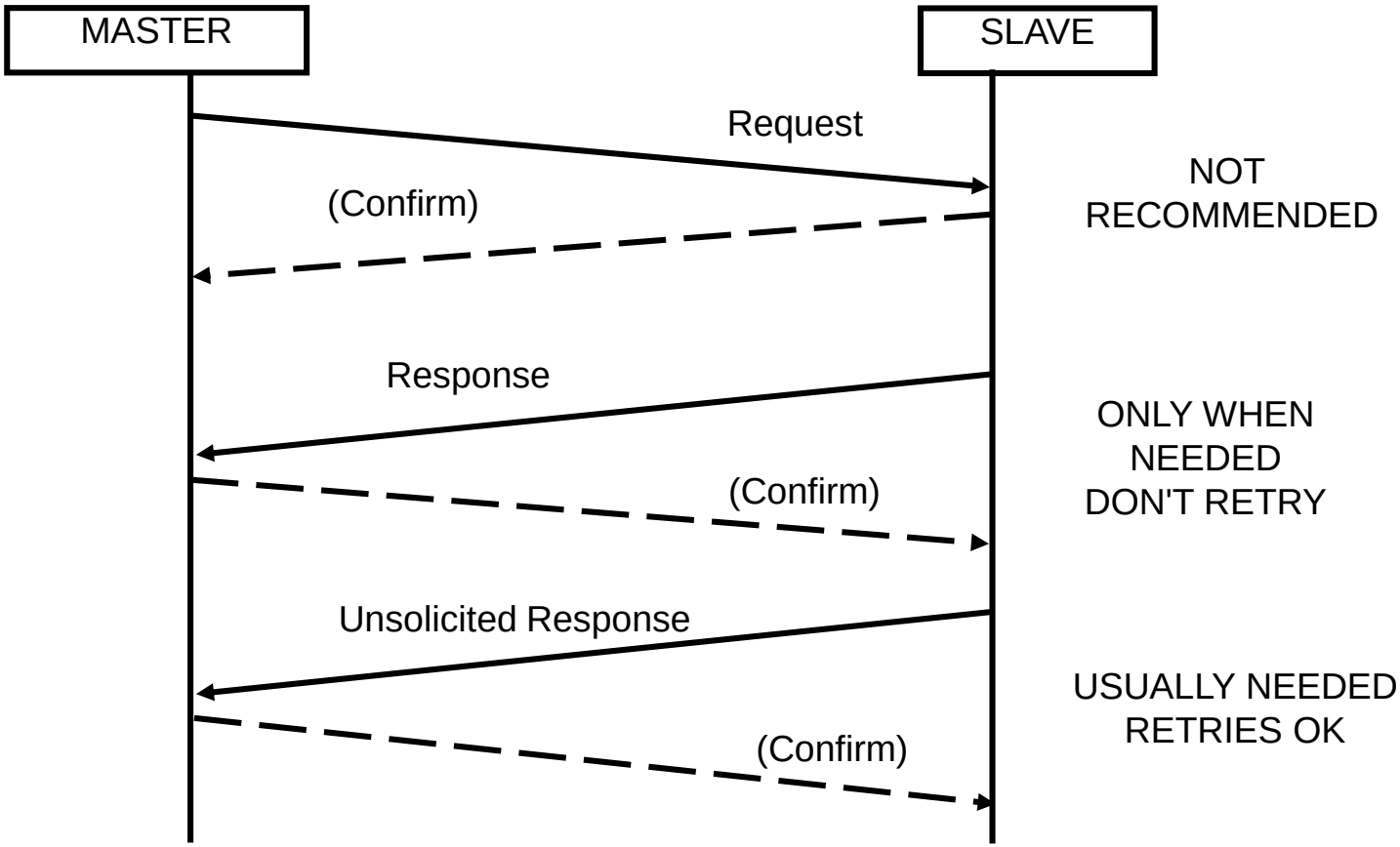
Confirmation Services – DNP Implementation:

- Application Layer Confirmation:
  - confirms message fragment transmission
  - application may or may not support retries
  - each frame of each fragment may also be retried several times by the Data Link layer!
  - application confirms recommended only for
    - multi-fragment responses
    - fragments containing event data
    - unsolicited responses



Section 2 – DNP vs. 870-5 – cont.

Application Layer Confirm



Section 2 – DNP vs. 870-5 – cont. Review Questions

1. How many layers does DNP use? Describe them.
2. What are the 4 Application Layer Primitives used in DNP? Describe them.
3. Describe Collision Avoidance. What formats support it? What formats do not?
4. What is the difference between Fragmentation and Segmentation?
5. What is the difference between Data Link Confirms and Application Layer Confirms? When should each be used?

3 Using DNP 3.0

Section 3 – Using DNP 3.0

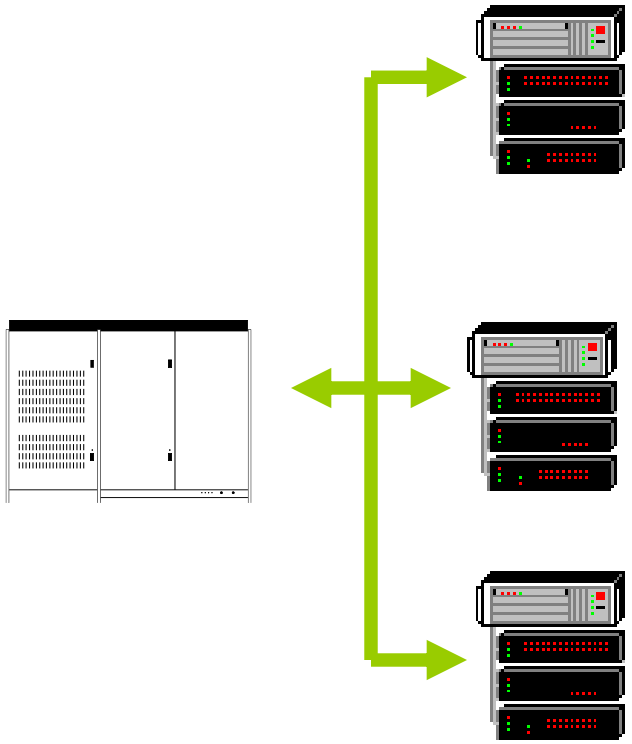
Objectives:

1. Use DNP 3.0.
2. List the DNP Subsets.
3. Describe the GE implementation of DNP.
4. Explain how to configure a GE Automation Product to communicate using DNP.

Section 3 – Using DNP 3.0 – cont.

Methods of retrieving Data:

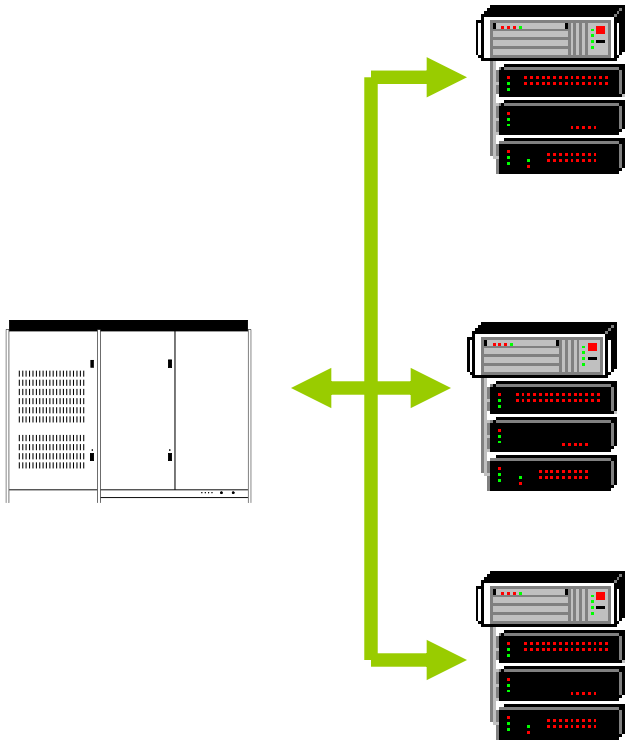
- Poll for Classes of Data
  - Class 0 non event, static data
  - Class 1, 2, 3 event data
- Poll for any variation of an object type
- Poll for specific variations or ranges
- Unsolicited responses
- File transfer
- Private registration objects



Section 3 – Using DNP 3.0 – cont.

Communication Techniques:

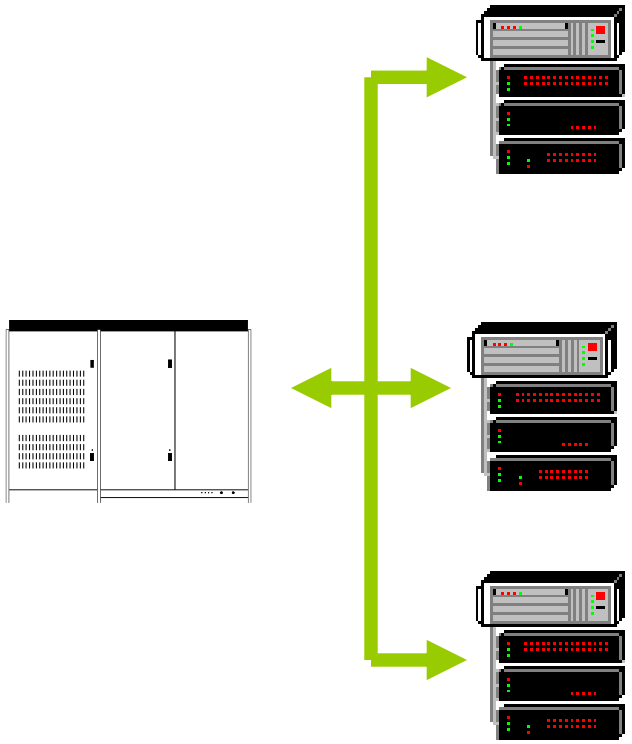
- Quiescent Operation:
  - all unsolicited responses, Master never polls
  - best with many devices with small point counts
- Unsolicited Report-by-Exception:
  - unsolicited responses for events
  - Master occasionally checks integrity of database
- Polled Report-by-Exception
  - event polls interspersed with integrity polls
  - no unsolicited operation
- Polled Static Operation
  - only integrity polls performed.
  - no event polling



Section 3 – Using DNP 3.0 – cont.

Output Options:

- Select/Operate or Direct Operate Function Codes
- Direct Operate - No Ack for broadcasting
- Multiple Outputs in the same message
- Control Relay Output Block options:
  - Trip
  - Close
  - Latch On/Off
  - Pulse On/Off
  - Pattern Masks
  - Repeat Counts
  - Time Intervals
- Analog Output Blocks (setpoints)



Section 3 – Using DNP 3.0 – cont.

Subset Definitions:

- DNP is a large protocol
- No vendor currently implements it all
- Customers need a simple way to specify compatibility
- Vendors need guidelines on what to implement

General Rules:

- Master cannot transmit any requests not defined for the level - may accept additional responses.
- Slave cannot transmit any responses not defined for the level - may accept additional requests.
- Vendor must record all differences in a Device Profile Document.
- Any differences beyond these rules must be configurable.

Section 3 – Using DNP 3.0 – cont.

DNP-L1:

- Suggested for the simplest Master Station and IED applications
- Master acquires data from Slave by either:
  - Class Data Polls
  - polls for variation 0 of output objects
  - unsolicited response from Slave
- Master will not request any specific data objects
- Includes most basic data types
- No frozen objects
- Binary inputs only time-stamped objects
- Time synch, delay measurement if needed
- Limited subset of qualifiers
- Cold Restart and Writes to Restart IIN

Section 3 – Using DNP 3.0 – cont.

DNP-L2:

- Restricted implementation for Master to smaller remote devices:
  - meters, reclosers, small RTUs
  - intended for similar applications as L1

- Has a few more features:
  - Freeze requests
  - Frozen counter objects (not Analogs or Binary)
  - Reads for variation 0 of most objects
  - Reads for all variations of Binary Inputs
  - Unsolicited responses with static data

- This is the level recommended by IEEE RP-1379.

Section 3 – Using DNP 3.0 – cont.

DNP-L3:

- Suggested for Master to medium sized Slave devices:
  - RTUs, concentrators
- Much larger range of Objects, Variations, Function and Qualifier codes
- Read Requests of specific objects and variations
- Enable/Disable Unsolicited Responses
- Assigning objects to classes dynamically

Section 3 – Using DNP 3.0 – cont.

Not Defined in Any Subset:

- Frozen Counters with Time of Freeze
- Counter and Analog Events with Time
- Frozen Analog Inputs and Events
- Storage Object
- Device Profile Object
- Private Registration Objects
- Application Identifier Objects
- File Identifier Objects
- Floating Point and BCD Objects
- Function Codes:
  - FREEZE WITH TIME (ACK & NO ACK)
  - START/STOP APPLICATION
  - SAVE CONFIGURATION
  - WARM RESTART
  - INIT DATA TO DEFAULTS

Section 3 – Using DNP 3.0 – cont.

GE Implementation – General:

- Any DNP implementation will support a subset of the total protocol described in the DNP V3.0 Basic 4 Documentation set (977-0007)
- The applications included in the GE implementation are:
  - B021 – DNP DPA
  - B023 – DNP DCA
  - B015 – Bridge Manager
  - B013 – DNP Serial Data Link
  - B052 – DNP Ethernet Data Link

Section 3 – Using DNP 3.0 – cont.

B023 DNP DCA

- Communication between RTU and sub-remote or IED - The Data Collection Application:
  - allows the RTU to act as a "Master" to one or more remote devices
  - is a component of the Application Layer of the EPA Model

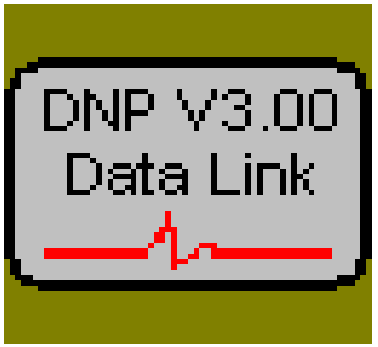
B023 DNP DPA

- Communication between RTU and Master Station – The Data Processing Application:
  - allows the RTU to act as a “Slave" to one or more master stations
  - is a component of the Application Layer of the EPA Model

Section 3 – Using DNP 3.0 – cont.

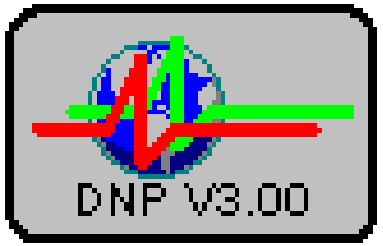
B013 DNP Serial Data Link

- Services for software applications to send and receive messages using DNP V3.0 protocols are provided by the Data Link Software
- Provides control of a serial physical layer



B052 DNP Ethernet Data Link

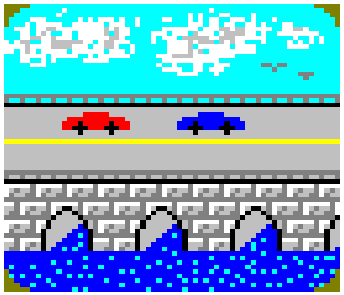
- Services for software applications to send and receive messages using DNP V3.0 protocols are provided by the Data Link Software
- Provides control of an Ethernet physical layer



Section 3 – Using DNP 3.0 – cont.

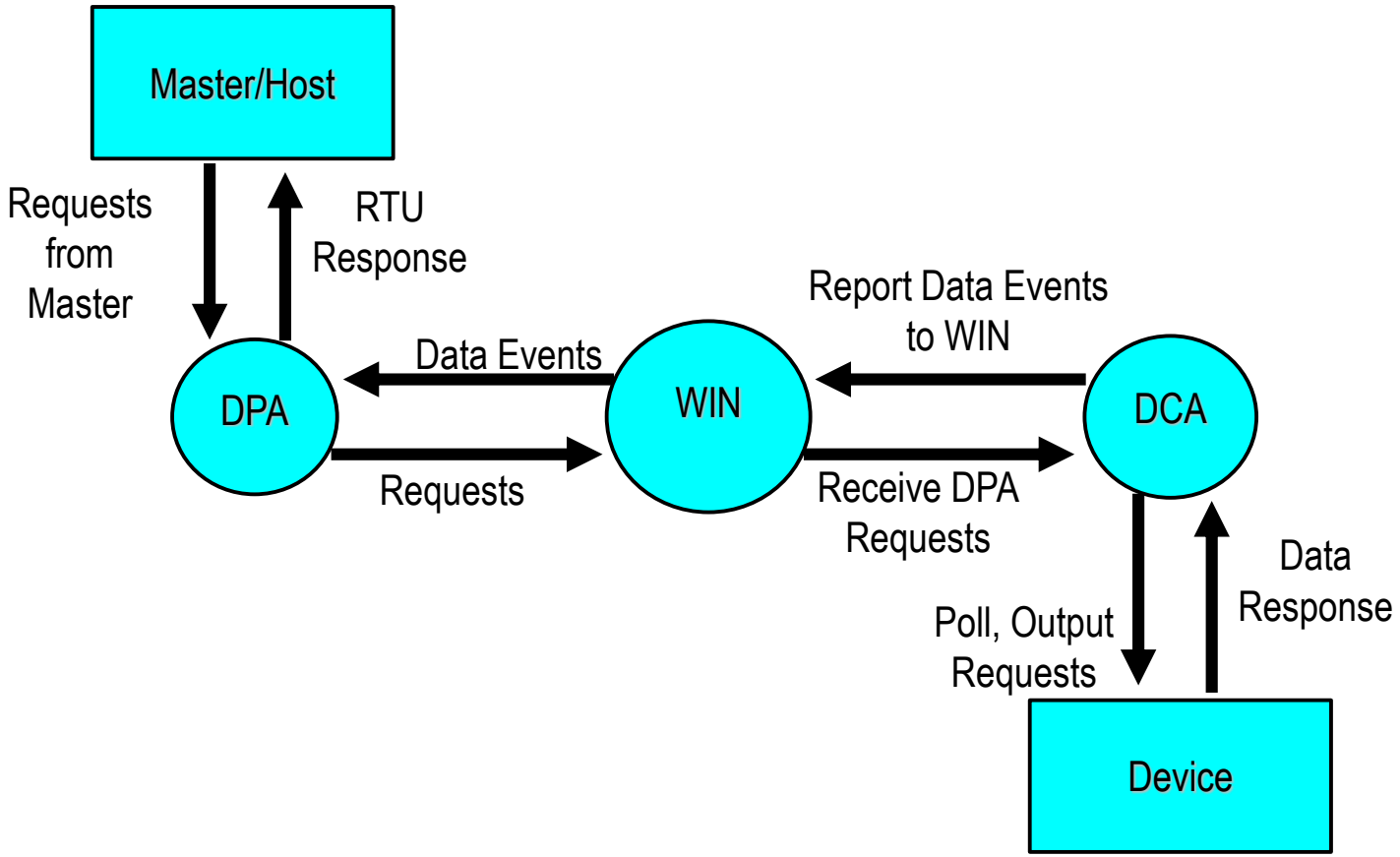
B015 Bridge Manager

- EPA Model does not provide a network layer for routing
- Bridge Manager provides this function without adding a header to the data
- Provides a connection between the DataLink and Application Layers

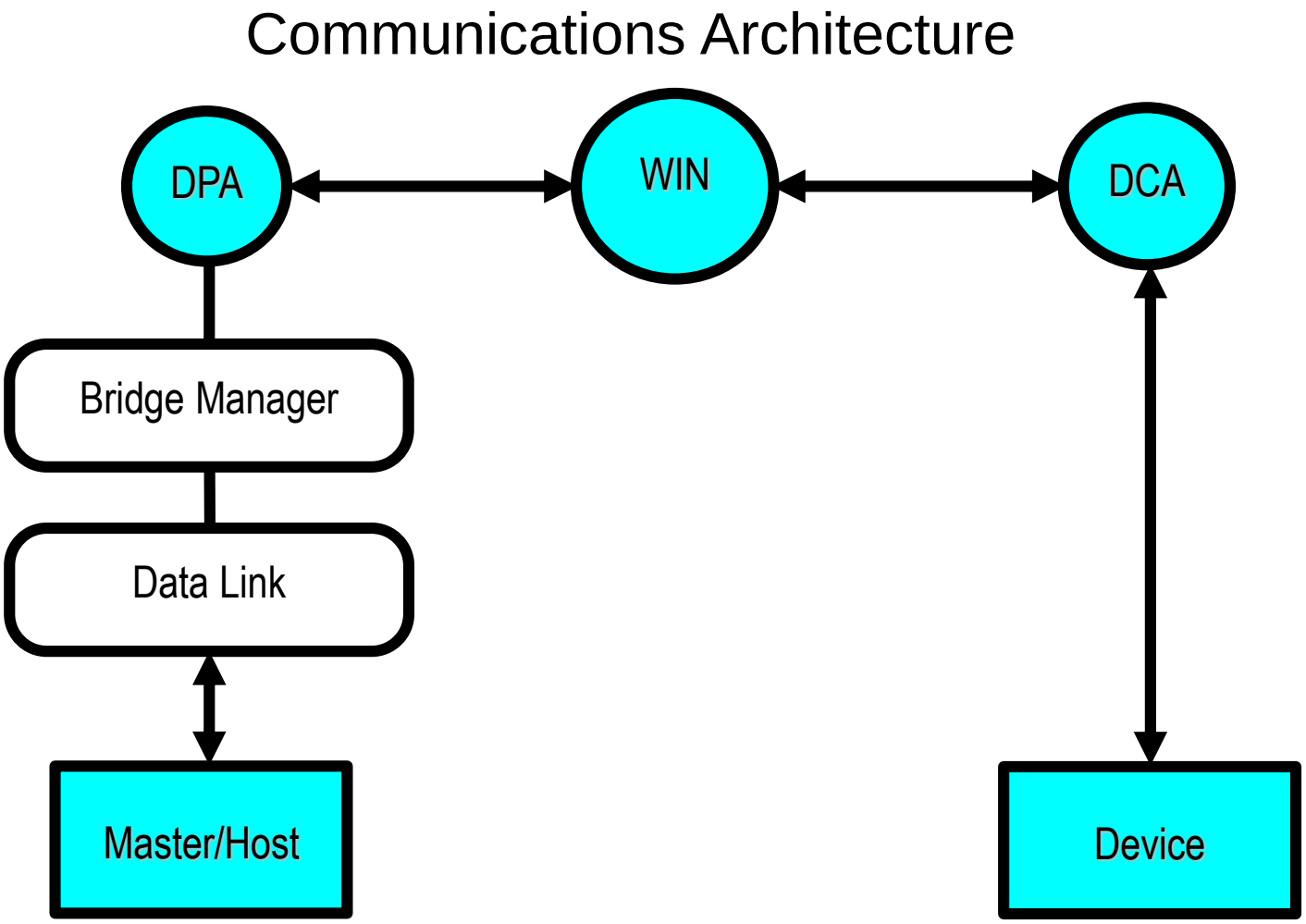


Section 3 – Using DNP 3.0 – cont.

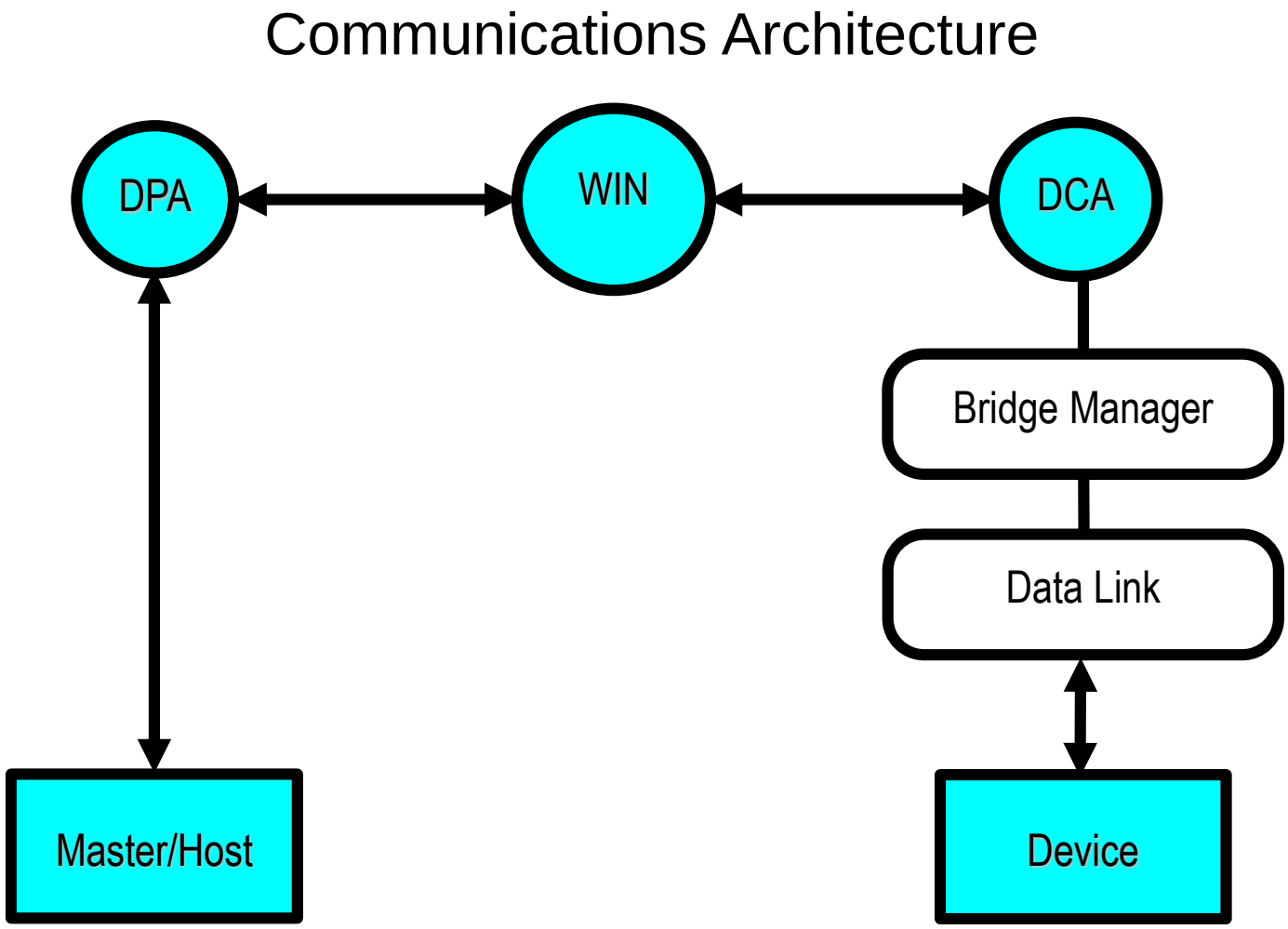
Communications Architecture



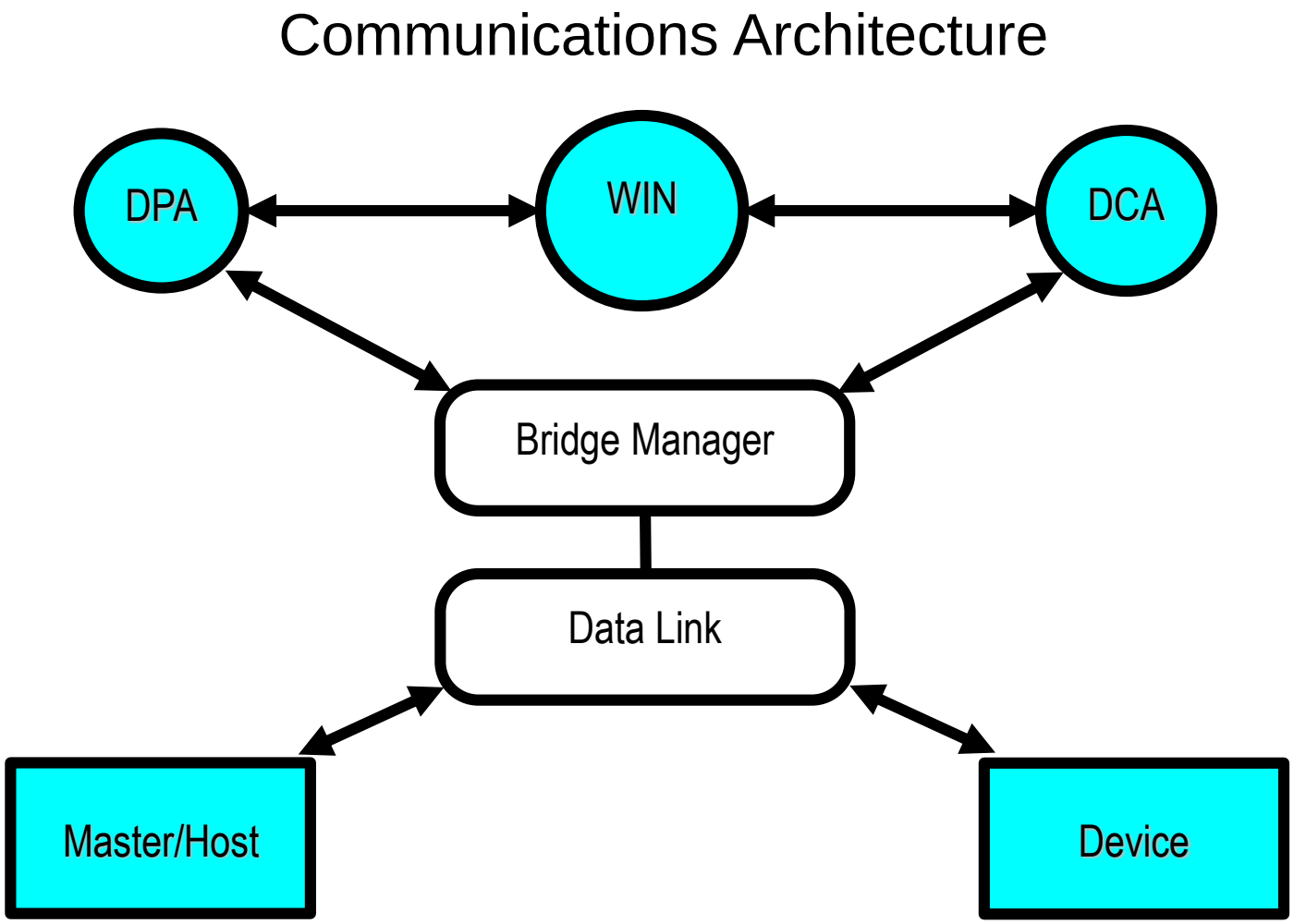
Section 3 – Using DNP 3.0 – cont.



Section 3 – Using DNP 3.0 – cont.



Section 3 – Using DNP 3.0 – cont.



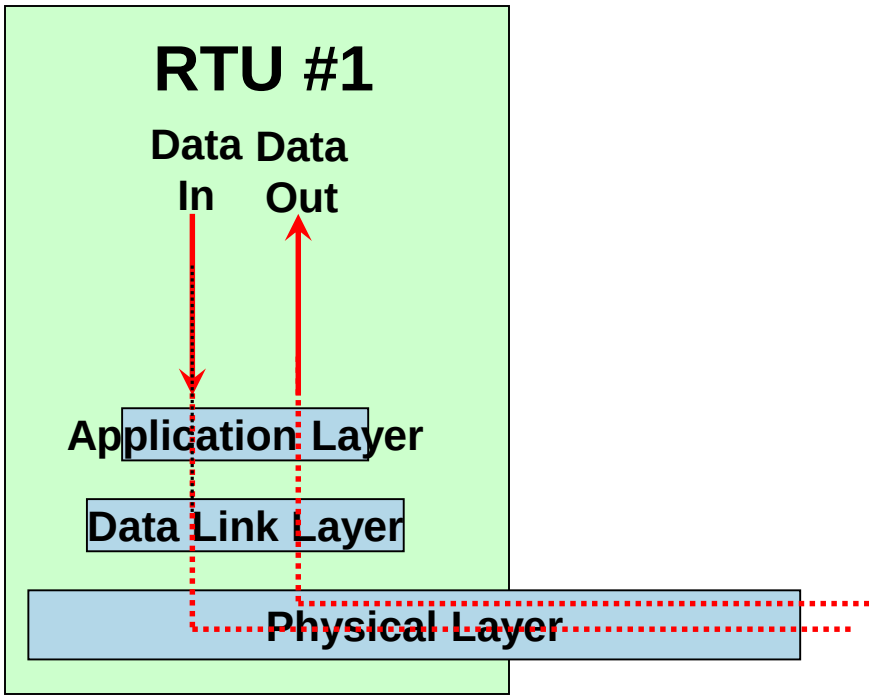
Section 3 – Using DNP 3.0 - Configuration

There are several applications that must be configured to use DNP:

- DNP DCA (B021) or DNP DPA (B023)
- Bridgeman (B015)
- DNP Serial Datalink (B013).

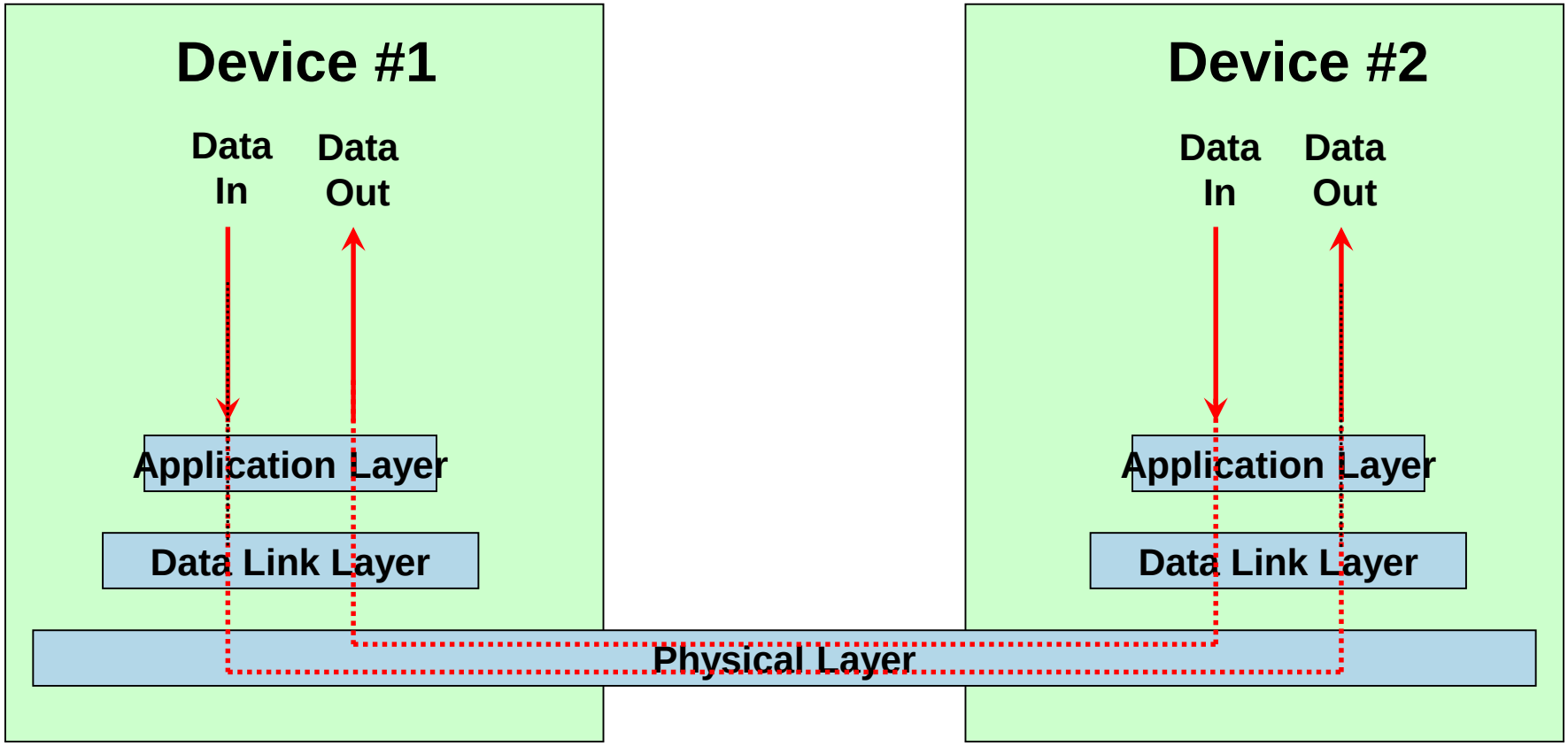
DNP operates using a 3 layer model called the Enhanced Performance Architecture:

- Application Layer (DNP DCA or DNP DPA)
- Datalink Layer (DNP Datalink)
- Physical Layer (physical media)
- Pseudo Network Layer (BridgeMan)



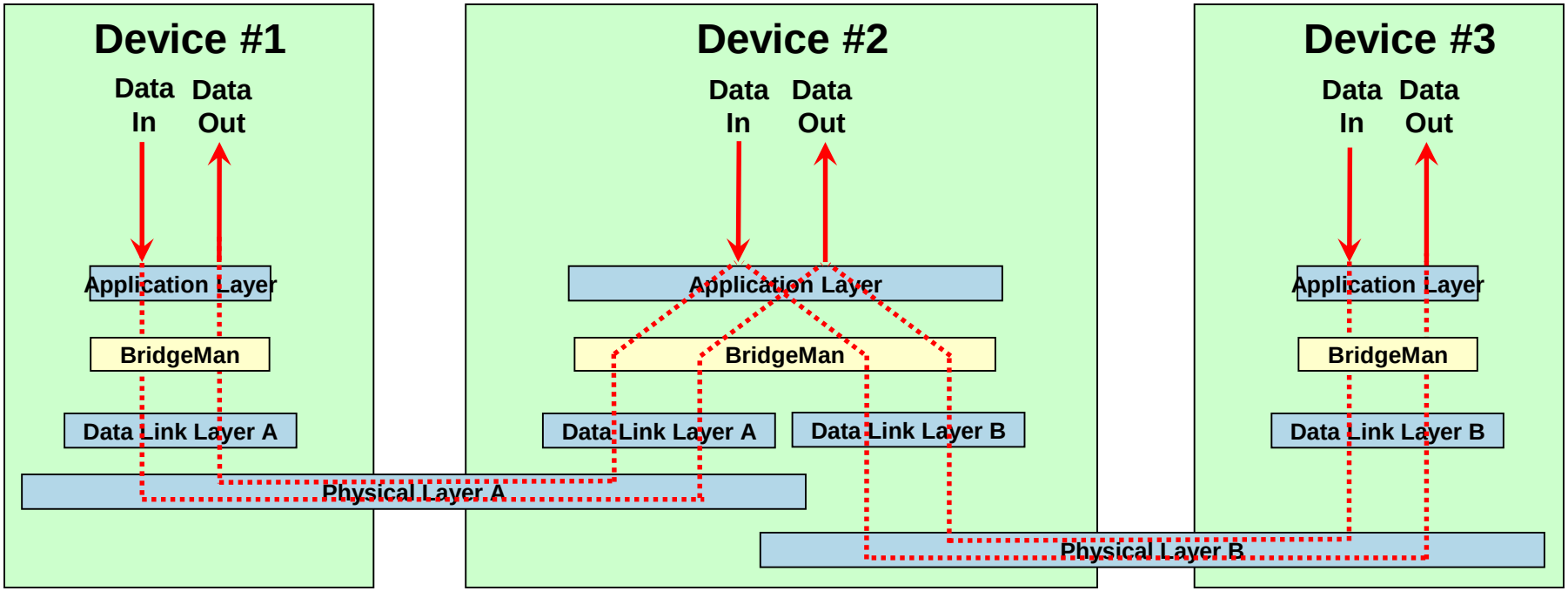
Section 3 – Using DNP 3.0 – Configuration – cont.

Device Communication

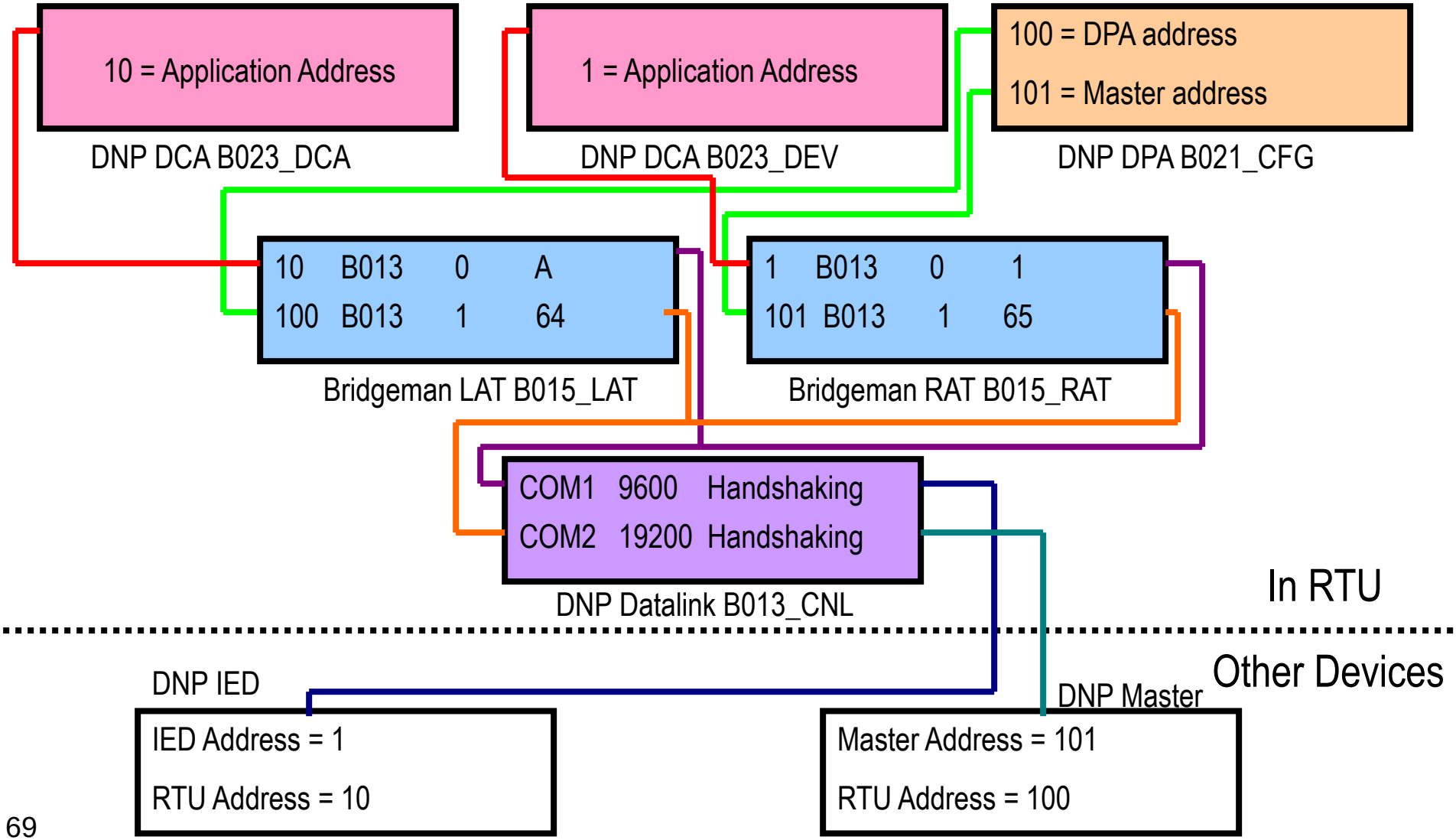


Section 3 – Using DNP 3.0 – Configuration – cont.

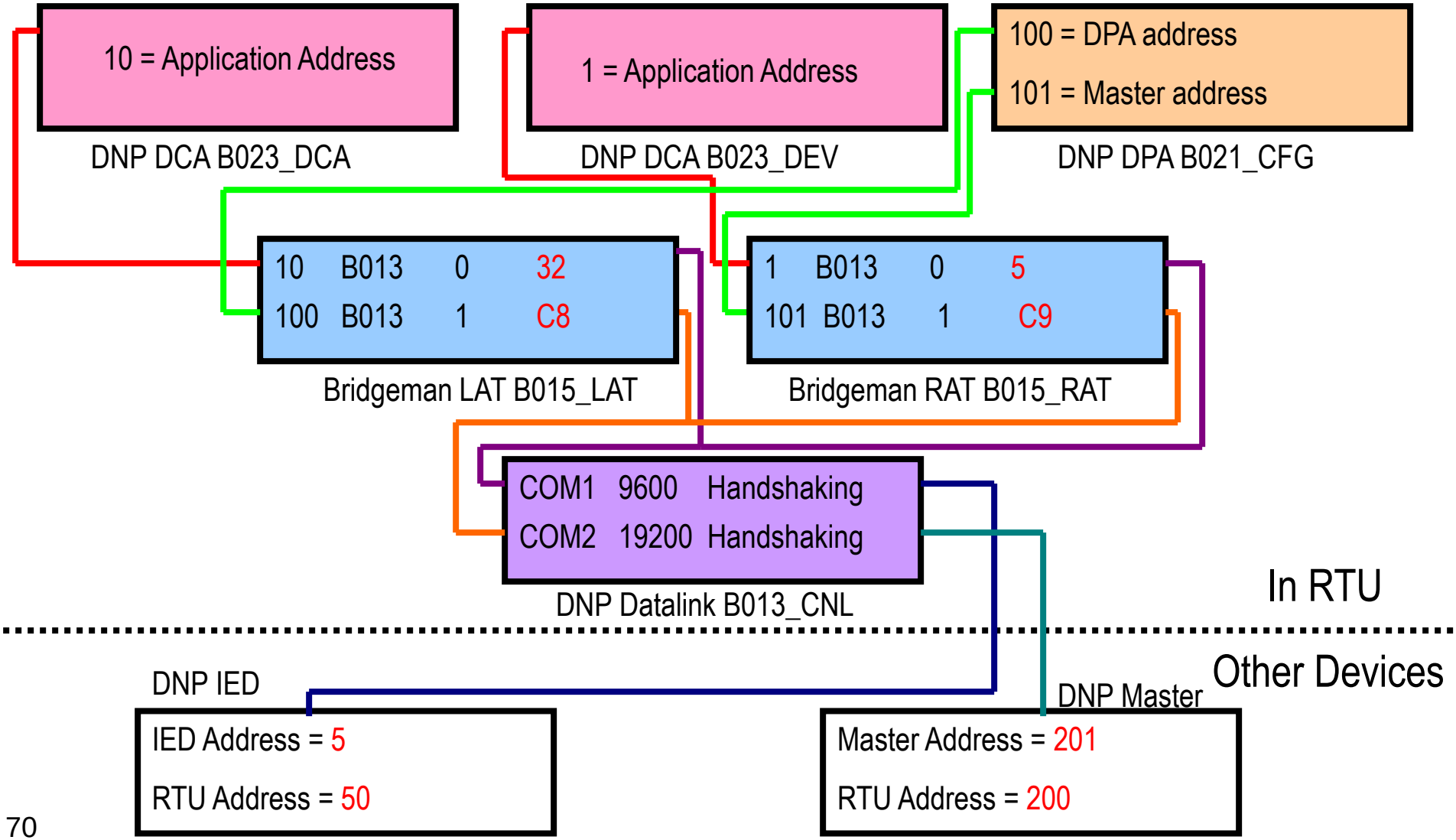
Device Communication
Multiple Data Links



Section 3 – Using DNP 3.0 - Configuration



Section 3 – Using DNP 3.0 - Configuration



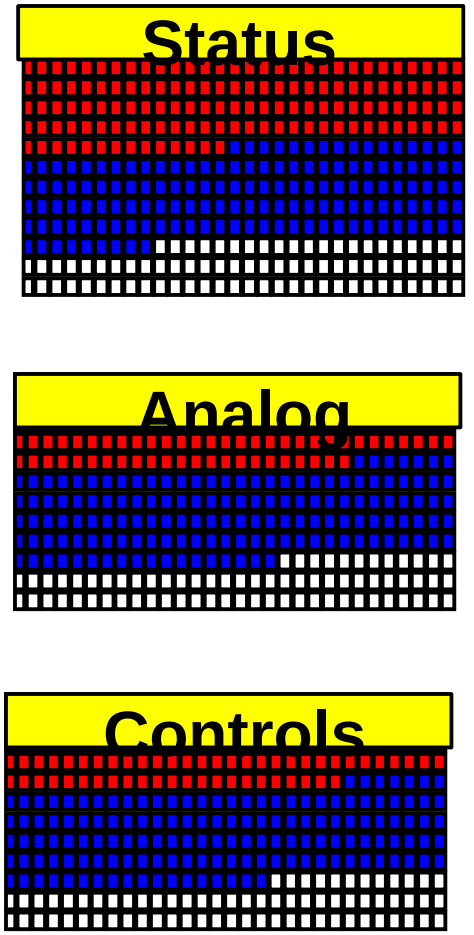
Section 3 – Using DNP 3.0 – Configuration – cont.

Once the applications are configured, you must reserve space in the database for the date generated by the DNP applications.

DNP DCA will require you to reserve space for both PSEUDO points and the data you are collecting for the IEDs.

Global pseudo points are associated with every device being polled by an instance of the DCA and only need to be configured once:

- 3 Global DI
- 9 Global DO



* Application Version Dependant. See Configuration Guide for Details.

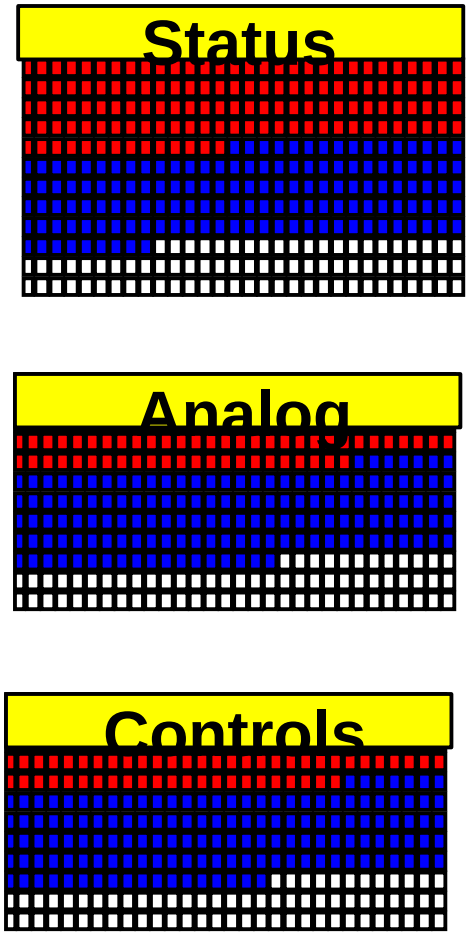
Section 3 – Using DNP 3.0 – Configuration – cont.

Device pseudo points are only associated with one device and need one set to be configured per device:

- 16 Per Device DI
- 10 Per Device DO
- 5 Per Device AI

The application data is stored in the following order:

- Global Pseudo Points
- Device 1 Real Data
- Device 1 Per Device Pseudo Points
- Device 2 Real Data
- Device 2 Per Device Pseudo Points

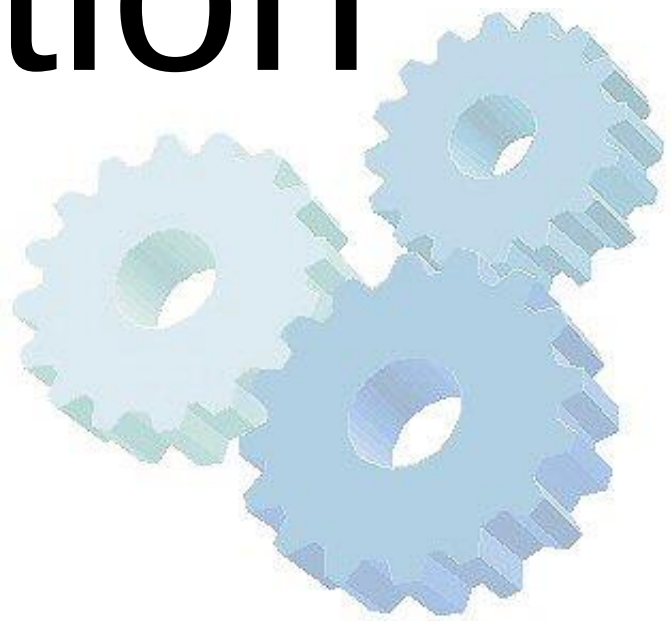


* Application Version Dependant. See Configuration Guide for Details.

Section 3 – Using DNP 3.0 Review Questions

1. Describe 5 methods for transferring data.
2. Describe the 4 communications techniques.
3. Outline the differences between the three DNP Subsets.
4. What are the five applications in the GE implementation of DNP and what are their functions?
5. Sketch and describe the communications architecture in the GE Implementation of DNP.

DNP Configuration



LAB 1

DNP Configuration

Objective:
The objective of this lab is to create and test an RTU configuration that will allow a GE RTU to communicate using DNP.

Introduction:
In this lab, the student will create and test a DNP Configuration. The following slides will outline the configuration requirements.
This configuration will be used in subsequent labs.

LAB 1

DNP Configuration

Meter Requirements

Configure the RTU to communicate with a meter using the following specifications:

- DNP address of the meter = 1
- DNP address of the RTU = 10
- 10 DI Points, 10 AI points, 10 DO Points
- Poll for events every 2 seconds and the database every minute
- Use Com3, 9600 baud, no flow control

LAB 1

DNP Configuration

Master Requirements:

Configure the RTU to communicate with a master using the following specifications:

- DNP address of the Master = 10
- DNP address of the RTU = 1
- 10 DI Points, 10 AI points, 10 DO Points
- All data Class 1, no unsolicited responses
- Use Com4, 9600 baud, no flow control

LAB 1

DNP Configuration

Connections and Testing:

Use a null modem cable to connect Com3 and Com4 of the RTU. Check to ensure communications is established with the following tests:

- Wesmaint Device Status Display
- All data is ONLINE in the Wesmaint Data Display
- SA COM3
- SA COM4

LAB 1

4 Message Structure

Section 4 – Message Structure

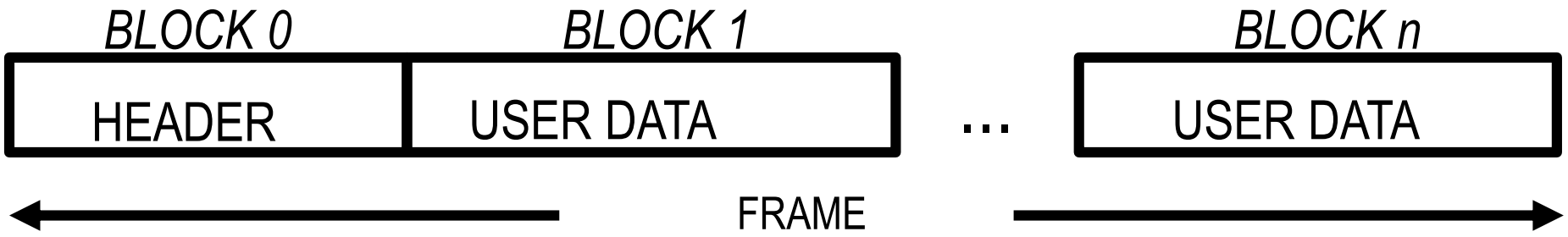
Objectives:

1. Describe the DNP Message Structure.
2. Parse DNP messages.

Section 4 – Message Structure – cont.

FT3 Frame Format

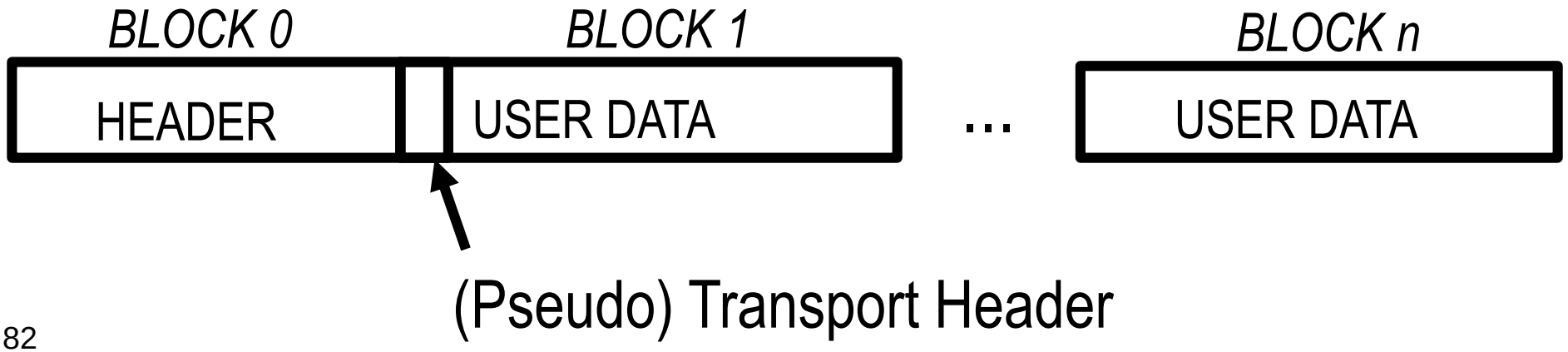
- Header has fixed length of 8 octets. However, three of these octets are not considered user data.
- Blocks of user data have a maximum length of 16 octets.
- A frame has a maximum length of 255 octets of user data not including 2 octet CRC



Section 4 – Message Structure – cont.

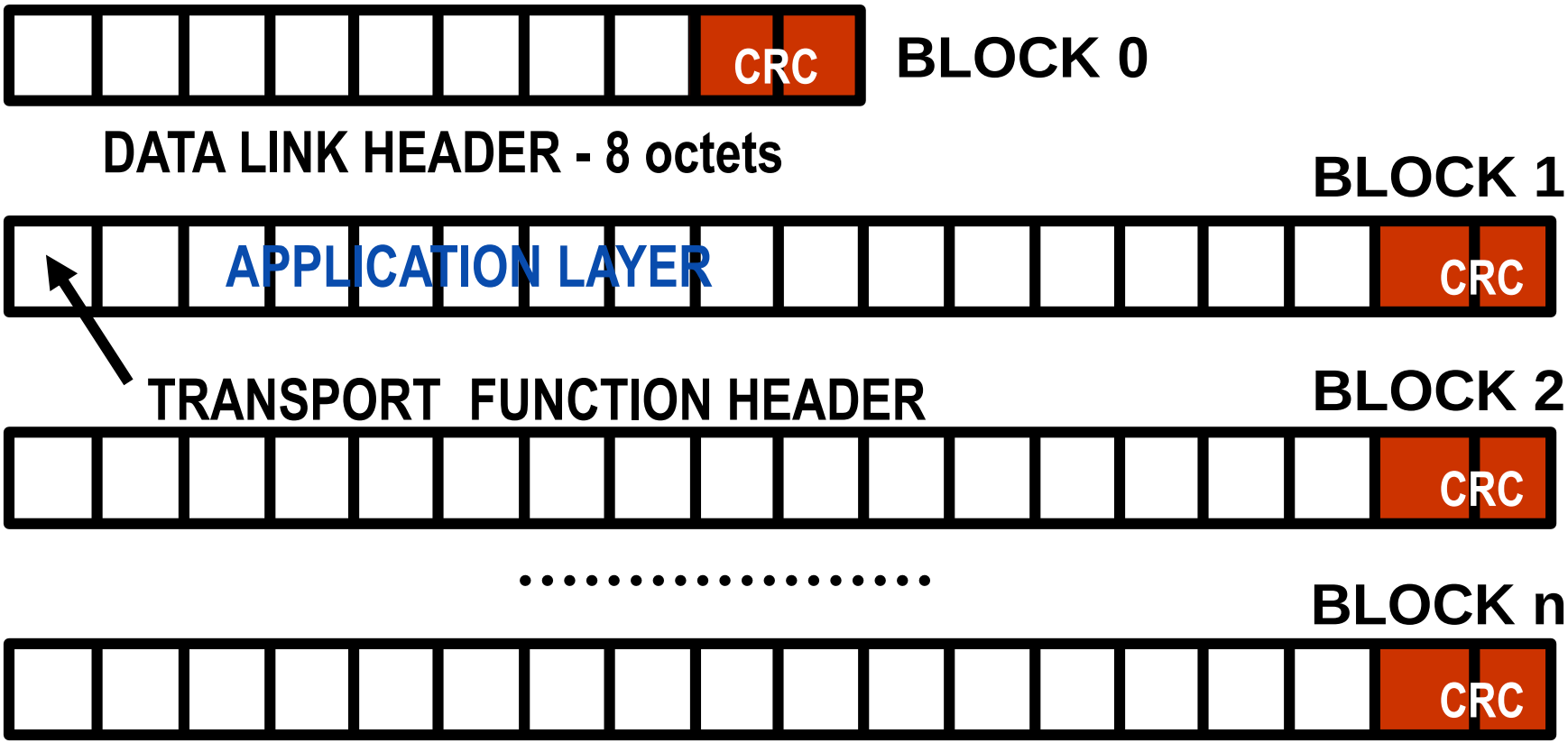
Transport Functions

- Required because some messages will not fit in the 255 octets of user data allowed.
- Breaks down large messages to be transported.
- Assembles large messages that are received.
- Uses a sequence number to track data.



Section 4 – Message Structure – cont.

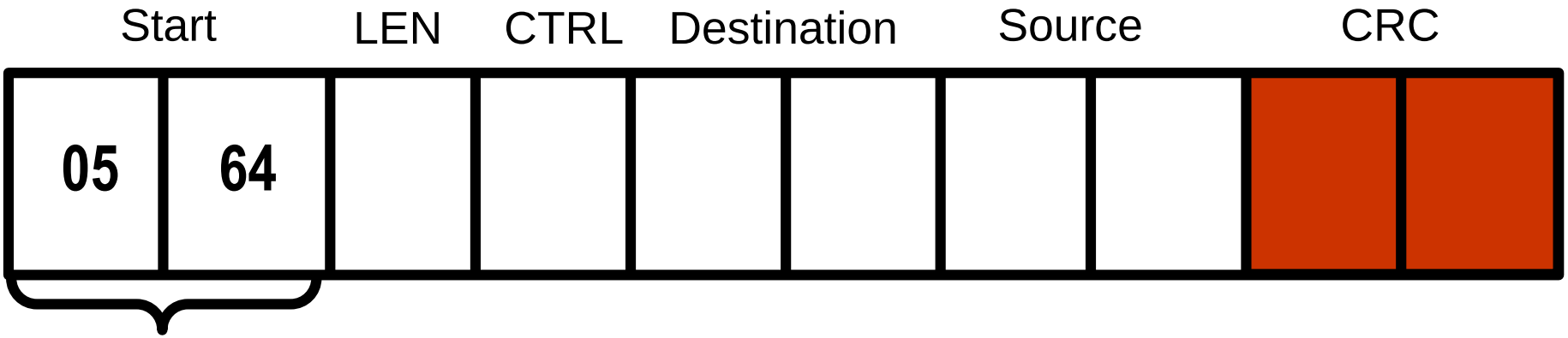
Typical Message Layout



Section 4 – Message Structure – cont.

Data Link Header

- The data link header is used on the data link layer of the protocol to ensure proper routing and error checking of a message.

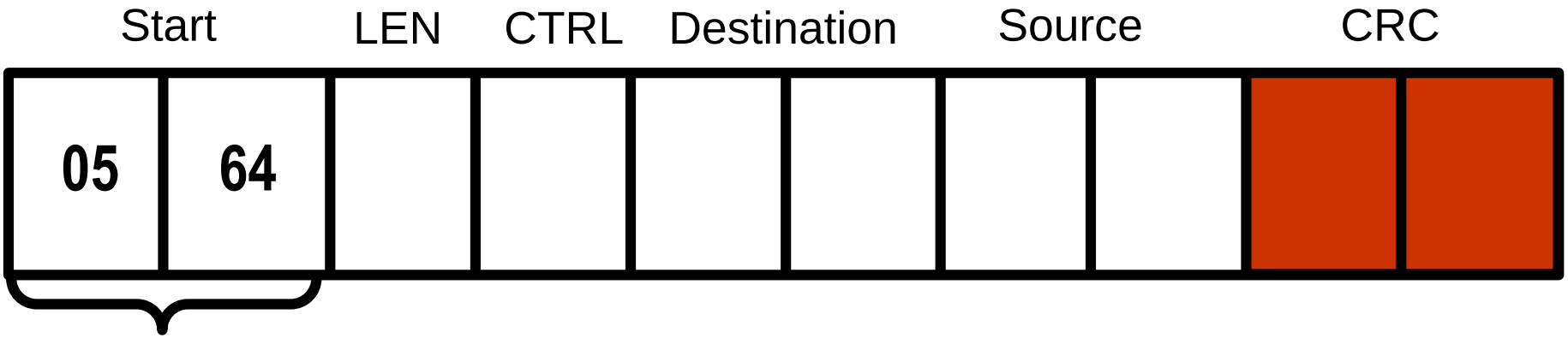


Start octets are always as shown.

Section 4 – Message Structure – cont.

Data Link Header

- The ‘Start’ field always consists of the two octets 05 64 (hex) indicating the start of a message.
- The ‘Length’ field indicates the total number of octets in a message, excluding the START, LEN, and CRC octets.
- The ‘Frame CTRL’ field ensures that the frames are delivered properly and in the correct order.

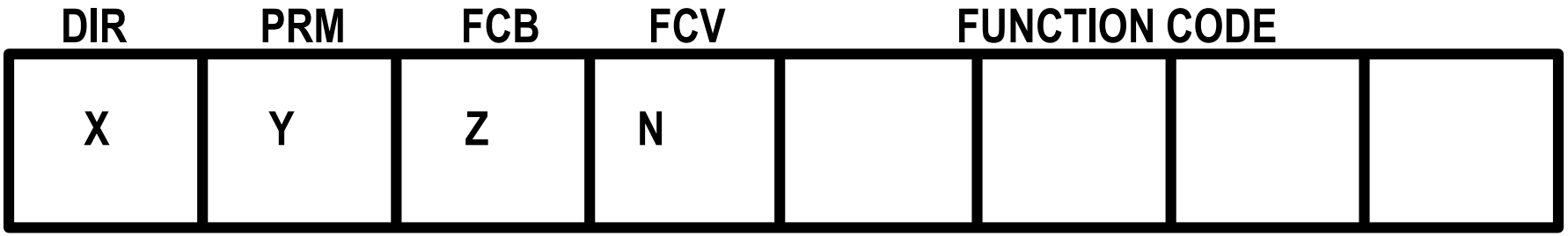


Start octets are always as shown.

Section 4 – Message Structure – cont.

Frame Control Octet

DIR	= X	If the direction bit is set, message is sent by the master.
PRM	= Y	If the PRM bit is set, message is sent by the primary.
FCB	= Z	The FCB (Frame Count Bit) toggles state each time the primary sends a message.
FCV	= N	If the FCV (Frame Count Valid) is set, the FCB is enabled, if not it is ignored.



Section 4 – Message Structure – cont.

Primary Function Codes

Function	Frame Type	Service Function
0	Send - Confirm Expected	Reset of remote link
1	Send - Confirm Expected	Reset of user process
2	Send - Confirm Expected	Test function for link
3	Send - Confirm Expected	User Data
4	Send - No Reply Expected	Unconfirmed User Data
9	Request - Respond Expected	Request Link Status

Section 4 – Message Structure – cont.

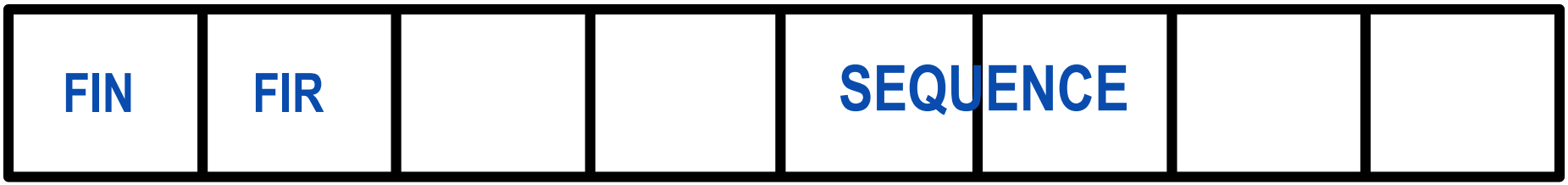
Secondary Function Codes

Function	Frame Type	Service Function
0	Confirm	ACK - Positive Acknowledgment
1	Confirm	NACK - Message Not Accepted
11	Respond	Status of Link (i.e. DFC = 1 or 0?)

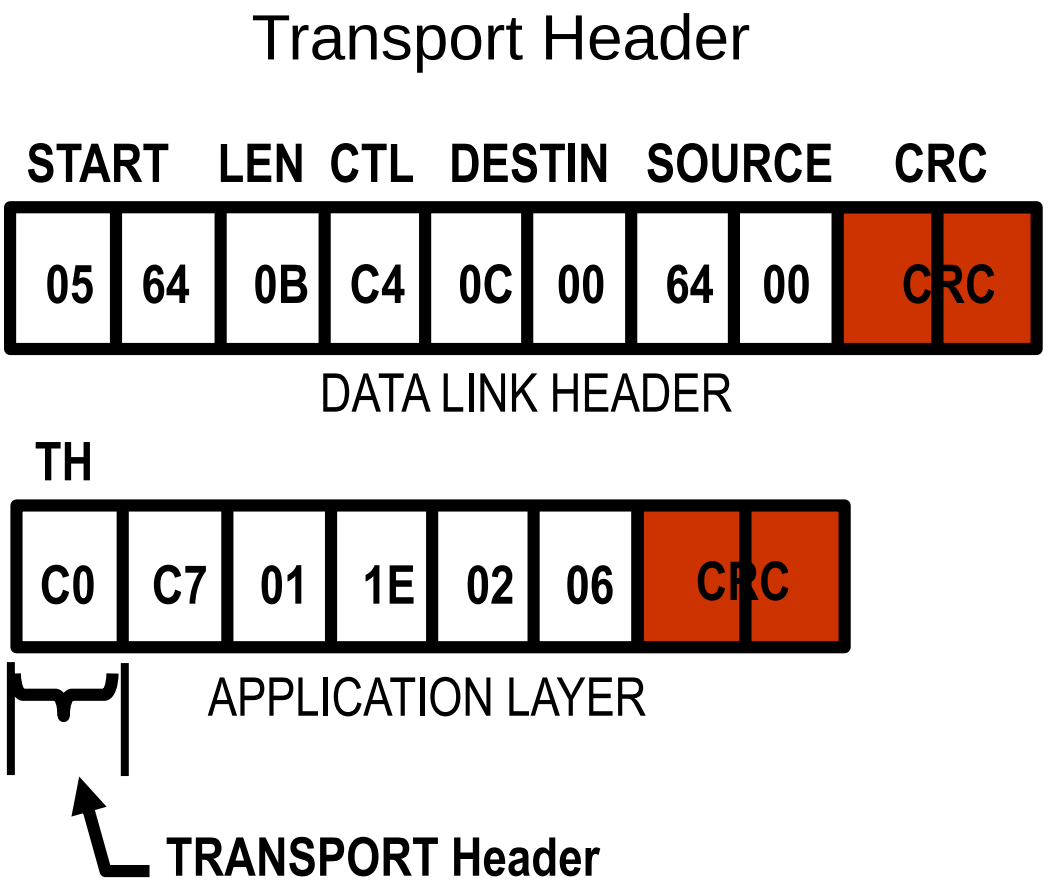
Section 4 – Message Structure – cont.

Transport Header

- This header is required to allow the transmission of messages longer than the 255 octets allowed in one frame.
- The header is a single octet between the Data link Header CRC and the application data
- FIN = Indicates the final frame in a sequence (1 bit)
- FIR = Indicates the first frame in a sequence (1 bit)
- Sequence = Indicates the sequence number of each frame (6 bits)



Section 4 – Message Structure – cont.



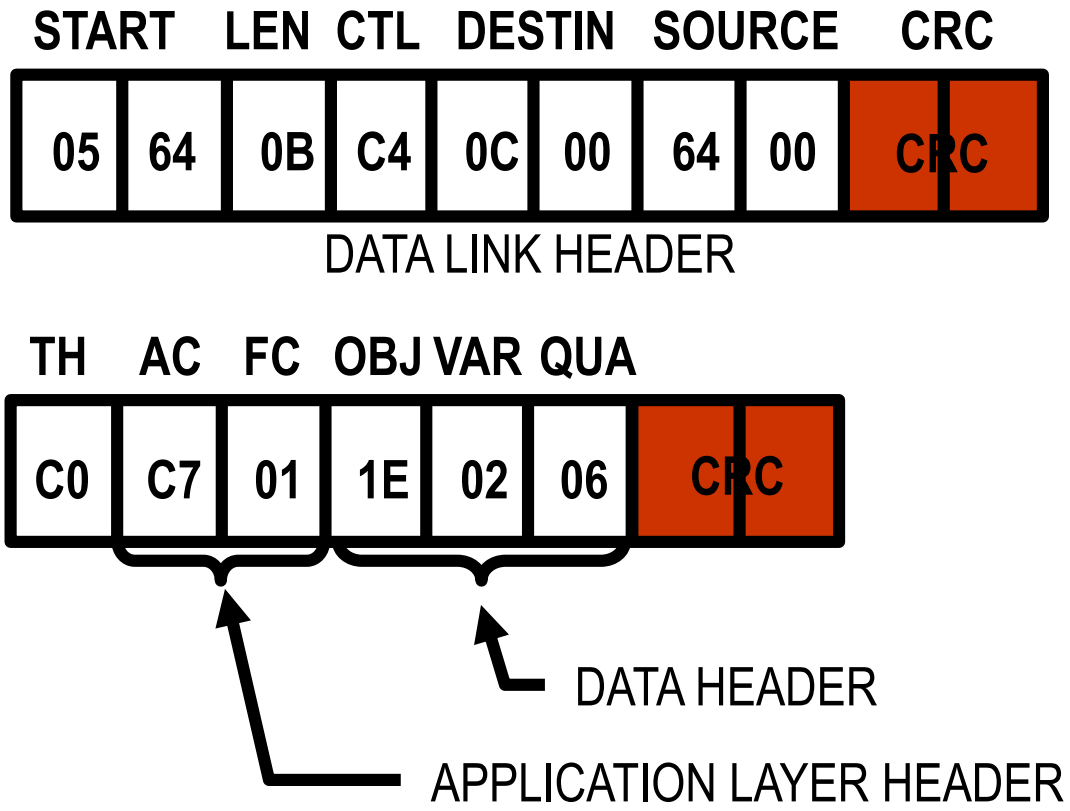
Section 4 – Message Structure – cont.

Application Layer – User Data

- The application layer user data consists of three sections:
  - The Application Header identifies the purpose of the message
  - The Object Header identifies the type of data objects to follow
  - The Data is the actual data being transmitted in the message.

Section 4 – Message Structure – cont.

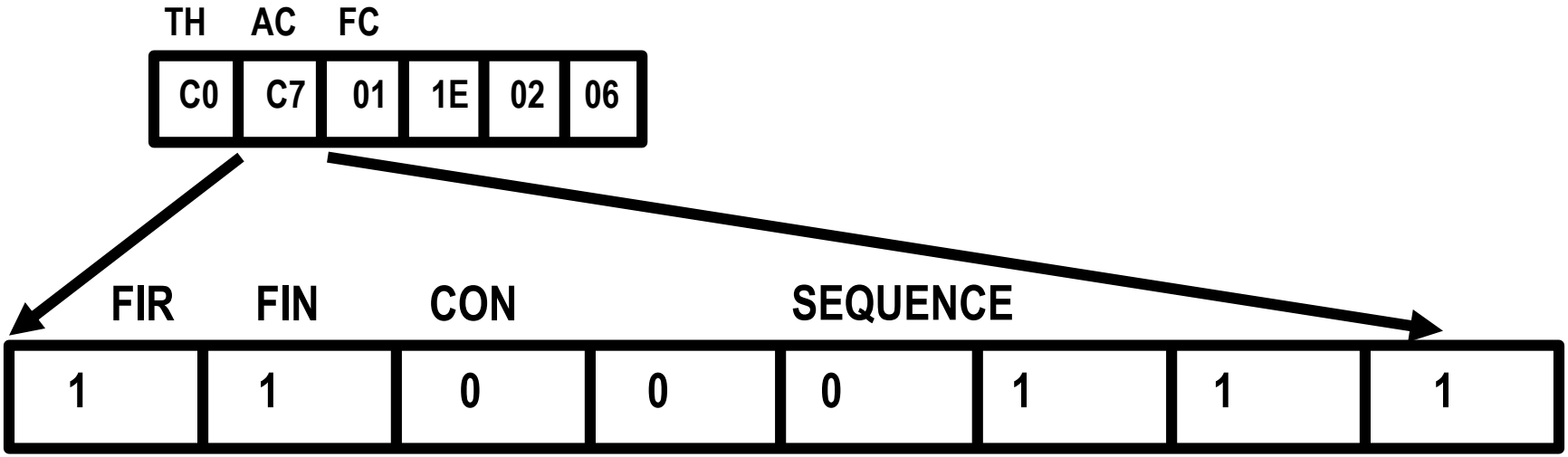
Application and Data Header



Section 4 – Message Structure – cont.

Application Header – Request Message – Application Control

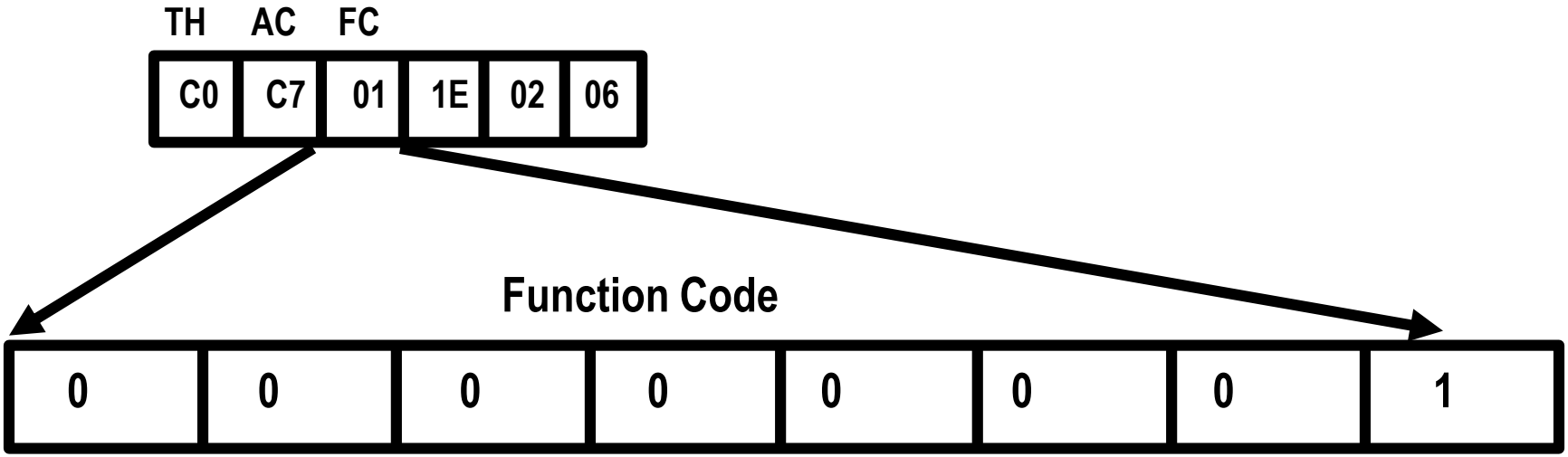
- FIN = Indicates the final fragment in a sequence (1 bit)
- FIR = Indicates the first fragment in a sequence (1 bit)
- CON = Indicates the sending application expects a confirmation (1 bit)
- Sequence = Indicates the sequence number of each fragment (5 bits)



Section 4 – Message Structure – cont.

Application Header – Request Message – Function Code

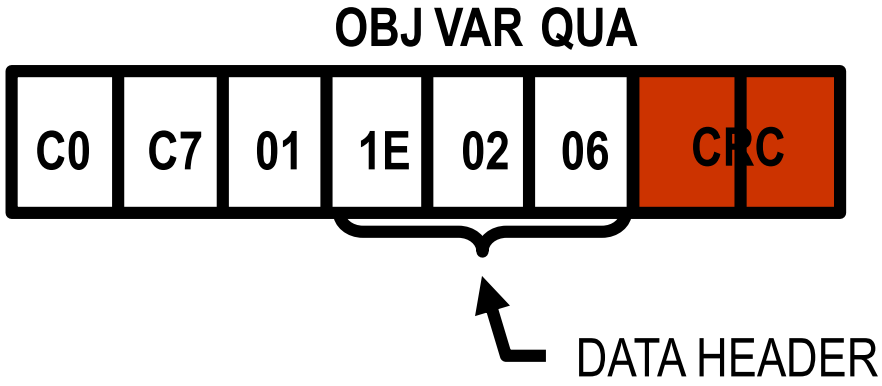
- The function code indicates the function of the application request
- This can include (but is not limited to)
  - See a copy of the DNP Basic 4 for a list of all function codes



Section 4 – Message Structure – cont.

Data Header

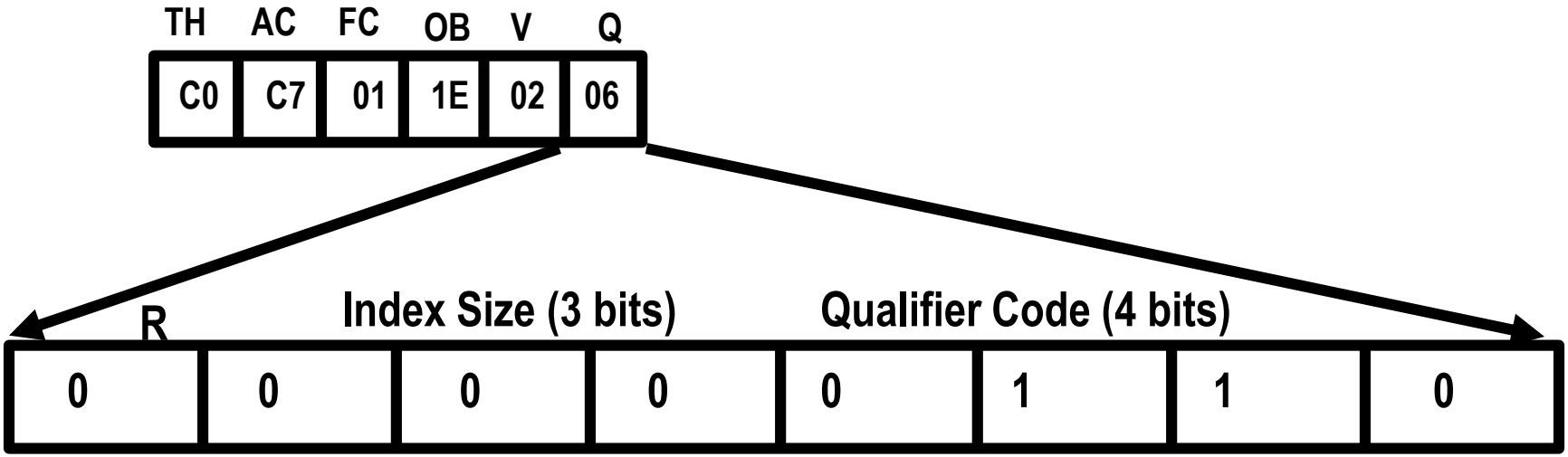
- The data header indicates the type of data that is to follow.
- There are three components to the header:
  - Object = indicates the type of data object
  - Variation = indicates the variation of the data object
  - Qualifier = indicates the range of data that will be presented
  - See the Basic 4 for details on all objects, variations and qualifiers



Section 4 – Message Structure – cont.

Data Header - Qualifier

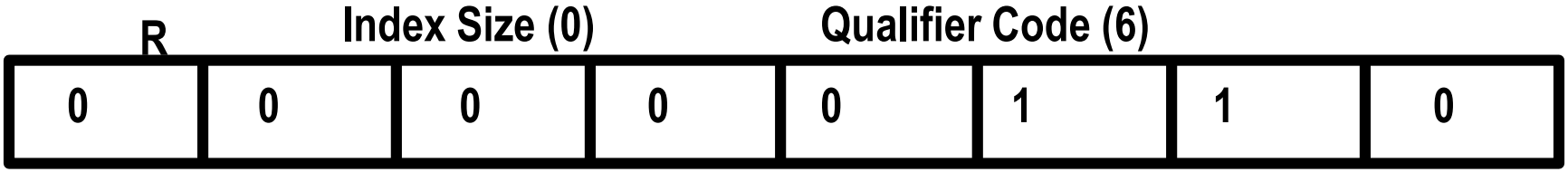
- Determines how data objects will be addressed in the Range Field:
  - The first bit is reserved, and will always be 0
  - Index Size = indicates the size of the index prefixing the data objects
  - Qualifier Code = indicates the size and format of the range field.

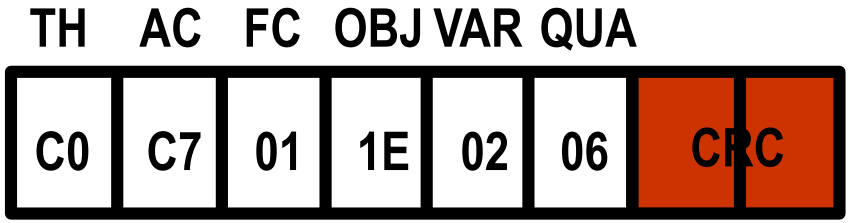


Section 4 – Message Structure – cont.

Data Header – Qualifier – Example 1

- Index = 0x – Data is not prefixed by an index field
- Qualifier = 6x – There is no Range field
- This combination implies that all data for the requested object will be reported.

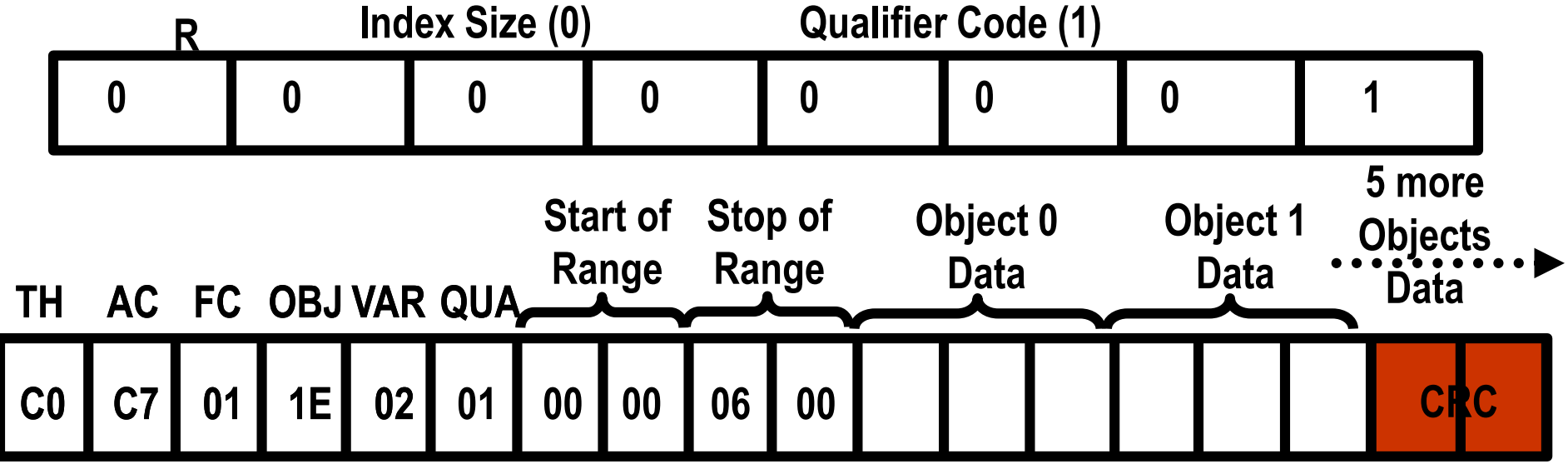




Section 4 – Message Structure – cont.

Data Header – Qualifier – Example 2

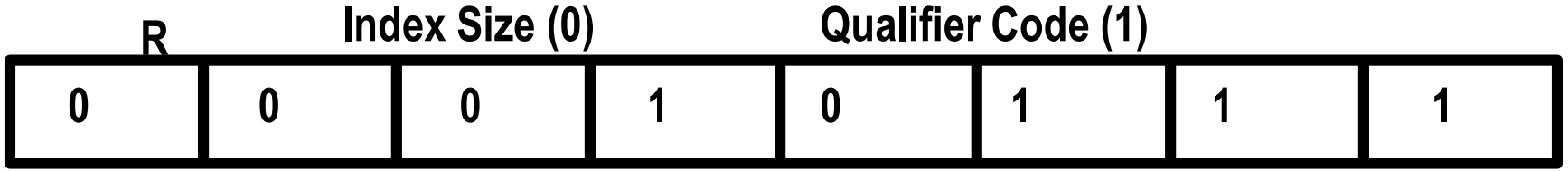
- Index = 0x – Data is not prefixed by an index field
- Qualifier = 1x – 16 bit Start and Stop range field
- The range field will indicate the first and last point to be reported. All points within the range are reported in order.

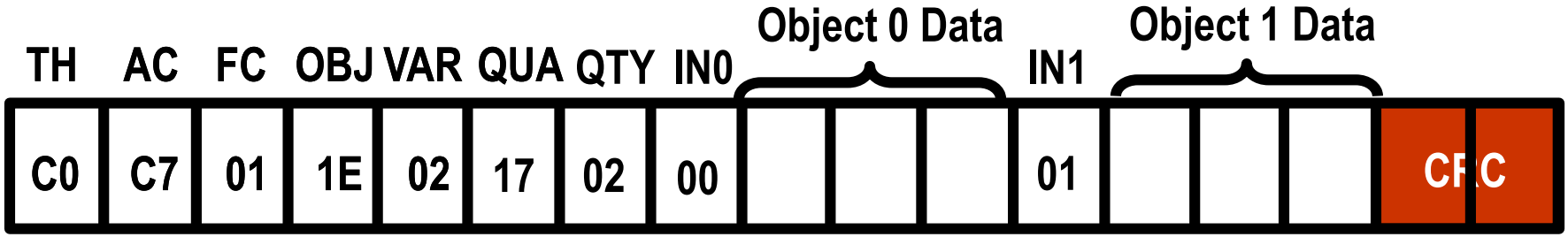


Section 4 – Message Structure – cont.

Data Header – Qualifier – Example 3

- Index = 1x – Data is prefixed by a one octet index field
- Qualifier = 7x – 8 bit quantity in the range field
- The range field will indicate the first and last point to be reported. All points within the range are reported in order.

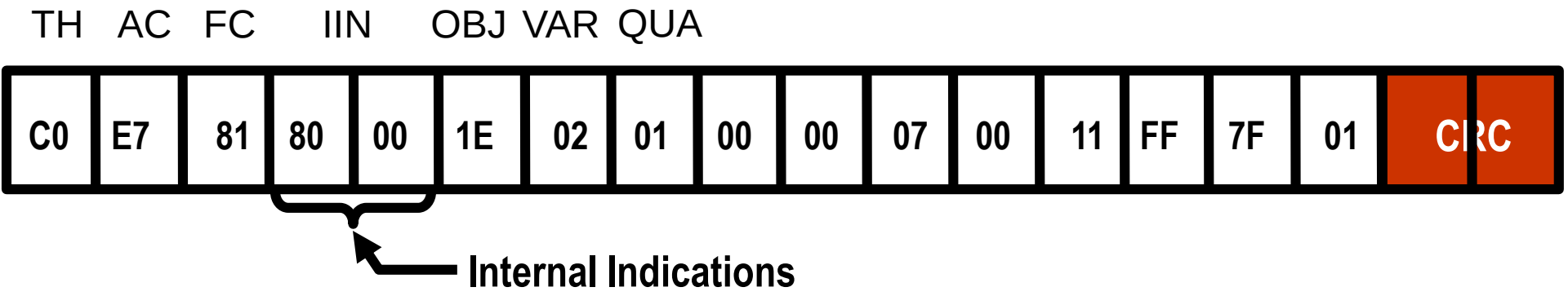




Section 4 – Message Structure – cont.

Application Header – Response Message

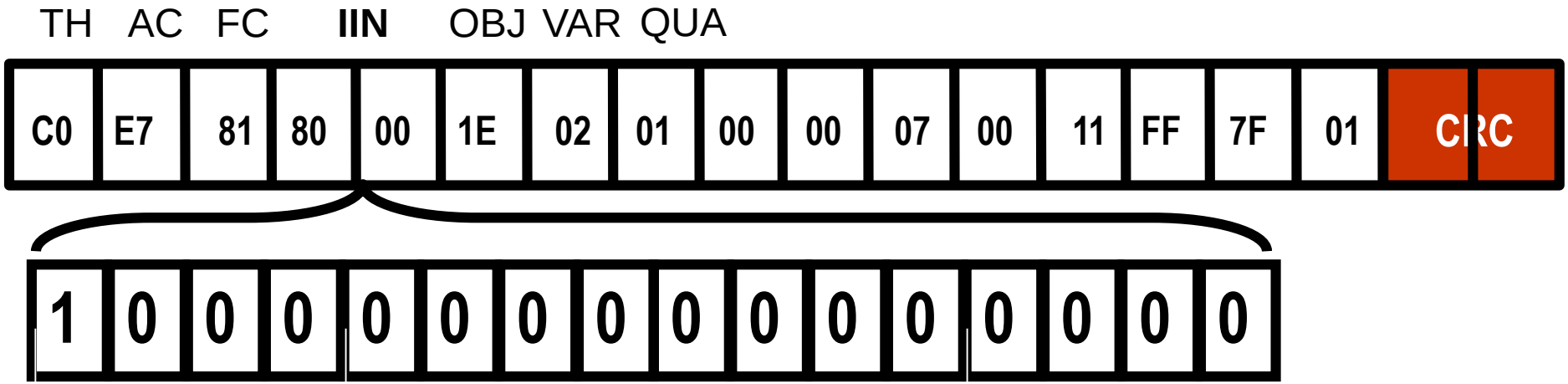
- The application header for a response message is different that that of a request message.
- A 16 bit Internal Indication (IIN) is installed after the function code and before the Data Header



Section 4 – Message Structure – cont.

Application Header – Response Message – Internal Indications

- The internal indication field breaks into 16 flags each indicating a different condition in the RTU.
- Refer to the DNP Basic 4 for a full list of all internal indications.

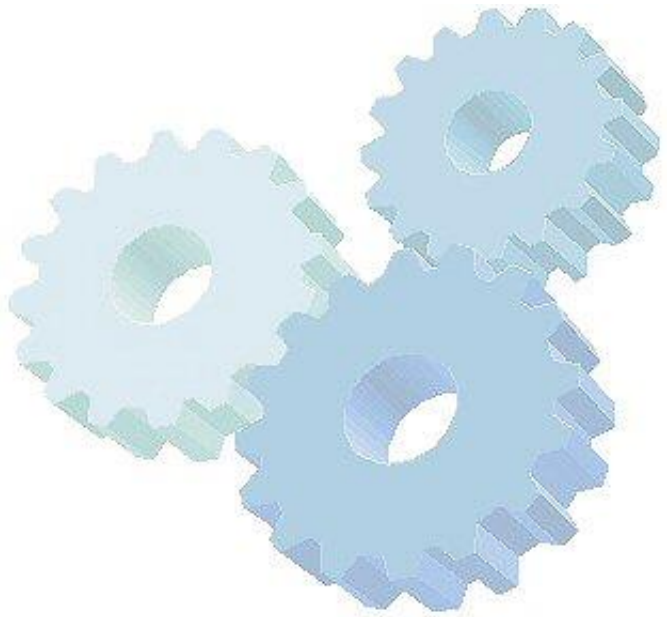


Section 4 – Message Structure – cont. Review Questions

1. Describe the components of the Data Link Header.
2. What is the function of the Transport Header?
3. Describe the components of the Application Header, including the data header and IIN.
4. What do each of the 16 bits in the IIN represent?
5. List at least 3 request and 2 response function codes.

DNP Parsing

LAB 2



DNP Parsing

Objective:
The objective of this lab is to identify and parse several DNP messages.

Introduction:
This lab will consist of several sections:

1. Parse pre-captured DNP messages.
2. Use SA Com to locate a class 123 scan and response and parse the messages.
3. Use SA Com to locate an integrity scan and response and parse the messages.

LAB 2

DNP Parsing – cont.

Procedure 1: Pre-captured Messages

1. Select two of the six messages from the supplementary training materials.
2. Using the parsing work sheets, parse the DNP message.

LAB 2

DNP Parsing – cont.

Procedure 2: Class 123 Scan

1. Using the configuration created in LAB1, capture 30 seconds worth of communication.
2. Locate a class 123 scan and response in the communications capture.
3. Parse the class 123 scan.

LAB 2

DNP Parsing – cont.

Procedure 3: Integrity Scan

1. Using the configuration created in LAB1, capture 2 minutes worth of communication.
2. Locate an integrity scan and response in the communications capture.
3. Parse the integrity scan.

LAB 2

We're done!