PYRAMIDSOLUTIONS

VISIONARY SOLUTIONS ▲ EXCEPTIONAL RESULTS

12 Critical Pre-Implementation Steps

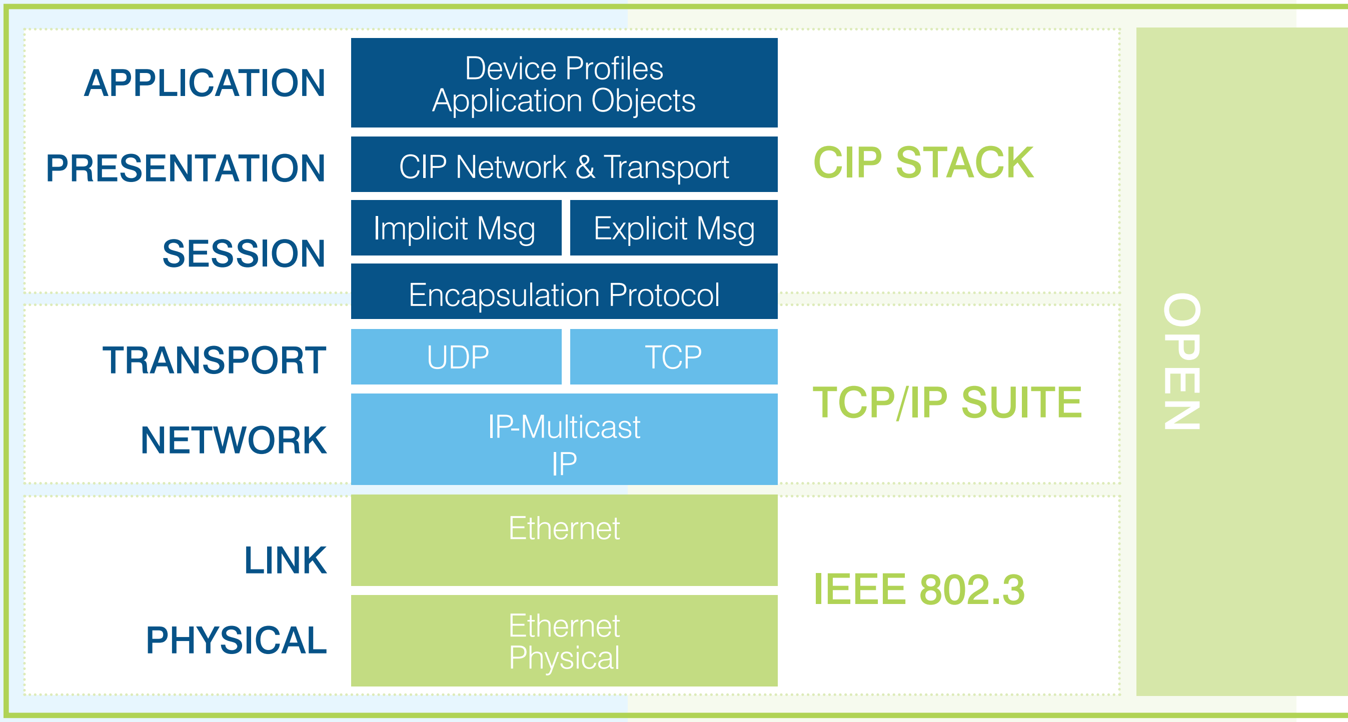
for a Successful EtherNet/IP Project

INTRODUCTION

The rise of smart factories and the Industrial Internet of Things (IIoT) has caused industrial automation companies with legacy infrastructures to explore networks that support Ethernet/IP (Ethernet Industrial Protocol) due to its speed, determinism and seamless integration. **EtherNet/IP runs on standardized Ethernet and standard TCP/IP**, allowing it to facilitate internet communications.

Open standardized Ethernet, a networking protocol used to connect wired networks and allow devices to communicate via a communication protocol, was first established by Bob Metcalfe and others in 1980. Its adoption has increased dramatically due to its low cost, reliability and consistent improvements to the physical layer and network speed over the years.

TCP/IP is the base protocol for Industrial protocols that run on Ethernet, including DeviceNet, PROFINET and EtherNet/IP. It's a protocol stack that combines TCP and UDP messaging and governs the connection of devices to the Internet. EtherNet/IP leverages TCP/IP to build messages and send them to nodes on the Ethernet network.



The fact that EtherNet/IP is so widely adopted (the most popular of all the industrial network protocols) has proven to aid in greater business agility and sustainability in ever-changing manufacturing technology and industry standards. Not only does EtherNet/IP have the ability to collect information in real-time from devices, but it uses this monitoring capability to control devices, increasing accuracy, efficiency and safety across your enterprise.

If you've determined that EtherNet/IP is the right protocol for your systems and products, there are a few things you should consider. Conformance testing is often an after-thought for new initiatives, however this is something that should be planned out before rolling out the solution. Luckily, EtherNet/IP is an ODVA-certified protocol that is highly-standardized, so conformance and testing are easily accomplished with the right tools as part of your development project. Additionally, you should know your objectives and product requirements before deploying EtherNet/IP in your unique environment. Below is a list of lesser-known yet crucial pre-implementation steps that will help you capture and narrow down the scope of your EtherNet/IP project and capitalize on your schedule and budget.

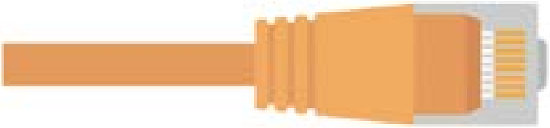
1. DETERMINE IF YOU NEED TO ENLIST THE HELP OF AN ETHERNET/ IP VENDOR.

The right partner can also provide products along with consulting and development services to further accelerate your development effort.

If you've never implemented, designed, developed or tested networked devices or EtherNet/IP before, you should consider partnering with a vendor that is an EtherNet/IP expert and offers field-proven and conformant EtherNet/IP products and solutions– one that can help you navigate the thousands of pages of ODVA specifications. Leveraging a technology partner that understands your business objectives and will work alongside your engineering team throughout the entire project will accelerate implementation and guarantee conformance. The right partner can also provide products along with consulting and development services to further accelerate your development effort.

2. ALLOCATE THE CIP AND ETHERNET/IP SPECIFICATIONS AND A VENDOR ID FROM ODVA.

This is required under the ODVA Terms of Usage. The CIP (Common Industrial Protocol) and EtherNet/IP specifications are the definitive sets of information for these protocols. Field-proven EtherNet/IP protocol stacks aid in implementation and provide the necessary features and functionality so there's no need to develop the protocol stack from scratch and risk conformance issues when connecting to other network devices.



3. DETERMINE THE CLASS OF YOUR ETHERNET/IP DEVICE OR APPLICATION.

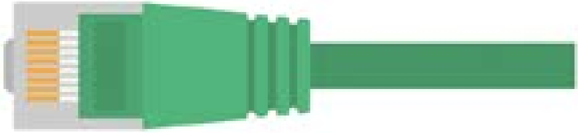
Based on your business requirements and project objectives, decide if you need your devices to be a scanner(s) (originator or client of connections), adapter(s) (target or server of connections) or both. Ask yourself the following questions to determine what classes you'll need in your EtherNet/IP solution:

- Is there any point in the process where you need a scanner to open an implicit connection with the adapters?
- What is the intent of your device and what features do you need? For example, if you only need to access the collected data every hour, you may only need an adapter or series of adapters to send explicit messages when you need them. If you need data to be accessible every 10 milliseconds, you will want to use a scanner class device to ease I/O messaging originated by your PLC or controller.

4. DETERMINE THE INTENDED NETWORK VIEW OF YOUR DEVICE.

A significant part of your development effort will be in determining the network view of your device. This is generally called the "device profile," which is basically a map of your device on the network. Your device may support a standard profile already defined by the ODVA, a unique profile with vendor-specific objects, or a combination of both.

This step in the process defines the type of messaging and connections your device will support and what data will be accessible from your device via the network. This includes the application object definition for your device as well as all other objects. Your objects are your storehouses for data (provided by the device, collected by the device and for configuration and diagnostics), so it's important to define your device's object model early on in your development process. For example, your device will require a set of mandatory objects and may expose assembly objects used for implicit (I/O) messaging via a PLC originated connection as well as objects for configuration and diagnostics.



5. PURCHASE OR BUILD AN ETHERNET/IP PROTOCOL STACK.

Developing an EtherNet/IP protocol stack can take months to years. Licensed EtherNet/IP Stacks, such as Pyramid Solutions' NetStaX EtherNet/IP Scanner and Adapter Stacks can significantly reduce your development and testing efforts and will help ensure that your device will pass ODVA Conformance and Interoperability testing.

6. DEVELOP AN EDS FILE FOR YOUR DEVICE.

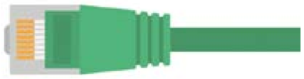
The EDS file, which ingests and provides basic information on your device such as identity and data, provides connected point information and default sizes of the adapter or target.

An EDS (Electronic Data Sheet) is a human-readable file that contains device specific information including company, ODVA Vendor ID, device identification, software version, device parameters and connection points.

A properly formatted EDS file is necessary to pass ODVA conformance, but also will allow your device to properly work with tools that use EDS files for configuration. The EDS file, which ingests and provides basic information on your device such as identity and data, provides connected point information and default sizes of the adapter or target. Without a correctly configured EDS file, your connection is much more likely to fail since it will pull inaccurate information from the network. The EDS file prevents users from having to guess the instance or size of the target. Even though it's required by ODVA, the EDS file also makes the connection faster and smoother.

7. DETERMINE YOUR TARGET PLATFORM.

When developing a device, the selection of the Hardware platform and operating environment is critical in meeting your EtherNet/IP requirements for the product. Will you be utilizing a small microcontroller or a larger hardware platform? If your target platform does not have the horsepower or resources to do what you've defined as its intention, additional development may be required, which would increase your costs and extend your project timeline. Will you target a Windows, Linux, or bare-metal embedded environment to best match your operational needs?



8. VERIFY THE AVAILABLE RESOURCES FOR YOUR PLATFORM.

The EtherNet/IP stack will consume FLASH, RAM and processing power from your device. RAM requirements will increase depending on the number of simultaneous connections and messages your device will support. The EtherNet/IP stack must be executed reliably and in a timely fashion so connections are maintained and response time is acceptable. It's also important to keep in mind that your runtime size and requirements are different when you're developing and debugging. This may require that you develop and troubleshoot your EtherNet/IP solution on a higher-end version of the microprocessor with more storage, and then switch to a lower-end version before you deploy into production.

9. VERIFY THE REQUIRED TOOLS AND ENVIRONMENT TO DEVELOP ON YOUR PLATFORM.

ODVA tools such as the Conformance Test Software and EZ-EDS will greatly improve conformance and interoperability. You may also consider third-party network or EtherNet/IP-specific tools, such as ones to debug, simulate and test for interoperability. These can accelerate your project since testing will require much less coding.

It's important that you document your device design to act as a solution roadmap.

10. DETERMINE HOW THE ETHERNET/IP SOFTWARE LAYER WILL INTERFACE WITH YOUR APPLICATION SOFTWARE.

Will you interface your device direct through a software API, a dual port interface, an SPI or something else? Your answer will depend on the type of data being transmitted, the size of the data, how critical that data is to business processes and the design of the device. Without a well-defined interface specification, developers won't know the intent of the device. It's important that you document your device design to act as a solution roadmap.

11. ESTABLISH THE TIMING, PERFORMANCE AND DATA REQUIREMENTS FOR YOUR DEVICE.

How quickly does your device have to produce or consume network data? How much data does it need to move? This is the step in the process where you must logically define how your data needs to be exchanged within the network. Your device may need to support multiple connections at varying data sizes and speeds. It may need to provide Scanner as well as Adapter functionality. These specifications all need to be determined before implementation.

12. DETERMINE IF YOUR DEVICE WILL REQUIRE A CONFIGURATION TOOL OR ENHANCEMENT OF AN EXISTING CONFIGURATION TOOL TO SUPPORT THE ETHERNET/IP FEATURES.

Very simple devices may not require an add-on tool to aid in configuration. More complex devices may require development or modification of an existing configuration tool for user configuration of your product's EtherNet/IP features.

Successful EtherNet/IP projects require a lot of thought and preparation before any development or implementation can take place. Development that lacks preparation on requirements and available resources could result in project delays, expensive development changes and failed conformance testing. If you've taken the time to develop a comprehensive business strategy that leverages the benefits of EtherNet/IP, you should also take the time to ensure you have followed these steps to deliver a successful product that delivers the ROI you expect.

ABOUT PYRAMID SOLUTIONS

Pyramid Solutions has been providing industrial, vehicle and Internet solutions to enable product development communications and network connectivity since 1990. We help our clients gain business insights through data monitoring, reliable connectivity and control over devices. Our industry experience with cross-platform development and connectivity positions us to understand your unique technology needs and deliver a visionary solution with exceptional results. For additional information about Pyramid Solutions, visit PyramidSolutions.com.

FOR MORE INFORMATION:

Marketing Department

marketing@pyramidsolutions.com

PYRAMID SOLUTIONS

VISIONARY SOLUTIONS ▲ EXCEPTIONAL RESULTS