Standard Network Diagnostic Assembly

Raymond Romito
Rockwell Automation

October 10, 2018

Introduction

- The Standard Network Diagnostic Assembly whitepaper describes new diagnostic material coming to the CIP Networks Library
- It covers the concepts in TDE 0001-025, Network Diagnostic Assembly
  - Created by the Roundtable for EtherNet/IP Developers
    - Defines diagnostic structures in six object classes
    - New common service to help clients interpret structure information
    - Establishes device-independent Assembly Object instance IDs
      - Standard Network Diagnostic Assembly
    - Includes enhancements to Assembly object definition
      - New attribute, and access rules for diagnostic assemblies
    - Describes for a scalable diagnostic architecture in devices that allows for variations in product features and port assignments
  - First, a little background on what has been driving this

Background - Roundtable Diagnostic Activities

Created a Diagnostic System Framework

- Created early on to illustrate the concepts this activity is considering
- This has been the reference for the Roundtable's work

Created scope of work document for a diagnostic system (TDE-0001-023)

- Includes data definitions, event logging, delivery mechanisms

Refined/added diagnostic attributes to various objects

- These were added to the CIP Networks Library in 2016

EtherNet/IP Troubleshooting Guide

- Currently work in progress

Standard Network Diagnostic Structure Proposal (TDE 0001-025)

- Approved by TRB in spring 2018

Current State

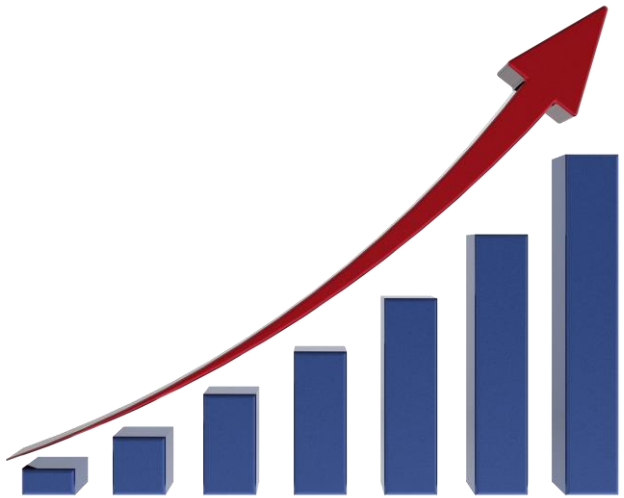
- TDE 0001-025 has been translated into two Specification Enhancements
  - *CIPSE 0001-283 Standard Network Diagnostic Assembly*
  - *ESE 0001-064 Diagnostic Connection Points*

- These are currently going through SIG review
  - The paper is based on what was provided to the SIGs, which could change

- Targeting inclusion in the Spring 2019 release of the CIP Networks Library

The Need for Diagnostics

- There are more devices connected than ever before
  - And it continues to grow
- Networks are larger and more complex
- Information is needed
  - To effectively manage networks
  - To troubleshoot networks that have issues
  - For early detection of potential problems
- This presents a few challenges



Challenges for Diagnostics – Data Organization

- Variation between devices is prevalent
  - Many device features are optional in CIP
  - Optional features often vary from device to device
  - For many devices, user-selected feature settings can change data

- Tools and Users must deal with this inconsistency, often on a device by device basis
  - Customer application programs are more complex

Challenges for Diagnostics – Data Organization

- Consider these commonly used diagnostic values
  - Only four are part of required attributes
- Client tools must deal with variations on a device by device basis
- This adds complexity

Attribute	CIP Path (class/inst/attr)	Required?
Auto or Forced	F6/01/02, bits 2-4	Yes
CPU Utilization	05/01/11	No
Link Status	F6/01/02, bit 0	Yes
Port Speed	F6/01/01	Yes
Duplex	F6/01/02, bit 1	Yes
Ethernet Errors	F6/01/14	No
CIP Connections	05/01/05	No
TCP Connections	F5/01/16	No
HMI PPS	05/01/17	No
Connection Timeouts	05/01/08	No
Class 1/0 PPS	05/01/15	No
Missed I/O packets	05/01/18	No

Challenges for Diagnostics – Messaging Requirements

- Diagnostic data is organized as attributes of various objects
- The attributes must be discretely read
  - Requires 10 Get Single requests to retrieve 12 diagnostic values
- Consider the impact on a large network
  - 100 nodes x 10 messages/node – 1000 messages
  - Polled every 5 seconds = 1 message every 5ms

Attribute	CIP Path (class/inst/attr)
Auto or Forced	F6/01/02, bits 2-4
CPU Utilization	05/01/11
Link Status	F6/01/02, bit 0
Port Speed	F6/01/01
Duplex	F6/01/02, bit 1
Ethernet Errors	F6/01/14
CIP Connections	05/01/05
TCP Connections	F5/01/16
HMI PPS	05/01/17
Connection Timeouts	05/01/08
Class 1/0 PPS	05/01/15
Missed I/O packets	05/01/18

Customer Wish List

- One place to go for diagnostics
  - Standardized for all products and vendors
- Accommodate the variations in devices
  - Extensible with minimal configuration requirements
- Minimize traffic needed to obtain the data
  - Quieter networks are better
- Simplify control programs
  - Lower complexity usually means lower development and maintenance costs
- Minimize configuration of diagnostics
  - Let tools figure it out instead of making a controls engineer do it

How The New Functionality Addresses the Wish List

- Standard Assembly Instance puts the data at a known location regardless of Device Type
- Standardized content is described in class-specific diagnostic structures
  - Variable based on features implemented and user feature assignments
- Rules governing how the content is organized help reduce the need for special handling of devices
- All this results in less messaging to get the pertinent data
- Devices determine the content that is provided, no user setup needed

- Let's take a closer look at the details

Standard Assembly Instance

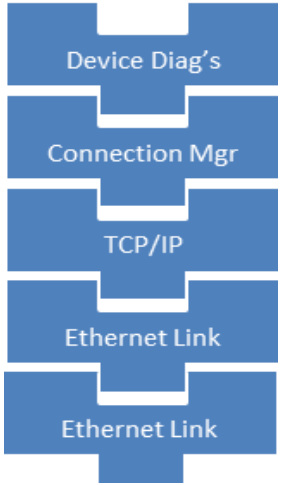
- Range of IDs are reserved for standard assemblies
  - Independent of device type
- Standard Network Diagnostic Assembly – instance 0xD2
- Other instances can be used for different diagnostic information
  - Not in scope for the current SEs

Table 5A-5.2 Assembly Instance ID Ranges

Range		Open **Device Profile Usage**	Vendor-specific **Device Profile Usage**	Quantity
Hex	Decimal			
0x01 0x63	1 - 99	Open - Defined by the device profile	Vendor Specific	99
0x64 0xC7	100 - 199	Vendor Specific	Vendor Specific	100
0xC8 - 0xD1	200 - 209	Open - Defined by the device profile	Vendor Specific	10
0xD2 – 0xD7	210 - 215	Predefined Diagnostic Assemblies	Predefined Diagnostic Assemblies	6

Standard Network Diagnostic Assembly

- Instance 0xD2
- Spec-defined content and placement rules make it easier for clients to understand content
- Accommodates configuration and/or feature variations
- Designed to create a “pluggable” diagnostic structure
- Rules for Member List content assures forward/backward compatibility



Instance 0xD2 Content Definition

Object Class Diagnostic Structure	Placement in the Assembly	Number of Instances Required
Member List Signature	1	1 indicates whether the Member List has changed
Ethernet Link	2	1 instance per EtherNet/IP capable port on the device
TCP/IP Interface	3	1 instance per EtherNet/IP port that has individually configured IP address settings
Connection Manager	4	1 instance
Device Level Ring	5	1 instance per pair of ports configured for DLR (see DLR tables below), omitted if no ports are configured for DLR operation.
Parallel Redundancy Protocol	6	1 instance per pair of ports configured for PRP, omitted if no ports are configured for PRP.
Time Sync	7	1 instance if the device is configured to support CIP Sync, omitted if the device is not configured to support CIP Sync.

Figure 2 Member Content/Placement for the Standard Network Diagnostic Assembly

- This is not network-specific, it is applicable to any CIP network
  - Requires future additions to network specific objects e.g. Dnet object, etc
- Rules governing: placement and number of instances of members, when/if member can be omitted
- Limits members to just the listed items

Instance 0xD2: Examples of 2 Common Devices

- 1-Port EtherNet/IP Device

Member List Value	Source Object	Contents			
		Attribute Name		Data Type	Size
16, 6, "20 04 24 D2 30 05"	Assembly object, Instance 0xD2, Attribute 5	Member List Signature		UINT	2 bytes
16, 0, ""	Pad	Shall be zeroes		n/a	2 bytes
128, 6, "20 F6 24 01 2C 01"	Ethernet Link Object, Instance 1, Connection Point 1	Link Status	Bit 0	BOOL	16 bytes
		Half/Full Duplex	1	BOOL	
		Negotiation Status	2-4	BOOL	
		Reserved	5	BOOL	
		Local Hardware Fault	6	BOOL	
		Reserved	7	BOOL	
		Reserved		3 bytes	
		Port Speed		UDINT	
		Link Down Count		UDINT	
		Ethernet Errors		UDINT	
64, 6, "20 F5 24 01 2C 01"	TCP/IP Interface Object, Instance 1, Connection Point 1	Non-CIP Encapsulation Messages/Second		UDINT	8 bytes
		Active TCP Connections		UINT	
		16 bits pad		n/a	
224, 6, "20 05 24 01 2C 01"	Connection Manager Object, Instance 1, Connection Point 1	CIP I/O Connections		UDINT	28 bytes
		Missed I/O Packets		UDINT	
		Explicit Packets Per Second		UDINT	
		I/O Packets Per Second		UDINT	
		CIP Explicit Connections		UDINT	
		Connection Timeouts		UINT	
		CPU_Utilization		UINT	
		Percent I/O Utilization		UINT	
		16 bits pad		n/a	
				Total Size	**56 bytes**

- 2-Port EtherNet/IP Device

Member List Value	Source Object	Contents			
		Attribute Name		Data Type	Size
16, 6, "20 04 24 D2 30 05"	Assembly object, Instance 0xD2, Attribute 5	Member List Signature		UINT	2 bytes
16, 0, ""	Pad	Shall be zeroes		n/a	2 bytes
128, 6, "20 F6 24 01 2C 01"	Ethernet Link Object, Instance 1, Connection Point 1	Link Status	Bit 0	BOOL	16 bytes
		Half/Full Duplex	1	BOOL	
		Negotiation Status	2-4	BOOL	
		Reserved	5	BOOL	
		Local Hardware Fault	6	BOOL	
		Reserved	7	BOOL	
		Reserved		3 bytes	
		Port Speed		UDINT	
		Link Down Count		UDINT	
		Ethernet Errors		UDINT	
128, 6, "20 F6 24 02 2C 01"	Ethernet Link Object, Instance 2, Connection Point 1	Link Status	Bit 0	BOOL	16 bytes
		Half/Full Duplex	1	BOOL	
		Negotiation Status	2-4	BOOL	
		Reserved	5	BOOL	
		Local Hardware Fault	6	BOOL	
		Reserved	7	BOOL	
		Reserved		3 bytes	
		Port Speed		UDINT	
		Link Down Count		UDINT	
		Ethernet Errors		UDINT	
64, 6, "20 F5 24 01 2C 01"	TCP/IP Interface Object, Instance 1, Connection Point 1	Non-CIP Encapsulation Messages/Second		UDINT	8 bytes
		Active TCP Connections		UINT	
		16 bits pad		n/a	
224, 6, "20 05 24 01 2C 01"	Connection Manager Object, Instance 1, Connection Point 1	CIP I/O Connections		UDINT	28 bytes
		Missed I/O Packets		UDINT	
		Explicit Packets Per Second		UDINT	
		I/O Packets Per Second		UDINT	
		CIP Explicit Connections		UDINT	
		Connection Timeouts		UINT	
		CPU_Utilization		UINT	
		Percent I/O Utilization		UINT	
		16 bits pad		n/a	
				Total Size	**72 bytes**

Additions to the Assembly Class Definition

- Add the Member List Signature attribute (instance attribute #5)
  - Provides a signature over the Member List (Attribute #2) contents
    - Value changes when device features change (causes the Member List to change)
    - Detects positional and content changes in the data
  - Clients use this to tell when the Member List needs to be parsed for changes
  - Always the first member in the Standard Network Diagnostic Assembly

- New assembly type is needed because of the Standard Network Diagnostic Assembly requirements are not quite in line with dynamic or static assemblies
  - e.g. the Member List attribute is required as read only, Member List Signature required as first member

Diagnostic Content for Object Classes

- Diagnostic content will be defined in a new optional section in the object definition “Diagnostic Connection Points”
  - SIGs maintain, know the diagnostic significance/importance of the content
- Defines one or more structures of diagnostic attributes from the class
  - Values that when they change indicate that device/network health or resource loading has changed
  - Does not include values that are generally static like device settings
- These structures are addressed using the Logical Segment type “Connection Point”
  - These have typically only been used for establishing I/O connections before

Connection Point Definition Details

- Provides the building blocks for standardized ‘pluggable’ diagnostic content
- Doesn’t require the use of EDS files
  - Many client devices can’t utilize these effectively
- Together with the Standard Diagnostic Assembly, the content is discoverable
  - Assembly instance is always 0xD2
  - The Assembly Member List contains the EPATHS to the object-class defined structures
  - Classes implement a new CIP Common service it obtain the EPATHs of the members of each Connection Point
    - This allows devices to use Parameter Instances to provide more meta data about the diagnostic attribute

Example Diagnostic Structure

- Example of Connection Point 1 of the Connection Manager object

Attribute ID	Attribute Name	Data Type	Display Name	Attr Size	Size of Structure
19	CIP I/O Connections	UDINT	Active I/O Connections	4 bytes	28 bytes
18	Missed I/O Packets	UDINT	Missed I/O Packets	4 bytes	
17	Explicit Packets Per Second	UDINT	Explicit Packets Per Second	4 bytes	
15	I/O Packets Per Second	UDINT	I/O Packets Per Second	4 bytes	
20	CIP Explicit Connections	UDINT	Active Explicit Connections	4 bytes	
8	Connection Timeouts	UINT	CIP Connection Timeouts	2 bytes	
11	CPU_Utilization	UINT	CPU Percent Utilization	2 bytes	
16	Percent I/O Utilization	UINT	Percent I/O Utilization	2 bytes	
n/a	16 bits pad	n/a	Shall be zero	2 bytes	

Connection Point Definition Details Continued

- The model developed is generic
  - Can be used for more than just network diagnostics
- The specification content establishes the rules for managing the content
  - Not just what goes into it but the order and packing
  - Future modification requirements are established
    - Can be extended, cannot have members removed/replaced, etc
    - Goal is to provide forward compatibility for clients and devices
- Supporting these implies all the member attributes are supported
  - Some are optional
  - No “holes” or placeholders for attributes not supported

Diagnostic Application Profile

- Borrows on the concept in Volume 2, Chapter 8 for COTS vs Industrial performance levels for the Ethernet interface port
- A way to specify that optional content is required if a product meets this Application Profile
- Designed to raise the bar for EtherNet/IP and CIP
- Makes it easier for vendors to know what optional content they should implement
- Define testable content that ODVA's CA can, if desired, test for compliance with
- This definition is planned as a potential next step in enhancing diagnostics capabilities of EtherNet/IP devices

THANK YOU