Impact of EDS Requirement Changes on Host Applications and EDS Files

Greg Majcher
Principal Application Engineer
Rockwell Automation

Kevin Peters
Senior Software Architect
Rockwell Automation

Presented at the ODVA
2026 Industry Conference & 24th Annual Meeting
October 21, 2026
Vilamoura, Portugal

Abstract

Recent enhancements to The CIP Networks Library Volume 1 modify requirements for device Electronic Data Sheets (EDS) and for the first time introduce requirements for applications that interpret them. The specification has historically left many features optional to promote EtherNet/IP adoption. In some cases, this led to interoperability issues. In others it created a disincentive for feature adoption due to not knowing if any other products supported a feature. The new requirements are expected to enhance consistency, interoperability, and ease device vendor development efforts by providing a predictable baseline for EDS file support.

This paper presents an overview of these changes, their impact on host applications and EDS files, and the resulting benefits to the EtherNet/IP ecosystem and our end users.

Keywords

Electronic Data Sheet (EDS), CIP, EtherNet/IP, Metadata, Interoperability, Device Profiles, EDS Interpreter

Definition of terms

EDS (Electronic Data Sheet) - A file used to describe a device's capabilities, parameters, and communication behavior.

EDS Interpreter - An application or tool that processes EDS files to configure or interact with devices.

Attribute Metadata - Additional semantic information describing device attributes beyond basic structure.

O→T – This nomenclature refers to information being transmitted between an originator and a target.

T→O - This nomenclature refers to information being transmitted between a target and an originator.

 ©2026 ODVA, Inc.

Classified as Business

Introduction

If you were born in this century it may come as a surprise to you that many industrial devices were at one time resource constrained. That is still true today for some types of products, but today it is more of a design decision than a necessity as it was years ago. Our network history even points to this. Historically vendors put more capable products on ControlNet and used DeviceNet or CompoNet for smaller, less capable devices. Early DeviceNet products were implemented with 8-bit microcontrollers and several kilobytes of memory. For this reason, we as CIP vendors have always been careful to specify functionality in the most efficient ways. That same philosophy was applied to the required functionality for products. As a vendor community we have appreciated that functionality was optional, giving us the ability to implement what we saw most valuable to our customers. However, the lack of required content led to:

- Inconsistent levels of detail across devices
- Interoperability issues between vendors
- Limited support for advanced features

One manifestation of these policies resulted in “stub” EDS files—minimal implementations sufficient for conformance testing but insufficient for interoperability across tools and the ecosystem.

Fast forward 30 years and you will find EtherNet/IP installations utilizing products containing memory and CPU bandwidth orders of magnitude larger than the products from those early days. The balance has shifted and now our focus can be on providing great functionality for end users and less on conserving resources.

Issues With Optionality

At the 2023 Industry Conference’s Roundtable for EtherNet/IP a discussion was held regarding interoperability. It was noted that historically the Roundtable for EtherNet/IP provided a forum for vendors to express interest in new functionality and even to get early feedback at “plugfests”. In recent years, those activities have been sparsely scheduled and attended, leading to a disconnect between control and device vendors. There was a renewed interest in reactivating this forum to bridge the gap between specification and implementation across vendors. Early interactions can provide insight into levels of adoption and help ensure that specification language is clear.

There were several other notable things that came out of that conversation. It became clear that CIP had several “chicken-or-the-egg” issues with respect to optional functionality. Control vendors were heard saying that they did not implement certain functionality because an insufficient number of device vendors supported it. Device vendors replied that they did not implement the functionality because no control vendors provided support. Some device vendors reported that customized EDS files were needed to get their products working with different control vendors’ EDS interpreters. These were not new issues; many have been previously recognized (see Schiffer [5]) but not adequately addressed.

The range of interoperability issues related to optionality can be described as follows:

- A fully supported feature in a target device cannot be used because the EDS interpreter provides no support
- A fully supported feature in an EDS interpreter cannot be used in a system because not all devices provide support
- A vendor provides partial support that misaligns with the partial support provided by a controller or device on the other end of the feature
- A feature is implemented with a unique interpretation that does not interoperate with other implementations of the feature
- An EDS interpreter requires an optional feature for its operation

All vendors want to have the confidence that if they invest in adding functionality to their products, that it can be used. Our customers want the assurance that if they buy conformance tested products, they will

work together. These concerns were the driver for the CIP System Architecture SIG to examine features with poor adoption and potential interoperability issues that stem from optional features. They reevaluated that optionality to define a consistent baseline for EDS content and interpreter behavior. This resulted in updates to The CIP Networks Library Volume 1 requirements for, and semantics of, functionality described by EDS in the April 2026 release of the CIP Networks Library.

To address the potential issues listed above the following guidelines were used when creating the specification enhancements:

- Ensure that specification content is clear and unambiguous. If vendors do not understand the rationale for, or the specification of a feature, they will either choose not to implement it or implement it incorrectly.
- Document the expected level of support for both devices and EDS interpreters.
- For features that could be valuable to a system, mandate support in device EDS files, but leave it optional for EDS interpreters. Limited adoption of a feature by device vendors makes adoption by tools unappealing or impractical. There is no mandate for support in tools because it is still a tool vendor's decision about whether a certain feature brings enough value to their customers.
- Some features are important in certain applications, implementations, or types of devices. These should be optional for devices to support as needed but required for all EDS interpreters and their associated connection originators to ensure that a large variety of devices and applications can be supported with EtherNet/IP.

These guidelines led to requirements being specified for EDS interpreters, changes to the requirements for device EDS content, and enhancements to semantic text.

Specification Enhancements

For as long as the CIP EDS definition has existed, the specification has explicitly defined requirements for the Sections, Entries, and Fields for the EDS files themselves. However, the requirements for EDS interpreters were not as specific. The specification simply stated that interpreters needed to interpret the EDS contents to determine the characteristics of each parameter and present to the user a list of choices or data entry fields for each device parameter. That high-level direction was not enough to ensure interoperability.

The most significant change discussed in this paper is that Edition 3.40 of The CIP Networks Library Volume 1 introduces explicit requirements on EDS Interpreters' use of EDS files. Every feature described in Chapter 7 Electronic Data Sheets has been updated with these requirements. Table 1 shows an example of how tables were updated. Previously there was one column that indicated if section, entry, or field was optional or required. That has been clarified to mean optional or required in the EDS file. A new column was added to indicate required support needed by EDS interpreters.

Table 1 - Example Table from Chapter 7

Field Name	Field Number	Data Type	EDS Interpreter Requirement	~~Required/Optional~~ Device EDS Requirement
Revision	1	UINT	Required	Required
MaxInst	2	UINT	Required	Optional
NumInst	3	UINT	Required	Conditional[1]

Table Footnotes:

1 Required if MaxInst is implemented, otherwise not allowed.

A point worth highlighting is the hierarchical nature of the requirements. EDS files are constructed of sections that contain entry keywords. Each entry keyword contains one or more fields. Requirements

Classified as Business

appear at all these levels which can at first seem confusing, however the requirements are hierarchical. If a section is optional, it does not need to be supported. However, if the section is supported, the implementation must follow the requirements for the entries in the section. The same can be said for the entry fields. If an optional entry keyword is supported, the required fields of that entry must be supported. Following these rules ensures a consistent level of support across vendors leading to interoperability.

These new EDS interpreter requirements address one side of the “chicken-or-the-egg” issue. The published interpreter requirements and conformance testing of interpreters will guarantee that device vendors know which features are supported by interpreter vendors. The other side was addressed by raising the bar on EDS file requirements. Interpreter vendors can now rely on specific EDS entries knowing that they must be present. The official date of enforcement for these changes is April 2027 however; the actual enforcement dates rely on releases of the conformance test tool. Starting with CT22 which will be released in October 2026, the conformance test tool will warn vendors of their EDS file deficiencies. Those warnings will become errors in the CT23 version of the conformance tool. That version will also enforce the EDS interpreter tests. Note, EDS interpreter testing will only be enforced on interpreters used in conjunction with configuring devices and building connections.

The changes being described in this paper fall into three categories: entries necessary for configuring modules and setting up connections, making products more discoverable and self-describing, and general system features. In the remainder of this paper the major changes in each category will be described with the rationale for the changes offered, impact on device EDS files, originators and tools will be discussed, and finally the benefits of the changes will be suggested.

Note that this paper does not contain a comprehensive list of the changes; the specification provides that. Also note that the changes are sometimes presented in a format that does not align directly with the specification text or tables. This was done to be concise in this paper. Finally, this paper uses red underline text for additions and red strikethrough for deletions to make clear what has changed (much like the official specification markups).

Configuring Devices and Connections

To have the greatest impact on interoperability, the initial work focused on EDS sections related to the configuration of devices and their connections. Table 2 shows the section requirements that changed related to configuration and connections.

Table 2 - EDS Sections Related to Configuration and Connections

Section Name	Legal Delimiter	EDS Interpreter Requirement	Device EDS Requirement
Parameters	[Params]	Required	~~Optional~~ Required
Parameter Groups	[Groups]	Required	Optional
Assembly Object	[Assembly]	Required	~~Optional~~ Conditional[1]
Connection Manager Object	[Connection Manager]	Required	~~Optional~~ Required
Parameter Object	[ParamClass]	Required	~~Optional~~ Conditional[1]

Table Footnotes:

1. Required if the device supports the corresponding CIP Object, otherwise not allowed.

Interpreters that configure devices and/or their connections are required to support all the related sections.

Classified as Business

For each connection that a device wants to expose, the EDS file is required to describe that connection in a Connection entry of the Connection Manager section. The O→T and T→O data formats as well as any configuration are often described in assemblies in the Assembly section. All parameters referenced by the assemblies must be defined in Parameter entries. EDS files can optionally organize parameters into groups and if they do, interpreters are expected to present them in those groupings.

Most interpreters and most EDS files were already using these sections as described. However, devices can no longer submit a “stub” EDS file and be compliant. Now that the Connection Manager and Parameters sections are required, EDS files must describe a device’s connections and how to configure them. These changes should ensure that devices can be configured by another vendor’s tool and controllers can communicate with any other vendor’s device. More interesting impacts can be seen on deeper levels where the entries and fields of these sections are described.

Parameter Changes
ParamN entries have 24 fields and interpreters are now required to support most of them. The 17 required fields include the parameter’s name, units string, help string, minimum value, maximum value, and default value. Fields related to internationalization and scaling based on linked parameters are optional.

Parameter descriptors which are required at the instance and class level can be used to describe detailed parameter behavior and interpreters are required to use these descriptors. There are also descriptor bits that can help the interpreter interact with the product such as an indication of Read Only or Write Only, or the Monitor Parameter bit which indicates that the parameter’s value is subject to change and should be refreshed periodically.

EDS files can optionally use a collection of associated entries: EnumN, RangesN, and FixedN to further define a parameter. An Enum entry specifies enumerated strings for parameter values. Ranges entries define ranges of valid values within the overall minimum and maximum range of the parameter’s value. The Fixed entry is used to define bits that are not selectable within parameters of bit string data types. EDS interpreters are required to support all these associated entries of parameters if they are provided.

One of the most impactful changes related to parameters deals with data types. The Data Type field of the ParamN entry has always been a required field in the EDS, however since there had never been requirements on EDS interpreters, there were different data types supported by each vendor’s tools. This resulted in parameters not being presented to users. Table 3 shows the list of required data types for interpreters.

Classified as Business

Table 3 - Required Data Types

Data Type	EDS Interpreter Requirement	Device EDS Requirement
BOOL	Required	Optional
SINT. INT, DINT, LINT	Required	Optional
USINT, UINT, UDINT, ULINT	Required	Optional
REAL, LREAL	Required	Optional
BYTE. WORD, DWORD, LWORD	Required	Optional
STRING, STRING2, STRINGI, SHORT_STRING	Required	Optional
TIME, STIME, UTIME, FTIME, LTIME, ITIME, NTIME	Required	Optional
EPATH	Optional	Optional
ENGUNIT	Optional	Optional
DATE	Optional	Optional
TIME_OF_DAY	Optional	Optional
DATE_AND_TIME	Optional	Optional

Finally, interpreters are required to support assemblies, arrays, and variants. These elements allow device vendors to fully describe their data for use in connections and configuration.

Connection Manager Changes

The Connection Manager section was updated with required support for interpreters. The most common features of connections were made mandatory.

Table 4 shows the newly added requirements for EDS interpreters for the Connection Manager section. They can be summarized as follows:

- If EDS files specify Production Inhibit Time Segment use, interpreters are required to consume that information and expose it when configuring the connection.
- If an EDS file exposes a Maximum Connection Establishment Time, originators are expected to respect that time.
- If IO Aggregation is supported, devices can identify connections that should be excluded from aggregation, and originators are expected to respect that election.

Table 4 - Connection Manager Requirement Changes

Entry Name	Entry Keyword	EDS Interpreter Requirement	Device EDS Requirement
Connection	ConnectionN	Required	Conditional
Proxy Connection	ProxyConnectN	Optional	Conditional
Proxied Connection	ProxiedConnectN	Optional	Conditional
Connectivity Test Connection	ConnectivityTestConnect1	Optional	Conditional
Production Inhibit Time in Milliseconds Network Segment	PITNS	Required	Conditional

Production Inhibit Time in Microseconds Network Segment	PITNS_usec	Required	Conditional
Max Connection Establishment Time	MaxConnEstTime	Required	Conditional
Exclude from Aggregation Connection	Exclude_AggregationN	Conditional *	Conditional*
Table Footnotes: * When I/O Aggregation is supported, this keyword is required.			

Table 5 shows the details of the required interpreter support for connection entries.

Table 5 - Connection Entry Requirement Details

Field Name	Data Type	EDS Interpreter Requirement Details
Trigger and transport	DWORD	Class 1 Cyclic and Change of State Exclusive Owner, Input Only, and Listen Only
Connection parameters	DWORD	Data: fixed size with Real-time transfer formats Connection Type: NULL, MULTICAST, & POINT2POINT Priority: HIGH & SCHEDULED
O→T & T→O RPI	UDINT or ParamN	Constraints specified by fixed value or Param
O→T & T→O size	UINT or ParamN	Size specified by fixed value or Param
O→T & T→O format	ParamN or AssemN or AssemExaN, ArrayN or ConstructedParamN	Structure of data defined in an associated entry
Target Config size	UINT or ParamN	Size specified by fixed value or Param
Target Config format	ParamN or AssemN or AssemExaN, ArrayN or ConstructedParamN	Structure of data defined in an associated entry
Connection name string	STRING	
Help string	STRING	
Path	Padded EPATH	

Discoverable and Describable

A second goal for the changes was to make products more discoverable and self-describing. The EDS definition provided a lot of functionality in this regard, but it was incomplete and adoption was very low. Because of this, no tools could rely on an EDS to fully describe a product. That has changed with two additions and a mandate that if you support an object, you must describe your level of support.

The EDS definition already had entries that allowed you to indicate the presence of CIP objects, instances, attributes, and services in an implementation. Instances were identified using two entries; a maximum instance number entry, and a total instances entry. This presented a problem for devices with sparsely populated lists. If you supported instances 1, 50, and 99, a tool would need to send a lot of messages to discover which three instances were present. A new entry was added to reflect a list of static instances making it very clear which instances were present.

Classified as Business

The EDS definition provided a way for devices to describe services, but it did not provide any mechanism to describe attributes. New entries (that will appear in the 3.41 edition of Volume 1) were defined that enable an EDS to provide detailed information about class and instance attributes. Table 6 shows the definition of these new entries.

Table 6 - New Attribute Details Entry

Field Name	Field Number	Data type	EDS Interpreter Requirement	Device EDS Requirement
Reference	1	ParamN, AssemN, AssemExaN, ArrayExaN, ConstructedParamN	Required	Required
Description	2	STRINGI	Required	Optional
Units	3	UINT[1]	Required	Optional
Access Rule	4	EDS_Field_Keyword, allowed values: GET, SET, GET_SET	Required	Required
Rate of Change	5	EDS_Field_Keyword, allowed values: MS, S, M, H, DAY, STATIC	Required	Required
NV Storage	6	BOOL	Required	Required
Save and Restore	7	BOOL	Required	Optional
Apply Required	8	BOOL	Required	Optional
Set Limited to Implicit Connections	9	BOOL	Required	Optional
Set Protection	10	BOOL	Required	Optional
Explicit Set Requires Class 3 Connection	11	BOOL	Required	Optional

Table Footnotes:

1. See Tables in Appendix D-2 (Engineering Units) for valid values.

With these additions, it is now possible to completely describe the static details of an object's implementation in the EDS.

Object Class Sections are used to document the level of support for any CIP object. The addition of these sections is where device EDS files will see the biggest impact. These sections had never been required and even when they were included in an EDS, few keywords were required to be present. These sections are now required, enabling tools to easily discover what devices support so they can interact with them more efficiently. EDS files are required to have a section for every public object the product supports. Vendor specific classes may optionally be included if the vendor wishes to expose their details.

Common keywords are defined for every CIP object class entry. Any class can add keywords specific to that class. Table 7 shows the common keywords and whether they are required within an EDS file.

Classified as Business

Table 7 - Common Object Class Entry Keywords

Entry Name	Entry Keyword	EDS Interpreter Requirement	Device EDS Requirement
Revision	Revision	Required	Required
Maximum Instance Number	MaxInst	Required	Conditional [1]
Number of Static Instances	Number_Of_Static_Instances	Required	Conditional [1]
List of Static Instances	List_Of_Static_Instances	Required	Conditional [1]
Maximum Number of Dynamic Instances	Max_Number_Of_Dynamic_ Instances	Required	Required
Class attribute identification	Class_Attributes	Optional	Required
Instance attribute identification	Instance_Attributes	Optional	Required
Class service support	Class_Services	Optional	Required
Instance service support	Instance_Services	Optional	Required
Object Name	Object_Name	Conditional	Required
Object Class Code	Object_Class_Code	Conditional	Required
Service Description	Service_DescriptionN	Optional	Optional
Class Attribute Details	Class_Attribute_DetailsN	Required	Optional
Instance Attribute Details	Instance_Attribute_DetailsN	Required	Optional

Table Footnotes:

1. Either List_Of_Static_Instances or the combination of MaxInst and Number_Of_Static_Instances is required. These are mutually exclusive.

The combination of the File, Device, Device Classification, and Object Class sections allow the EDS to present tools with a comprehensive view of the device's information model.

Miscellaneous Features

The final goal was to examine EDS entries covering underutilized features that could provide end users with a lot of value. The SIG searched through ODVA's database of EDS files and found two sparsely adopted features worth investment. Changing the requirements for these should ultimately benefit end users as the features get exposed.

Capacity
The Capacity section documents a device's communication capacity. Clients and client tools could use this information to correctly budget communications for a system. EDS files are now required to populate this section which should incentivize client tools to develop features using the information.

Vendor-specific Extended Device Status
The Device Description section contains two entries where vendor-specific extended device status information can be defined. It is likely that many products chose not to utilize this feature knowing that no EDS interpreters provided support. EDS interpreters are now required to support these fields to expose this vendor specific status information.

Classified as Business

Conclusion

These specification enhancements impact both device and control vendors. Device vendors must provide a more complete description of their information model, describe the structure of their data, and the information needed to configure their devices for IO connections. The EDS interpreters of control vendor's tools are required to support most configuration and communication constructs defined for EDS as well as most of the defined data types. By raising the minimum level of requirements for both EDS files and their interpretation, interoperability issues can be avoided, and valuable features can be brought to life.

These changes address some of the known, cross-vendor interoperability issues that have been identified, but there are more. Additional issues will be studied, and new specification enhancements will be written to correct and clarify the specifications. All vendors are encouraged to participate in this forthcoming work and bring their suggestions for areas of improvement.

References

[1] ODVA, The CIP Networks Library, Volume 1: Common Industrial Protocol, Ann Arbor: ODVA, Inc., 2001-2026.
[2] ODVA, The CIP Networks Library, Volume 3: DeviceNet Adaptation of CIP, Ann Arbor: ODVA, Inc., 1994-2026.
[3] ODVA, The CIP Networks Library, Volume 4: ControlNet Adaptation of CIP, Ann Arbor: ODVA, Inc., 1997-2026.
[4] ODVA, The CIP Networks Library, Volume 6: CompoNet Adaptation of CIP, Ann Arbor: ODVA, Inc., 2006-2026.
[5] Schiffer, Optimizing EDSs for EtherNet/IP I/O Connection Target Devices (https://www.odva.org/library_proceedings/optimizing-edss-for-ethernetip-io-connection-target-devices/)

****

The ideas, opinions, and recommendations expressed herein are intended to describe concepts of the author(s) for the possible use of ODVA technologies and do not reflect the ideas, opinions, and recommendation of ODVA per se. Because ODVA technologies may be applied in many diverse situations and in conjunction with products and systems from multiple vendors, the reader and those responsible for specifying ODVA networks must determine for themselves the suitability and the suitability of ideas, opinions, and recommendations expressed herein for intended use. Copyright ©2026 ODVA, Inc. All rights reserved. For permission to reproduce excerpts of this material, with appropriate attribution to the author(s), please contact ODVA on: TEL +1 734-975-8840 FAX +1 734-922-0027 EMAIL odva@odva.org WEB www.odva.org. CIP, Common Industrial Protocol, CIP Energy, CIP Motion, CIP Safety, CIP Sync, CIP Security, CompoNet, ControlNet, DeviceNet, and EtherNet/IP are trademarks of ODVA, Inc. All other trademarks are property of their respective owners.