An IPv6 Roadmap for EtherNet/IP

Jakub Korbel
Networks Architect
Rockwell Automation

Brian Batke
Engineering Fellow
Rockwell Automation

Filip Zembok
Principal Development Engineer, Embedded Software
Rockwell Automation

Presented at the ODVA
2025 Industry Conference & 23rd Annual Meeting
March 19, 2025
Clearwater Beach, Florida, USA

Abstract

As IT systems increasingly converge with OT (Operational Technology) environments, IPv6 readiness of protocols used in industrial automation becomes critical to ensure network-level interoperability in this broadening ecosystem. IPv6 is closing in on 50% of all Internet traffic per Google **[1]**. The growing proliferation of IoT devices, sensors, and interconnected machinery on the OT floor demands more address space, which IPv4 cannot provide. After many years with no clear user demand for on-premise IPv6 operation, the scales are shifting in OT applications, and the time has come for EtherNet/IP to adopt these new patterns.

This paper will lay out a framework and proposed timeline for work across SIGs that will allow vendors to deploy EtherNet/IP-based solutions in an all IPv6 or hybrid network infrastructure. These enhancements will cover not just use of the longer IP address for all CIP communications (including Security and Safety), but also propose enhancements to enable name-based operation as IPv6 addresses do not lend themselves well to either human use or device replacement use cases. Lastly, this paper will describe how IPv6 concepts and protocols of the IPv6 family can improve a user's device replacement and discovery experience.

Keywords

IPv6, IoT, IIoT, EtherNet/IP, CIP Safety, CIP Security, Device Replacement, IPv4

Definition of terms

Term	Definition
ACD	Address Conflict Detection [2] [3]
CIP	Common Industrial Protocol
CIPSE	CIP Specification Enhancement (see SE) of Volume 1, CIP [4].
Connection open message	One of the Forward_Open, Large_Forward_Open or Concurrent_Forward_Open request or reply, as defined in [4]
CPF	Common Packet Format [3]
CSSE	CIP Security Specification Enhancement (see SE) of Volume 8, CIP Security [5].
DAD	Duplicate Address Detection protocol based on Neighbor Discovery (see ND) protocol.
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
ESE	EtherNet/IP Specification Enhancement (see SE) of Volume 2, EtherNet/IP [3].
EtherNet/IP	EtherNet/Industrial Protocol, adaptation of CIP for Ethernet.
FQDN	Fully Qualified Domain Name
IANA	The Internet Assigned Numbers Authority
IP	Internet Protocol
ISP	Internet Service Provider
IT	Information Technology, i.e. hardware, software and communication technologies for general information processing
LAN	Local Area Network
LLDP	Link Layer Discovery Protocol
mDNS	Multicast Domain Name System
MTU	Maximum Transfer Unit
ND	Neighbor Discovery protocol, a part of IPv6 family
OT	Operational Technology, i.e. hardware, software and communication technologies that detect or cause a change through the direct monitoring or control of industrial equipment
OUNID	Originator's Unique **I**dentifier (CIP Safety)
ODVA PlugFest	ODVA event to check interoperability among EtherNet/IP devices
PMTUD	Path MTU Discovery [6]
SAN	Subject Alternative Name [7]
SE	Specification Enhancement. This is a document produced by SIG groups containing description of changes required to the CIP Networks Library Specification Volumes. SEs are identified by their IDs, which are composed like this: TYPE-VENDOR_ID-SERIAL_NUMBER, so that for example ESE-0001-089 means Enhancement to EtherNet/IP, from Rockwell Automation (Vendor ID: 1), this is the 89th ESE of the vendor.
SEND	Secure Neighbor Discovery
SIG	Special Interest Group. In ODVA context, SIG groups are responsible for preparing specification enhancements and proposing them to the TRB (see TRB).
SLAAC	Stateless Address Auto-Configuration
SNN	Safety Network Number (CIP Safety), 6 bytes of network identification.
TLV	Type, Length, Value – usually represents a filed in a data structure
TRB	Technical Review Board is a group responsible for reviewing technical standards, proposals, or issues to ensure quality and alignment with ODVA organization's goals and objectives.
TUNID	Target's Unique **I**dentifier (CIP Safety)
UCMM	Unconnected Message Manager
UNID	Device **Un**ique **Id**entifier (CIP Safety). It is composed of SNN (6 bytes) and Node ID (4 bytes)

Introduction

IPv6 is no longer an option. IPv6 adoption has become more widespread in IT systems [1]. Additionally, as IT and OT continue to merge, OT gradually becomes a part of IT rather than a separate entity [8]. This is further magnified by governments starting to mandate devices to be able to operate in IPv6-only environments as dual IPv4/IPv6 stack solutions were deemed overly complex [9].

In this situation, it is important that the CIP specification and especially the EtherNet/IP transport of CIP reflects the new requirements of regulators and customers.

This paper will discuss practical proposals for SIGs in every area of CIP specifications, laying out a timeline for work that needs to be done.

General Aim

As it was laid out in the previous paper for IPv6 introduction into the EtherNet/IP Ecosystem [10], there were several ways to incrementally introduce IPv6 into the system, such as by IPv4/IPv6 tunneling and IPv4/IPv6 translation and by temporary dual-stack solutions, but most of them proved too complex as reported by a US government memorandum from 2020 [9] calling for IPv6-only operation. Therefore, the specification work introducing IPv6 into CIP Networks Library and its volumes [11] should be organized to first cover the separate IPv6 use-case, but to allow dual-stack operation in cases of devices and software that need to communicate with both types of devices at once.

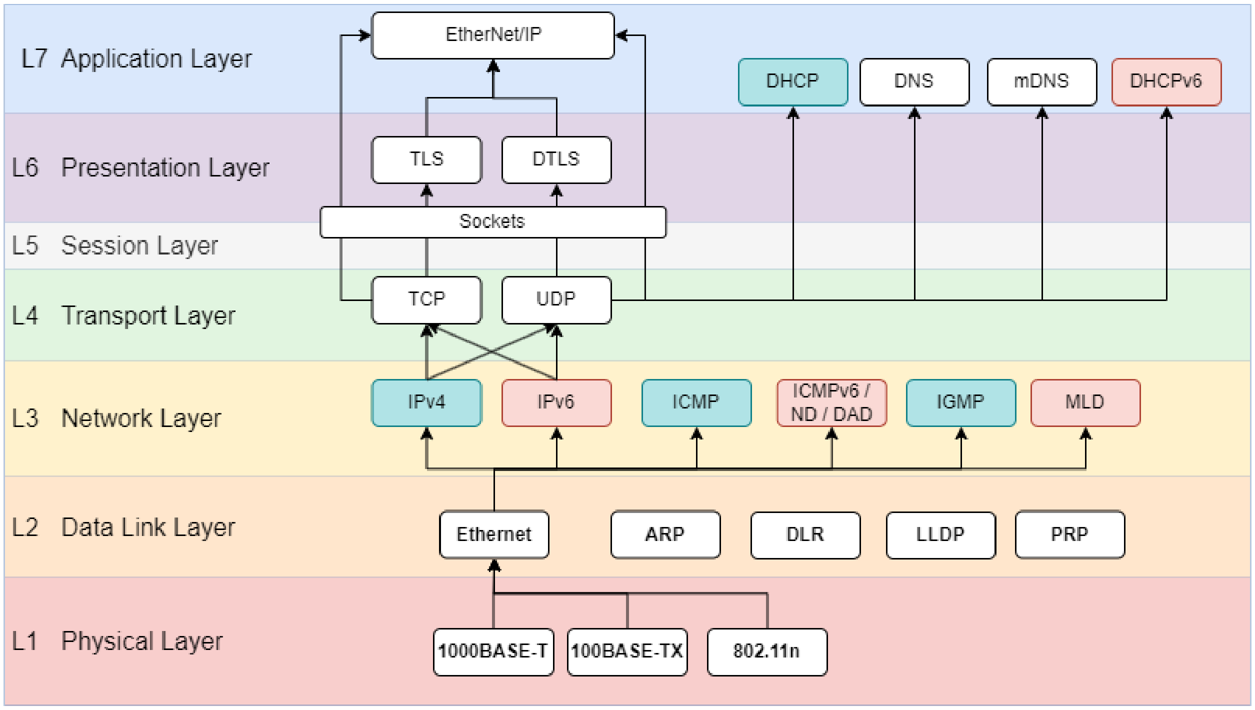


Figure 1 Dual Stack Design

The dual-stack solution can use a hybrid TCP/IP stack where both IPv4 and IPv6 protocol family is implemented. This mode of operation is generally available in most major embedded TCP/IP stacks on the market (such as: lwIP [12], Segger emNet [13], CycloneTCP [14], Treck [15], WindRiver VxWorks [16], and QNX [17]). In this mode, both families can be enabled at the same time, or one or the other family can be switched off during stack initialization to reduce potential attack surface in single-family deployments.

Apart from the communication itself and the availability of the technology, the next most vital aspect to define is the user experience of configuring dual and IPv6-only devices and using the new technologies in the OT environment.

A short discourse into why IPv6 addresses are (not) that simple

Here is a simple, soothing picture of an address:

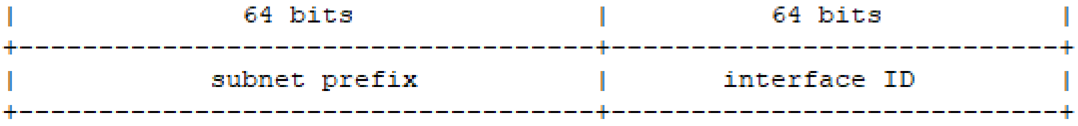
`192.168.1.10`

Here is a nightmare:

`2001:0004:130F:0001:0002:09C0:876A:130B`

Trying to type the address correctly can look equally intimidating to typing a serial key when prompted during a Windows XP installation in early 2000’s.

But is the address really to be afraid of? Have a look on the following picture, which we eventually get to after reading the IPv6 Addressing Architecture RFC [18].



```
|               64 bits              |         64 bits            |
+------------------------------------+----------------------------+
|             subnet prefix          |       interface ID         |
+------------------------------------+----------------------------+
```

Figure 2 IPv6 Addressing Architecture

According to the RFC, any unicast IPv6 address except for the IPv4 mapped addresses (which is even simpler) can be represented like this. This is quite simpler than IPv4 where the host (interface ID) part and subnet part could be very dynamic depending on the size of the mask.

Subnet Prefix

The subnet prefix essentially covers a hierarchy of top-level subnets. It is further partitioned by the ISPs and IT departments, so that globally unique subnets are achieved. It is of limited use for users and machine builders to care about it, as the devices automatically discover their prefix by reading the Router Advertisement message of ND. It can be used if communication among internal devices of several machines on the factory floor is desired. If not, IPv6 offers a fixed local-link subnet prefix fe80::/64, where the devices on the same link can implicitly communicate, so users can avoid having to think about subnet prefix very easily.

Interface ID

The Interface ID must be unique on the local subnet, so that when the full IPv6 address is composed of the Subnet Prefix and this number, it is also always unique. Apart from this, which is automatically checked by IPv6’s DAD, it is up to the users to decide what this 64-bit number will be.

This can be as simple as putting 1 for the first device from the left in the cabinet physically labeled “1”, 2 for the second one and so on. To address the chicken-egg problem of how to set these numbers without having an address in the first place, IPv6 employs a mechanism called EUI-64 [19], which composes this 64-bit number from a device’s mac address.

Such a device can then use this address to report itself without any manual configuration involved using ListIdentity and/or mDNS-SD with all the properties, labels and names it obtained from manufacturing and afterwards (rotary switches and more sophisticated methods could be employed though entirely vendor-specific).

Discourse conclusion

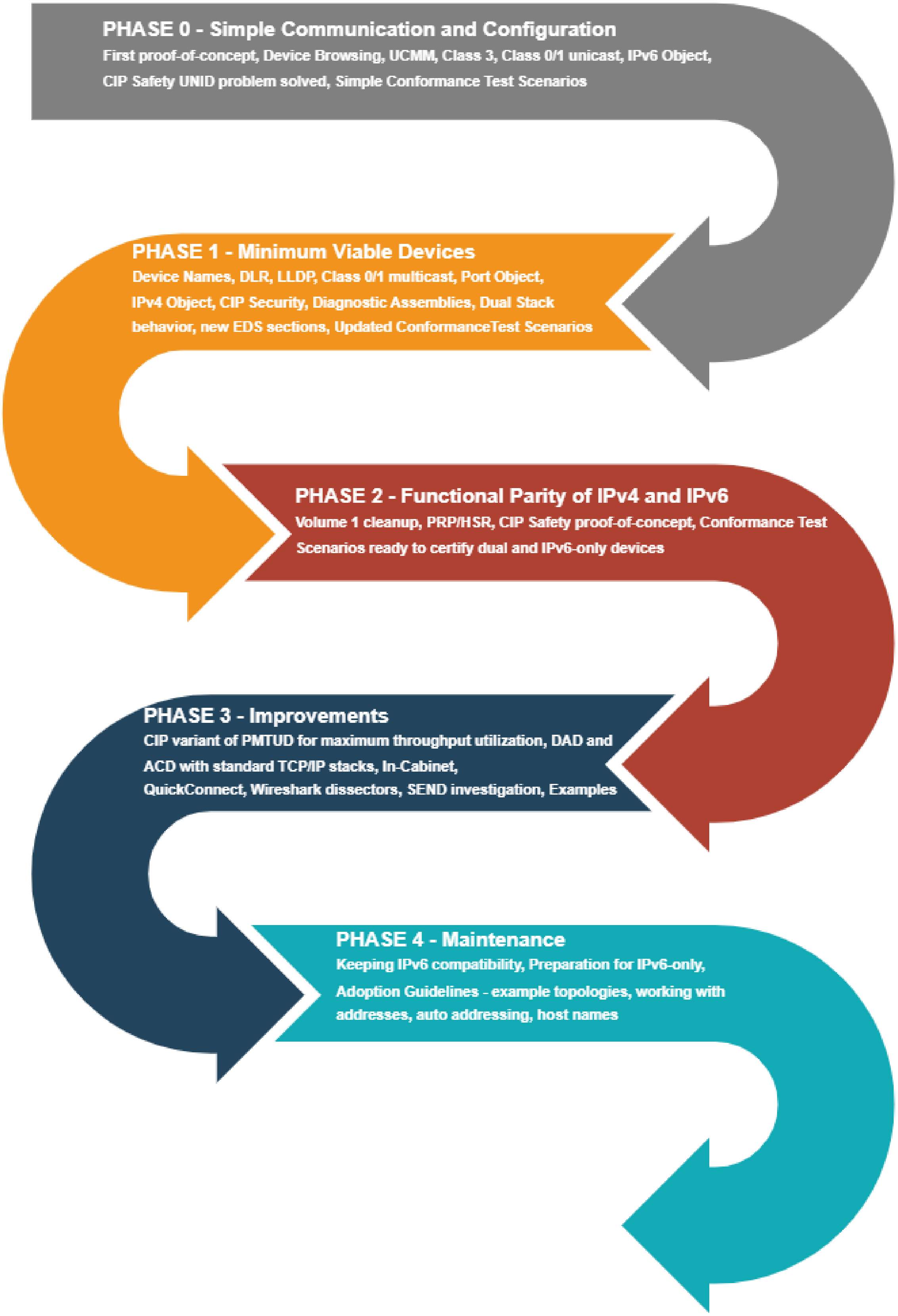
Compared to IPv4, working with longer IPv6 addresses presents usability challenges, however, the design of the IPv6 addressing is very well thought through. Additionally, other technologies from the IPv6 family can allow for host names and descriptions to be presented to the user out of the box during the discovery process, making device addressing comparably or even more convenient than with IPv4.

Timeline

The following timeline introducing IPv6 into CIP Networks Library volumes is laid out in phases, as the amount of work is significant and division into more manageable pieces of work is required. The sections of each phase propose an overall plan for partitioning and completing the specification work including prototyping and testing steps.

The timeline accounts for incremental improvements leading to having conformant and secure devices as soon as possible (end of Phase 1), allowing simpler devices to get conformant first. More specific technologies not required for all devices are considered in later phases. With the Cyber Resilience Act becoming effective in 2027 [20], security must not be omitted from the most basic viable device set (Phase 1).

The phases sections first lay out a list of changes for a specific CIP Networks Library volume and then elaborate more on complex individual items.



PHASE 0 - Simple Communication and Configuration

First proof-of-concept, Device Browsing, UCMM, Class 3, Class 0/1 unicast, IPv6 Object, CIP Safety UNID problem solved, Simple Conformance Test Scenarios

PHASE 1 - Minimum Viable Devices

Device Names, DLR, LLDP, Class 0/1 multicast, Port Object, IPv4 Object, CIP Security, Diagnostic Assemblies, Dual Stack behavior, new EDS sections, Updated ConformanceTest Scenarios

PHASE 2 - Functional Parity of IPv4 and IPv6

Volume 1 cleanup, PRP/HSR, CIP Safety proof-of-concept, Conformance Test Scenarios ready to certify dual and IPv6-only devices

PHASE 3 - Improvements

CIP variant of PMTUD for maximum throughput utilization, DAD and ACD with standard TCP/IP stacks, In-Cabinet, QuickConnect, Wireshark dissectors, SEND investigation, Examples

PHASE 4 - Maintenance

Keeping IPv6 compatibility, Preparation for IPv6-only, Adoption Guidelines - example topologies, working with addresses, auto addressing, host names

Figure 3 Timeline

Phase 0 – Simple Communication and Configuration

Possible deadline – third quarter of 2025 (calendar year)

In the early stages, it's crucial to identify the simplest and most common use cases quickly. This allows for the creation of initial proof-of-concept prototypes that demonstrate working real-life setups.

By generating these early proof-of-concept prototypes, SIGs can validate specification improvements in practical scenarios. This approach helps minimize the number of SEs, SE versions, and the back-and-forth interactions between SIGs and the TRB.

The specification enhancements in this phase do not cover Data Link Layer requirements (such as ACD, LLDP and DLR) and are designed to fit even user-space software stacks implementing EtherNet/IP without elevated privileges to control host system's TCP/IP stack and no access to raw socket layer or a tap interface. The creation of a proof-of-concept prototype at this phase must be as easy as possible.

The simplest device imaginable is discoverable on the network and can be contacted using UCMM without CIP Routing. Even with only UCMM, devices can be diagnosed and controlled. A mandatory set of EtherNet/IP communication objects needs to work. Class 3 and unicast Class 0/1 need no additional specification work, so they are initially listed in this phase too.

Phase 0 – EtherNet/IP System Architecture SIG

There is already an ESE in progress, ESE-0001-089. The EtherNet/IP SIG is the group to bring IPv6 into the specification.

Following components need to be specified at first:

- IPv6 Object to configure IPv6 addresses in the system (rationale in IPv6 Object section)
  - IP address and IP addressing method configuration (SLAAC and/or DHCPv6, manual, random, EUI-64)
- Device Discovery – ListIdentity
- Link Path update for binary addresses
- CIP UCMM
- CIP Class 0 and 1 (unicast only) connections
- CIP Class 3 connections
- Reference IPv6 Node requirements RFC 6434 [21] in the Mapping of Explicit and IO Messaging to TCP/IP section to describe the usage of protocols from IPv6 family
- Device Profiles section [3] update, including sets of mandatory objects for EtherNet/IP.

Proof-of-concept prototypes

Alongside specification change, ODVA member companies shall produce one or several proof-of-concept prototypes to validate a compliant device can be created and to limit the specification work in terms of implementation complexity. These proof-of-concept prototypes must present a device configuration use-case; hence a configuration tool must be a part of the work.

To ensure viability and interoperability of the proof-of-concept prototypes, a platform for device testing shall be created. This could be an interop event similar to ODVA Plugfest or ODVA Plugfest itself could be leveraged.

IPv6 Object

In the EtherNet/IP SIG meetings, the direction was set to not enhance the TCP/IP Interface object to allow for IPv6 addresses, as it contains multiple attributes specific to the IPv4 family of protocols, such as attributes related to ACD, and attributes related to contents of the IPv4 header. Instead, a new IPv6 object shall be created for IPv6 configuration. The continued existence of the TCP/IP Interface object remains questionable, this paper offers two alternatives:

1. a new IPv4 object that would be designed alongside the IPv6 object in later phases as if both objects had the same “parent class” (object-oriented programming term).
2. A document-based configuration of interfaces (discussed in the Document-based configuration section)

The new object needs to allow for the setting of the IPv6 address in both local and global scope, including a selection for interface ID assignment, such as Manual, DHCPv6, Random per RFC 4941 [22], Random per RFC 7217 [23], and EUI-64 [18].

Standard YANG model shall be considered while designing the IPv6 object [24].

Device Discovery

If the discovering device or tool wants to get all EtherNet/IP devices on the network, there are several challenges caused by the fact that IPv4-only, IPv6-only, and dual-stack devices are present on the network. List Identity would need to be sent twice, once on IPv4 global or local broadcast address and for the second time on the “IPv6 broadcast”, i.e. link-local all nodes multicast group at address ff02::1.

The SIG needs to decide how the devices represent themselves, whether they respond with address information per the request received (IPv4 information replied only to an IPv4 request and IPv6 information returned only to an IPv6 request or whether the devices reply with all the information via any of the requests).

The SIG can also consider other possibilities, such as specifying additional request data for ListIdentity requests providing filtering or querying capabilities. These could allow asking the network for “just IPv6-enabled devices”, “just the devices within this domain” or futuristic “just the devices with this neighbor”. Any selected option shall be analyzed for performance and impact on legacy systems.

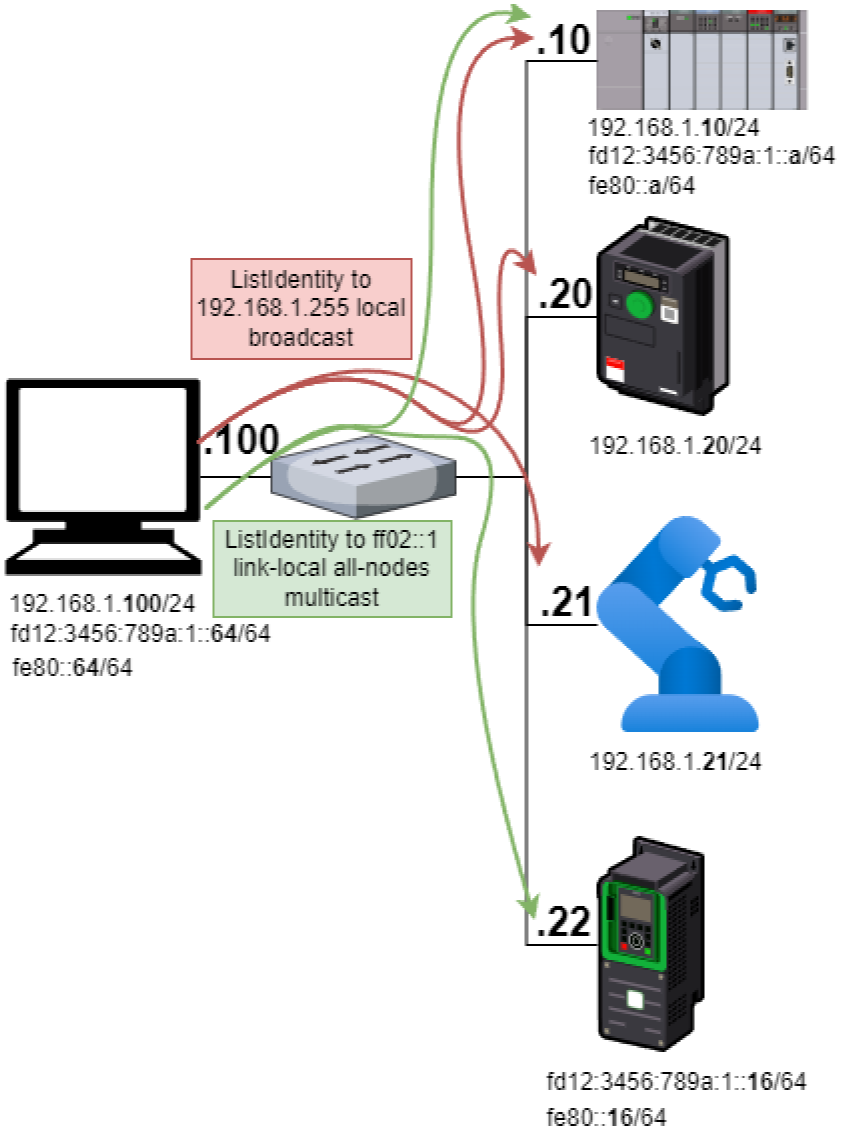


Figure 4 ListIdentity on a dual network

Current problems with List Identity are the coupling of CIP Identity information with a single IPv4 address and the lack of any human-readable device name. Possible solutions are to re-use T->O Sockaddr Info item (0x8001) for addresses (allowing for multiple being present) and define a new item with the device name information replacing or retaining the original CIP Identity item.

The SIG shall consider the possibility of mDNS-SD to find EtherNet/IP devices instead or alongside the ListIdentity browsing. mDNS-SD implementation is already in some EtherNet/IP devices, but mostly in client-only for Pull Model support in Volume 8 [5].

Link path update for binary addresses

EtherNet/IP Link path (defined in Connection Path section of Volume 2 [3]) as part of EPATH (defined in Volume 1 [4]) is currently defined as ASCII characters with the text of either an address or a host name, optionally including a port. This would result in extremely long connection paths whenever there is an IPv6 segment in the routing path.

This path is also present in the connection open request messages alongside configuration data both limited to only 510 bytes (255 words) of the path (if it can fit the UCMM message itself, which has at most 504 bytes of CIP payload per Volume 2 [3]).

Therefore, to be more efficient, a binary format is proposed in the ESE-0001-089. This format shall be required for IPv6 and dual-stack devices as it is also able to represent IPv4 address in a more concise way (right now it is up to 15 bytes for IPv4 dotted-decimal, it can go down to 6 bytes, where for IPv6 it can be as little as 18 bytes).

When host names are used throughout the system, the path too long problem becomes unavoidable. The SIG shall analyze options and use-cases, thinking about expanding message length limit or considering IP routing replacing CIP routing as with the increasing traction of Single-pair Ethernet, Ethernet becomes the dominant physical layer technology.

CIP Class 0 and 1

For phase 0 the specification does not need any changes as only unicast is supported. In unicast setup, the connection open messages do not have to contain Sockaddr Info items, which are not ready for IPv6. These will be defined in the next phase. If even basic class 0 and class 1 communication is to be required for phase 0 proof-of-concept prototypes is to be decided by the SIG, because unlike for the specification work, there is non-zero effort to convert the IO subsystem of EtherNet/IP stacks to be IP version agnostic.

Phase 0 – CIP System Architecture SIG

This section describes alternative ways to generally set and inspect configuration data with respect to the complex case of multiple dependent configuration values, which are found in IPv4 and IPv6 settings. It also mentions possible changes stemming from possible TCP/IP Interface Object update or replacement.

There is already a CIPSE in progress, CIPSE-0001-355. The CIP SIG in general shall mostly support changes required by the EtherNet/IP Volume changes, as CIP is independent from EtherNet/IP.

Although the CIP itself is not dependent on the IP address being 4 bytes, some changes ripple into Volume 1. Notorious problems of TCP/IP Interface object’s configuration, where atomic configuration change of related attributes (Configuration Method, Attribute #3 and Interface Configuration, Attribute #5) and obtaining pending and active configuration, can be solved by leveraging common CIP services for getting pending data and applying pending configuration at once.

TCP/IP Interface object is also possibly going to be replaced by one or more new objects. This has implications for Volume 1, where the TCP/IP Interface object is directly referenced in multiple places.

Following is a list of supporting features already included or to be included in the referenced CIPSE:

- Usage of document-based configuration, and/or:
- Standardization of Atomic Configuration changes as per Volume 8 including behavior description and state machines:
  - Begin_Config service
  - Kick_Timer service
  - Apply_Config service
- Standardization of getting active and pending values:
  - Get_Attribute_Single service containing extra payload to denote the desire of getting active, pending, or both values of an attribute
  - Get_Attribute_List extended to support the same
- Port Object Changes
  - New possible set of logical link objects and associated communication objects to include TCP/IP Interface object, IPv6 Object, or possibly IPv4 Object

Document-based configuration

To get a solid data-model for the new IPv6 configuration, a standardized YANG model [24] shall be used. When a standardized YANG model is used, investing into replicating it in CIP is a question for the EtherNet/IP and CIP System Architecture SIG to decide, with respect to the YANG model for network interfaces [25] considering future usage for Ethernet Link Object functionality.

With the evolution of the Pull Policy model specified in CSSE-0001-061 Pull CIP Security Config, the same approach could be leveraged extending the configuration schema to cover also the Interface configuration (regardless of IPv4 or IPv6).

The IPv6 Object could then be used alongside the document-based configuration, or it could be reduced to only:

1. show extra statistics regarding EtherNet/IP traffic on the specific addresses,
2. serve as a container object for Port representation, and maybe
3. show what was configured using the document as read-only information.

The SIG would then need to figure out how to present what is possible within the device to the tools wishing to configure the device (for example how many Ethernet interfaces it has, how many IP addresses it can configure on one Ethernet interface and so on) and how to approach error reporting.

Atomic Configuration Services

Volume 8 introduces configuration state-machines for most of its new objects [5]. The state-machines make sure that there is a configuration session first, and that all the changes are validated together and applied at once, which proved to be a very practical mechanism for configuration tools and devices alike.

The TCP/IP Interface object could use this concept to set multiple dependent attributes, and the mechanism could be standardized throughout the specification. There is already a standard service called Apply_Attributes (0x0D), which talks about pending and active values, but other important services are missing.

Moreover, one of the values is always hidden from the configuration tools. In Volume 8, it is the active value – the configured yet not applied value is visible to get services. In the TCP/IP Object, this is the pending value (for example Interface Configuration attribute #5), as the tools need to see the address in use. Hence the specification obviously lacks get service, which could reliably give the tool the value to be possibly applied and the value that is active. For that the CIPSE offers a solution using extensions of get services for use with the new IPv6 object, but also more generically.

Phase 0 – Conformance SIG

Communication with the Conformance Test Team shall be initiated early and facilitated by the Conformance SIG group. The team shall also initiate discussions on ODVA PlugFest scenario definitions to reflect the Phase 0 – EtherNet/IP System Architecture SIG section. It is important that the resulting changes in Conformance tests shall be developed and verified against the proof-of-concept prototypes, so that conformance tests are prepared to test new IPv6-enabled products.

Phase 0 – CIP Safety SIG

The CIP Safety SIG shall figure out how to approach the problem of a too-short UNID. The device Unique Identifier in CIP Safety specification consists of a 4-byte node ID and 6-byte Safety Network Number (SNN). For EtherNet/IP safety, the node ID is the IPv4 address of the devices and the SNN is assigned by the safety network configuration tool. The SNN is either auto-generated by the software tool or manually set by users. Refer to [26] for more details on the UNID definition. When migrating to IPv6, because 16 bytes of IPv6 address is much larger than 4 bytes of node ID in the safety UNID, the IPv6 address cannot be used directly to construct the UNID [10].

Increasing the size of the UNID to accommodate the IPv6 address is going to present a breaking change in the specification and possibly induce a review from the safety certification authority. There are opportunities to use different identifiers or distribute the identifiers on the network or parcel the UNID differently, for example using the Ethernet MAC or interface id address and lowering the size of SNN to 4 or 2 bytes, respectively. Another option would be to temporarily mandate only 4 bytes of the IPv6 local id to be used for IPv6 safety networks.

Phase 0 – CIP Motion SIG

1. No apparent IPv6 problems detected, a thorough check is needed by the SIG
2. IEEE 1588v2 Contains IPv6 support [27]
3. Transparent clock implementations need to be checked for behavior with IPv6 packets
4. Relationship with QoS clarified DSCP information is the same, it is just used differently by the implementors [28]

Phase 1 – Minimum Viable Device

Possible deadline – first quarter of 2026 (calendar year)

The goal of this phase is to allow a simple, yet full-featured and secure product to pass the conformance tests including all aspects of EtherNet/IP, especially requirements for Data Link Layer protocols, such as LLDP and DLR. Equivalence of DAD to ACD shall be achieved. CIP Security shall be considered and IPv4-only aspects of it shall be generalized (Ingress / Egress Object).

Phase 1 – EtherNet/IP System Architecture SIG

1. Neighbor Discovery and Duplicate Address Detection
2. DAD Conflict Reporting using IPv6 object
3. LLDP Data Table and processing of LLDP Management Address TLVs
4. Phasing out TCP/IP Interface Object
    a. Creation of IPv4 Object
5. Device Names
    a. Device name and host name mapping (Identity Object vs IP objects)
    b. Device host name publishing using DHCP Option 12 and 81
    c. Device host name publishing using DHCPv6 Option 39 [29]
    d. Device host name publishing using ListIdentity Reply
    e. (m)DNS used on link-local addresses to publish device host name
6. Multicast usage and allocation defined for IPv6 Forward Opens and IPv6 object (ESE-0243-013)
7. DLR object, DLR protocol
8. Mandatory Port Object if IPv6 is used
9. Proof-of-concept improvements to reflect specification work

ND and DAD guidelines for IPv6

A section in Volume 2 must be provided covering connecting a device to LAN. No specific EtherNet/IP constraints shall be imposed on Duplicate Address Detection allowing for standard implementation using off-the-shelf TCP/IP stacks.

The IPv4 ACD mechanisms as required by Volume 2 often prohibit such implementation leading to custom bypassing logic in EtherNet/IP products increasing the cost of EtherNet/IP adoption. Whether tweaks of DAD state machine or timing are necessary for industrial networks shall be investigated in later phases.

LLDP

LLDP Data Table object explicitly ignores all non-IPv4 Management Address TLVs [30] present in LLDP messages. This object must be updated to provide the IPv6 list as well. LLDP protocol's Management Address TLV already accounts for various address families as defined by IANA already, including IPv6 [31].

IPv4 Object

As the TCP/IP Interface object will require changes because of the desired generalization of IP address usage, there is an opportunity to redesign it, either by updating the object or obsoleting it in favor of a new IPv4 object designed per the IPv6 counterpart keeping a vague motion of inheritance, that would anyways need to be represented in dual stacked implementations.

This object could benefit from the Atomic Configuration Services designed to support the IPv6 object and the same YANG data model could be used [24].

Device Names

When building a system full of EtherNet/IP devices on a local network (e.g. a machine), there is a problem of identifying which real device has been connected and shall be configured to perform a specific task. Later on, there is a problem with locating the device. For these use-cases, the solution could be to use device labels. Historically, setting IPv4 address using a physical method (e.g. using screwdriver on rotary switches) was a popular solution. The IPv6 can allow the same "comfort" if the Interface ID is set the same way, but there are opportunities to make the experience better.

The benefit of IPv6 is that all devices are ready to talk on the network immediately with the use of unique link-local addresses (although at this point without other protocol support, the only device identifying information is the MAC address). To match the talking agents with the real devices, ListIdentity can be leveraged to get identifying information about serial numbers and other CIP Identity information, but currently present information does not allow for user-specified labels. Commissioning of these user-specific labels can stay vendor-specific.
. There are several options to consider:

1. Using host names as the method for device naming, which brings consistency but does not allow for more elaborate location information
   a. Moving host names under CIP Identity Object as it is a system feature, while considering overrides at IPv6 and IPv4 objects per existing technology: dhclient's (a widely used Linux DHCP client) network interface specific send-hostname [32]
   b. Making host names visible in ListIdentity reply
   c. Assigned_Name (Identity Instance Attribute #15) and Assigned_Description (Attribute #16) remain unrelated to host names.
2. Using Assigned_Name (Attribute #15) and Assigned_Description (Attribute #16), which are currently optional in the Identity Object.
   a. Mapping Assigned_Name to host name. Assigned_Name's data type is STRINGI possibly containing multiple translations of the same string and non-ASCII characters, which might present serious mapping challenges
   b. Making Assigned_Name mandatory, or mandatory only for IPv6 devices
   c. Making Assigned_Name (and Assigned_Description) visible in ListIdentity reply
3. Using unique Interface IDs and keeping names as optional CIP Attributes (similar to current state with IPv4). Host names and Identity Object Attributes remain optional. With IPv6, these can be obtained via CIP and set using a vendor-specific method. These names could be also stored in a tool and not present in the devices.

If host names are selected as the labeling method, workflows must be defined for:

1. Host name configuration

2. Equivalent host name and IPv4/IPv6 address usage in the system
3. Device replacement

The devices currently are not mandated to publish assigned host names using DHCP Request options, so the only currently defined possibility is to configure DNS manually. The SIG must explore options to remedy the problem. There are several possible directions (and more can be uncovered):

1. Using DHCP Option 12 to consistently publish device's host name and then considering the FQDN sent from DHCP server in option 81, as defined in the RFC 4702 [33], so that a combined DNS and DHCP server can respond to DNS queries about newly connected devices.
2. Device host name publishing using DHCPv6 Option 39 [29], so that a combined DNS and DHCPv6 server can respond to DNS queries about the host name.
3. Using mDNS, so that the device is immediately reachable using its host name under ".local" domain and responds itself to the mDNS queries.

CIP Class 0 and 1 multicast

As noted in the previous IPv6 paper [10], the connection open messages contain optionally the Sockaddr Info items, which currently have only IPv4 family defined and are not large enough to hold equivalent contents of IPv6 family members. Even though the direction was preliminarily discussed during EtherNet/IP SIG meetings, and a decision shifted towards defining different CPF item IDs, because of how much this proliferates to the other parts of the specification [3], it might be worth re-considering, especially when the current items 0x8000 and 0x8001 already contain a distinguishing field called sin_family. Implementation-wise, the change made on the item ID vs. sin_family is not significant, while it will reduce the number of changes needed in Volume 2, covering all usages implicitly.

What the original paper did not consider specifically, is the allocation method for the multicast addresses that can be embedded inside the Sockaddr Info items, which is embedded in the TCP/IP Interface Object. There is already an ESE-243-013 to remedy this issue, but it appears to be abandoned and not reflecting possible changes in the CIP objects.

DLR Protocol, DLR Object

The DLR Protocol uses 4-byte IPv4 address in every DLR message now. The protocol must be extended to account for IPv4-only, IPv6-only and dual-stack devices on the network. One possibility would be to borrow the concept from LLDP Management Address TLV that contain Address Family as defined in [31], but other options, such as using device names or any available information, can be evaluated by the SIG.

The DLR Object must be refactored according to the protocol changes, changing the representation for Supervisor and Gateway addresses.

Mandatory Port Object

The TCP/IP Interface Object contains an attribute which provides a path to the object representing the physical link, Physical Link Object (Instance Attribute 4) [3].

There is the Port Object in Volume 1 [4], which contains an association array of paths. Removing the TCP/IP Object's Physical Link Object instance attribute #4 and mandating Port Object and using its Associated Communication Object's instance attribute #11 instead could better model the relationships for IPv4 and IPv6, especially when the IPv6 naturally brings two addresses, one in the link-local scope and typically also one in the global scope whenever it needs to communicate to a different link.

Phase 1 – EtherNet/IP SIG – CIP Security Subgroup

Even though not many parts of Volume 8 [5] directly reference IPv4, there are some places that must be updated.

1. Ingress Egress Object changes for IPv6
2. SAN clarifications for IPv6
3. Removed references to IPv4 and TCP/IP Interface Object

Ingress Egress Object

Inside the Ingress and Egress rules, IP address range is defined as IPv4. This is insufficient for IPv6-only and dual-stack setups. The rule format does not account for IPv6 addresses, so it must be updated. Which guidance to provide to the device vendors regarding rule buffer capacity increase shall be discussed in the subgroup. With host names being used instead of IP addresses, implications to security configuration must be considered.

SAN clarification for IPv6

When using host names throughout the system, certificates issued for a specific host name can be used in IPv6 and IPv4 without any limitation. If instead the device certificate relies on IP addresses, all the addresses the device wishes to be reachable on must be listed with the usage of the SAN extension as defined in [7].

Phase 1 – CIP System Architecture SIG

The CIP SIG shall support the EtherNet/IP SIG work in following tasks:

1. Depending on the direction, possible initiation of obsolescence of TCP/IP Interface Object
2. Device name and host name mapping (Identity Object vs IP objects)
3. Mandatory Port Object if IPv6 is used

Phase 1 – CIP Safety SIG

1. CIP Safety proof-of-concept prototype
2. Removed references to TCP/IP interface object and IPv4

CIP Safety Proof-of-concept

CIP Safety proof-of-concept prototype with IPv6 shall be created by one or more of the member companies, facilitated by the CIP Safety SIG. This proof-of-concept prototype shall demonstrate the usage of the newly specified UNID mechanisms.

Phase 1 – Conformance SIG

A new set of requirements resulting from specification updates shall be composed and discussed with the Conformance test team for the roadmap on Conformance test (CT) suite and the ODVA PlugFest scenarios suite update. It would be beneficial to test engineering builds of the CT suite against engineering builds of the proof-of-concept prototypes.

Phase 2 – Functional Parity of IPv4 and IPv6 possibly without In-Cabinet and QuickConnect

Possible deadline – fourth quarter of 2026 (calendar year)

This is the final stage ensuring all aspects of the original specification are projected into its IPv6 version, including conformance testing. The only possible exceptions are In-Cabinet use-cases (mainly LLDP commissioning), where the 10BASE-T1S subnets are isolated, and QuickConnect, for the same reason.

Phase 2 – EtherNet/IP System Architecture SIG

1. PRP/HSR Nodes Table
2. PRP/HSR and DAD
3. Diagnostic Assembly Connection Points
4. Defined behavior for dual stacks
5. EDS sections for the new objects
6. Proof-of-concept improvements

PRP/HSR

PRP Node Table Object contains IP address in form of 4 bytes. This needs to be changed for IPv6 nodes.

IPv6 and its DAD can on itself remedy the ACD problem of the PRP-ACD combination as discussed in [3] as it does not operate on Data Link Layer. IPv6-only environment is thus more friendly towards PRP, and future installations shall benefit from it.

Diagnostic Assembly Connection Points

There is a defined diagnostic aggregation mechanism for TCP/IP Interface Object’s packet counters. With the usage of IPv4 and IPv6 object, the SIG must decide on the solution. Because the counters cover layers above the Network Layer (TCP and UDP), it might make sense to either report IPv6 and IPv4 separately or to aggregate the counters to one.

Phase 2 – EtherNet/IP SIG – CIP Security Subgroup

As the proof-of-concept prototypes mature and with the updated revision of the Ingress/Egress Object, CIP Security Interop scenarios shall consider IPv6 while creating secure zones.

Phase 2 – CIP System Architecture SIG

The CIP SIG in this phase must ensure there are no loose ends and possible misinterpretations of the specification and that no section of Volume 1 [4] requires IPv4 or any obsolete object. Where possible and wherever communication examples reference IPv4, and IPv6 example shall also be presented.

1. No object directly references IPv4
2. No Identifiers are derived from IPv4
3. Supporting CIPSE for changes in other volumes of the CIP Networks Library [11]
4. Examples with IPv6

Phase 2 – CIP Safety SIG

The CIP Safety SIG shall remove references to TCP/IP interface object and IPv4, including ACD references from the specification depending on the UNID assignment method selected.

Phase 2 – Conformance SIG

The Conformance SIG shall make sure all conformance test suites are fully defined, including CIP Safety and CIP Security, so that any EtherNet/IP device could be tested via IPv6 and get a certification.

Phase 3 – Improvements

Possible deadline – first quarter of 2027 (calendar year)

This phase aims to capitalize on the benefits of IPv6 and concepts invented alongside this protocol as well as finished definition for less urgent technologies.

Phase 3 – EtherNet/IP System Architecture SIG

1. QuickConnect and DAD
2. In-Cabinet objects
5. In-Cabinet LLDP Commissioning TLV
1. ACD tweaks within standard stacks
2. DAD tweaks within standard stacks
3. Device Profiles Examples with IPv6
4. Wireshark dissector updates for the new objects
5. Wireshark dissector updates for protocol changes (DLR)

ACD and DAD tweaks, QuickConnect

Volume 2 defines very specific timing and count constants for the ACD process [3]. The ACD process can be also enabled or disabled, per the configuration of the TCP/IP Interface object. The ACD behavior also differs greatly with whether the QuickConnect technology is enabled or not. This complicates EtherNet/IP adoption, because standard TCP/IP stacks defined per the ACD RFC [2] often fail to fulfill the requirements.

The current ACD requirements must be analyzed and a path forward must be decided for what to do with the discrepancies to RFC 5227 [2]. Moreover, DAD shall be analyzed for compatibility and applicability to industrial settings including the QuickConnect features, with the general aim to harmonize with current internet standards if possible.

In-cabinet

The In-Cabinet Commissioning object is now only allowing IPv4 addresses inside the Reference Position Attribute structure [3]. A new revision shall be created to allow for IPv6-only setups. It is questionable whether a dual setup is applicable in this case as the commissioning node attempts to commission all devices in the cabinet.

The LLDP Commissioning Request TLV needs to be updated as well to allow for larger addresses, different objects and potentially host names. A directional decision needs to be made.

Because of the nature of needed In-cabinet Gateway the devices are typically behind; this change could be done in a later phase. In an IPv6-only environment where the gateway is replaced by a simple link converter device, the need for IPv6-only in-cabinet definition grows.

Phase 3 – EtherNet/IP System Architecture SIG – CIP Security Subgroup

SEND protocol investigation shall be performed. The state of the technology per IPv6 Node Requirements RFC [21] is not ready for widespread SEND adoptions yet: “While there have been some implementations of SEND, there has been only limited deployment experience to date in using the technology.“ Whether or not to mandate SEND usage in EtherNet/IP needs to be analyzed and decided by the SIG.

Phase 3 – CIP System Architecture SIG

1. PMTUD instead of artificial limits for traffic
2. Concepts of PMTUD allows for Maximum transport utilization messaging of a given path

PMTUD

The Path MTU Discovery is a dynamic concept of getting the maximum MTU of an ever-changing path between Node A and Node B in the IPv6 world. It works by attempting to send the largest non-fragmented IPv6 packet the Node A can process and waiting for ICMPv6 message Packet Too Big, reporting which node couldn't handle the size and which MTU it can handle. This procedure can dynamically assure that Node A utilizes full capacity of the link chain (path).

While this does not appear to be interesting to EtherNet/IP and CIP, similar concept can be employed to get rid of artificial UCMM limits imposed by minimal amount of data sent over any CIP transport (currently driven by the maximum size on ControlNet) [3]. As CIP paths tend to be static, the size can be calculated before Node A sends traffic and it can be changed only when the path or a device on the path changes. The information about maximum amount of data transferred over a specific path could enable CIP ecosystem to utilize the maximums.

Phase 4 – Maintenance of Compatibility
Possible deadline – Later than the first quarter of 2027 (calendar year)

Phase 4 – All specification SIGs

All CIP SIGs shall repeat the following question: “What about IPv6?” over and over, thinking about opportunities that IPv6 brings to the ecosystem. Providing examples of IPv6-based topologies, benefits of host names and possible simplification for machine builders and system integrators shall be considered. Continuation of conformance testing and temporary extension of Plug Fest activities could help in overall IPv6 and EtherNet/IP adoption.

Conclusion

The necessity of supporting IPv6 in EtherNet/IP is undeniable in the face of regulatory pressures, the convergence of Operational Technology (OT) and Information Technology (IT), and the shift towards IPv6-only network strategies in IT. The ODVA community must now focus on the hows and whens, following the strategic roadmap put forth in this paper.

First, it suggests bringing simpler, yet fully conformant devices to existence. Next, optional features shall be gradually added in. For clarity, CIP Security is not considered optional as European Union's Cyber Resiliency Act becomes effective in December 2027 [20]. Lastly, the specification work shall leverage IPv6 protocol family benefits and concepts, such as PMUTD, transitioning eventually to the maintenance phase.

This paper also identifies complex problems, such as too small CIP Safety UNID or SEND protocol usage in the system, which need to be further analyzed, and a cross-business consensus needs to be found.

IPv6 not only brings new tasks and challenges but also offers solutions to long-standing issues such as address exhaustion, complex subnetting with NAT, and PRP and ACD coexistence. Moreover, IPv6 provides future opportunities that can enhance the capabilities and performance of network infrastructures. Embracing IPv6 and making EtherNet/IP IPv6-ready can serve as another market differentiator, showcasing its commitment to innovation and future-proofing its technologies.

References

[1] Google, Inc., "IPv6 – Google," 17 September 2024. [Online]. Available: https://www.google.com/intl/en/ipv6/statistics.html. [Accessed 6 January 2025].

[2] Apple, Inc., " IPv4 Address Conflict Detection," July 2008. [Online]. Available: https://datatracker.ietf.org/doc/rfc5227/. [Accessed 6 January 2025].

[3] ODVA, The CIP Networks Library, Volume 2: EtherNet/IP Adaptation of CIP, Ann Arbor: ODVA, Inc., 1999-2025.

[4] ODVA, The CIP Networks Library, Volume 1: Common Indutstrial Protocol, Ann Arbor: ODVA, Inc., 2001-2025.

[5] ODVA, The CIP Networks Library, Volume 8, CIP Security, Ann Arbor: ODVA, Inc., 2015-2025.

[6] J. McCann, S. Deering, J. Mogul and R. E. Hinden, "Path MTU Discovery for IP version 6," July 2017. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc8201. [Accessed 6 January 2025].

[7] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile," May 2008. [Online]. Available: https://www.rfc-editor.org/rfc/rfc5280#section-4.2.1.6. [Accessed 6 January 2025].

[8] A. S. George, "The Impact of IT/OT Convergence on Digital Transformation in Manufacturing," *Partners Universal International Innovation Journal,* vol. 02, pp. 18-38, 2024.

[9] R. T. Vought, *Completing the Transition to Internet Protocol Version 6 (IPv6),* Washington. D.C., 2020.

[10] D. XU, Y. YU, P. Brooks and B. Batke, "EtherNet/IP over IPv6 - Evolution, not Revolution for the World's Leading Industrial Ethernet Variant," in *2012 ODVA Industry Conference & 15th Annual Meeting*, Stone Mountain, 2012.

[11] ODVA, Inc., "Specifications | ODVA Technologies | Network Specifications," [Online]. Available: https://www.odva.org/subscriptions-services/specifications/. [Accessed 20 February 2025].

[12] A. Dunkels and L. Woestenberg, "lwIP: Overview," 17 June 2018. [Online]. Available: https://www.nongnu.org/lwip/2_1_x/index.html. [Accessed 6 January 2025].

[13] SEGGER Microcontroller GmbH, "IPv6 The next generation internet protocol," [Online]. Available: https://www.segger.com/products/connectivity/emnet/technology/ipv6/. [Accessed 6 January 2025].

[14] Oryx Embedded, "CycloneTCP Embedded IPv4 / IPv6 Stack," 6 September 2024. [Online]. Available: https://www.oryx-embedded.com/products/CycloneTCP.html. [Accessed 6 January 2025].

[15] Treck, Inc., "Treck IPv4/v6 Dual Stack Datasheet," 2022. [Online]. Available: https://treck.com/treck-ipv4v6-dual-stack-datasheet/. [Accessed 6 January 2025].

[16] Wind River Systems, Inc., "VxWorks Datasheet," [Online]. Available: https://www.windriver.com/resource/vxworks-datasheet. [Accessed 6 January 2025].

[17] BlackBerry Limited, "QNX - Protocols," December 2024. [Online]. Available: https://www.qnx.com/developers/docs/8.0/com.qnx.doc.neutrino.io_sock/topic/protocols.html. [Accessed 6 January 2025].

[18] R. Hinden and S. Deering, "IP Version 6 Addressing Architecture," February 2006. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc4291. [Accessed 6 January 2025].

[19] S. Thomson, T. Narten and T. Jinmei, "IPv6 Stateless Address Autoconfiguration," September 2007. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc4862. [Accessed 6 January 2025].

[20] European Union, "Cyber Resilience Act," 23 January 2025. [Online]. Available: https://digital-strategy.ec.europa.eu/en/policies/cyber-resilience-act. [Accessed 20 February 2025].

[21] E. Jankiewicz, J. Loughney and T. Narten, "IPv6 Node Requirements," December 2011. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc6434. [Accessed 6 January 2025].

[22] T. Narten, R. Draves and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6," September 2007. [Online]. Available: https://datatracker.ietf.org/doc/rfc4941/. [Accessed 6 January 2025].

[23] F. Gont, "A Method for Generating Semantically Opaque Interface Identifiers with IPv6 Stateless Address Autoconfiguration (SLAAC)," April 2014. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc7217. [Accessed 6 January 2025].

[24] M. Bjorklund, "A YANG Data Model for IP Management," March 2018. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc8344. [Accessed 6 January 2025].

[25] M. Bjorklund, "A YANG Data Model for Interface Management," March 2018. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc8343. [Accessed 6 January 2025].

[26] ODVA, The CIP Networks Library, Volume 5: CIP Safety, Ann Arbor: ODVA, Inc., 2005-2025.

[27] G. M. Garner, "IEEE 1588 Version 2 - Summary," 24 September 2008. [Online]. Available: https://www.ieee802.org/1/files/public/docs2008/as-garner-1588v2-summary-0908.pdf. [Accessed 20 February 2025].

[28] K. Nichols, S. Blake, F. Baker and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers," December 1998. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc2474. [Accessed 6 January 2025].

[29] Cisco Systems, Inc., "The Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Client," October 2006. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc4704. [Accessed 6 January 2025].

[30] Institute of Electrical and Electronics Engineers, Inc., Station and Media Access Control Connectivity Discovery (IEEE Std 802.1AB-2009), New York: IEEE, 2009.

[31] IANA, "Address Family Numbers," 22 December 2023. [Online]. Available: https://www.iana.org/assignments/address-family-numbers/address-family-numbers.xhtml. [Accessed 6 January 2025].

[32] die.net, "dhclient.conf(5) - Linux man page," [Online]. Available: https://linux.die.net/man/5/dhclient.conf. [Accessed 6 January 2025].

[33] M. Stapp, M. Volz and Y. Rekhter, "The Dynamic Host Configuration Protocol (DHCP) Client Fully Qualified Domain Name (FQDN) Option," October 2006. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc4702. [Accessed 6 January 2025].

[34] International Telecommunication Union (ITU), "X.200 : Information technology - Open Systems Interconnection - Basic Reference Model: The basic model," 1 July 1994. [Online]. Available: https://www.itu.int/rec/T-REC-X.200-199407-I/en. [Accessed 6 January 2025].

****

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.