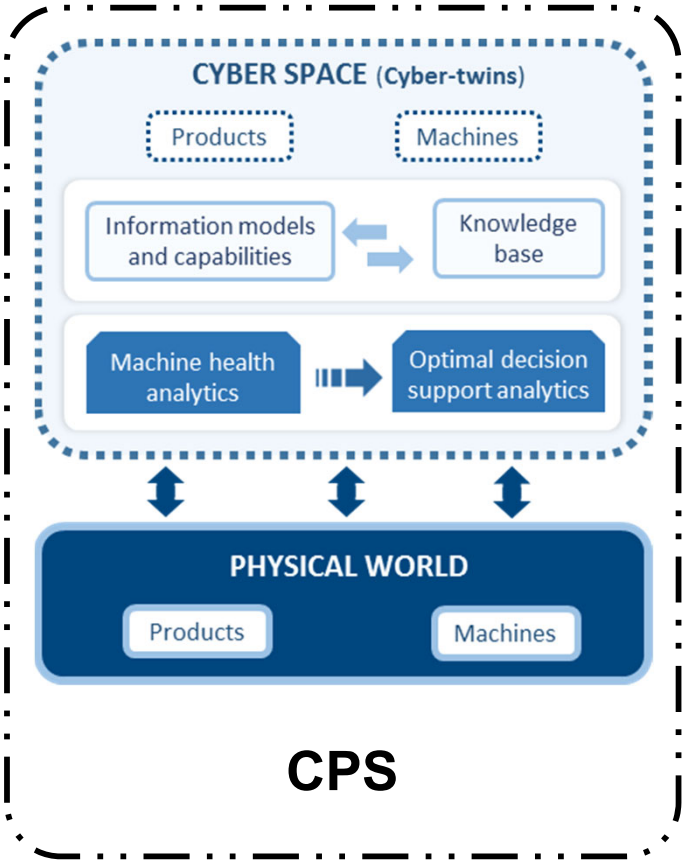
Industrial Communications for Smart Manufacturing:

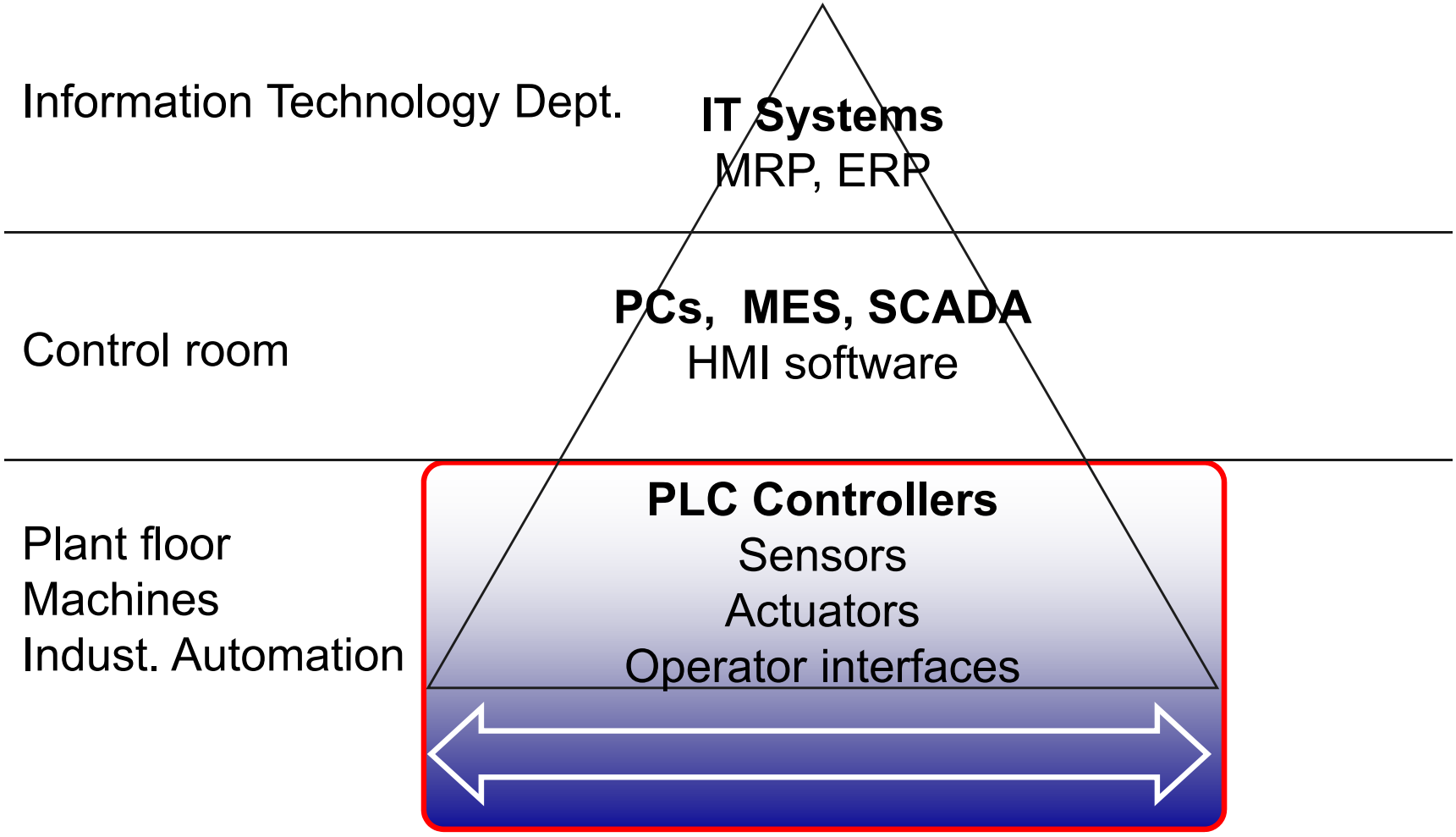
Basics

Some examples of industrial communications

- Fieldbus networks
  - ControlNet
  - DeviceNet
  - Modbus-RTU or ASCII
  - Profibus / Foundation Field Bus
  - PROFINET
  - Modbus-TCP/IP
- Ethernet-based industrial networks
  - EtherNet/IP
  - EtherCAT
  - Time Sensitive Networks (TSN) (Ethernet IEEE 802.1)
  - Highway Addressable Remote Transducer (HART)
- Industrial wireless networks
  - Wireless Sensor Networks (WSN)
  - WiFi-based (IEEE 802.11)
  - Bluetooth-based (IEEE802.15.1)
  - Zigbee-based (IEEE 802.15.4)
- Other open standard technologies
  - OPC UA, MTConnect, MQTT



Levels of industrial communication



Industrial equipment

- **Actuators**: convert a command to power (motors, valves, heater… controlled by the CPU) to drive a machine
- **Sensors**: convert a physical variable to current or voltage (detect / measure process information to bring to the CPU)
- **Programmable controller**: CPU that controls the machine
- **PC**: configuration, programming, maintenance…
- **Operator interface**: for interactions between human and machine
- …

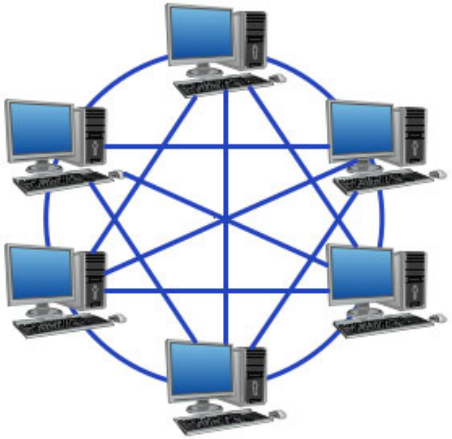
Industrial Communication Protocols

Profibus… EtherNet/IP… Modbus… TSN… ControlNet…OPC UA

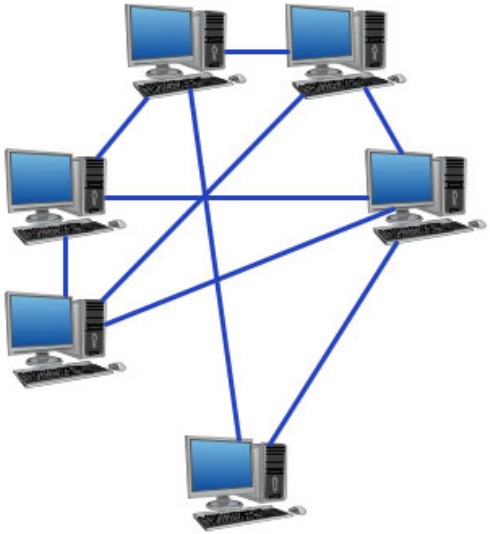
Communications topologies

- A **Network** is an interconnected group of computers and/or controllers, and devices that interact with computers and controllers.
- A **Node** is a computer or other device in a network. Networks are interconnected by many different types of conversion devices, cables, and sometimes, by radio transceivers.
- Five common **topologies**, or arrangements for networks:
  - **Bus**
  - **Star**
  - **Ring**
  - **Mesh**
  - **Full connection**

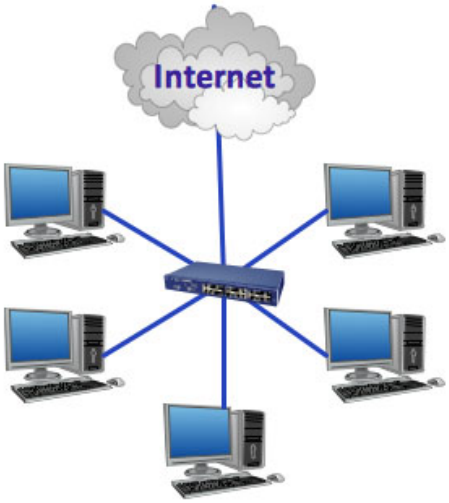
Common types of communications topologies



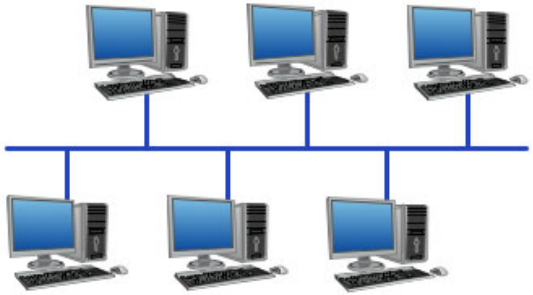
Fully Connected Network Topology



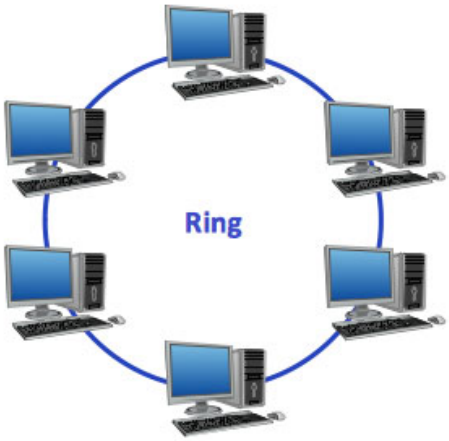
Mesh Network Topology



Star Network Topology



Common Bus Topology



Ring Network Topology

https://www.conceptdraw.com/How-To-Guide/network-topologies

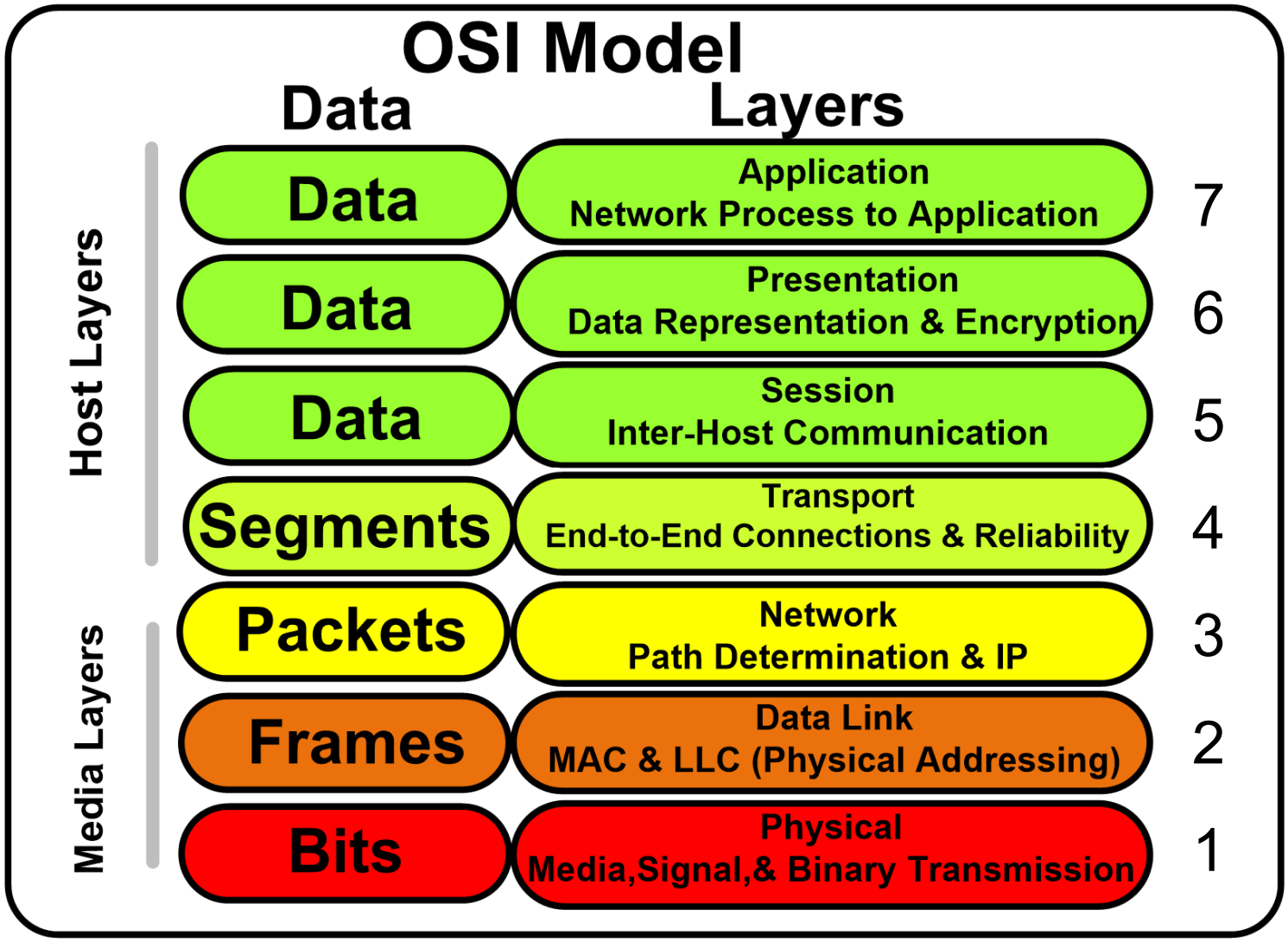
Communications protocols

- A system of rules for data exchange within or between computers, PLCs, instrumentation, control devices, etc.
- Each message has an exact meaning intended to elicit a response from a range of possible predetermined responses
- A protocol must define the syntax, semantics, and synchronization of communication

Systems typically do not use a single protocol to handle a transmission. Instead, they use a set of cooperating protocols, sometimes called a protocol family or protocol suite.

Open Systems Interconnection (OSI)

- A theoretical model of how communications occur on a network
- Published in 1984 by both the ISO (ISO 7498) and the Telecommunications Standardization.
- It has 7 layers:
  - Layer 1 (Physical): physical aspects of a network; the cables, converters, interconnecting devices, etc.
  - Layer 2 (Data-link): concerns itself with how Layers 1 and 3 work together.
  - Layer 3 (Network): addressing scheme for routing of data and messages.
  - Layer 4 (Transport): makes sure that messages get to their correct destination.
  - Layer 5 (Session): actual connections between systems.
  - Layer 6 (Presentation): the way different systems represent data.
  - Layer 7 (Application): software applications.



http://electricalacademia.com/wp-content/uploads/2018/12/image-result-for-osi-model-layers-and-its-function.gif

TCP/IP – OSI - Protocols

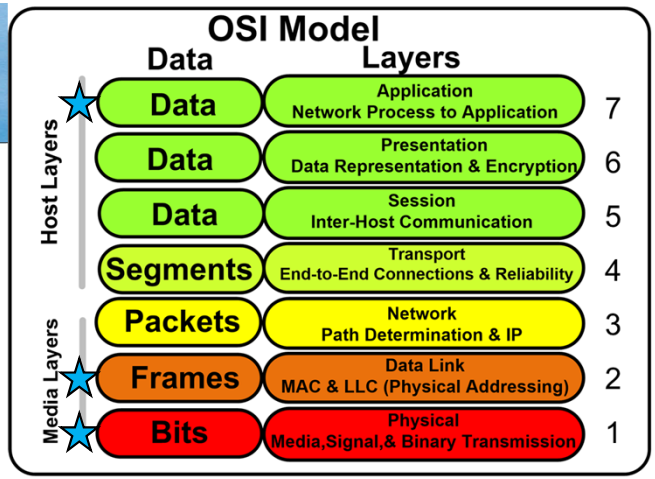
TCP/IP	OSI Model	Protocols
Application Layer	Application Layer	DNS - DHCP - FTP - HTTPS - LDAP - NTP - POP3 - RTP - RTSP - SSH - SIP - SMTP - Telnet - TFTP
	Presentation Layer	JPEG - MIDI - MPEG - PICT - TIFF
	Session Layer	NetBIOS - NFS - PAP - SCP - SQL - ZIP
Transport Layer	Transport Layer	TCP - UDP
Internet Layer	Network Layer	ICMP - IGMP - IPsec - IPv4 - IPv6 - IPX - RIP
Link Layer	Data Link Layer	ARP - ATM - CDP - FDDI - Frame Relay - HDLC - MPLS - PPP - STP - Token Ring
	Physical Layer	Bluetooth - Ethernet - DSL - ISDN - 802.11 - WiFi

https://www.certblaster.com/examnotes-for-network-plus-n10-007-1-1-explain-the-purposes-and-uses-of-ports-and-protocols-part-1-of-2/tcp-osi-protocols/

Fieldbus: an industrial network system for real-time distributed control

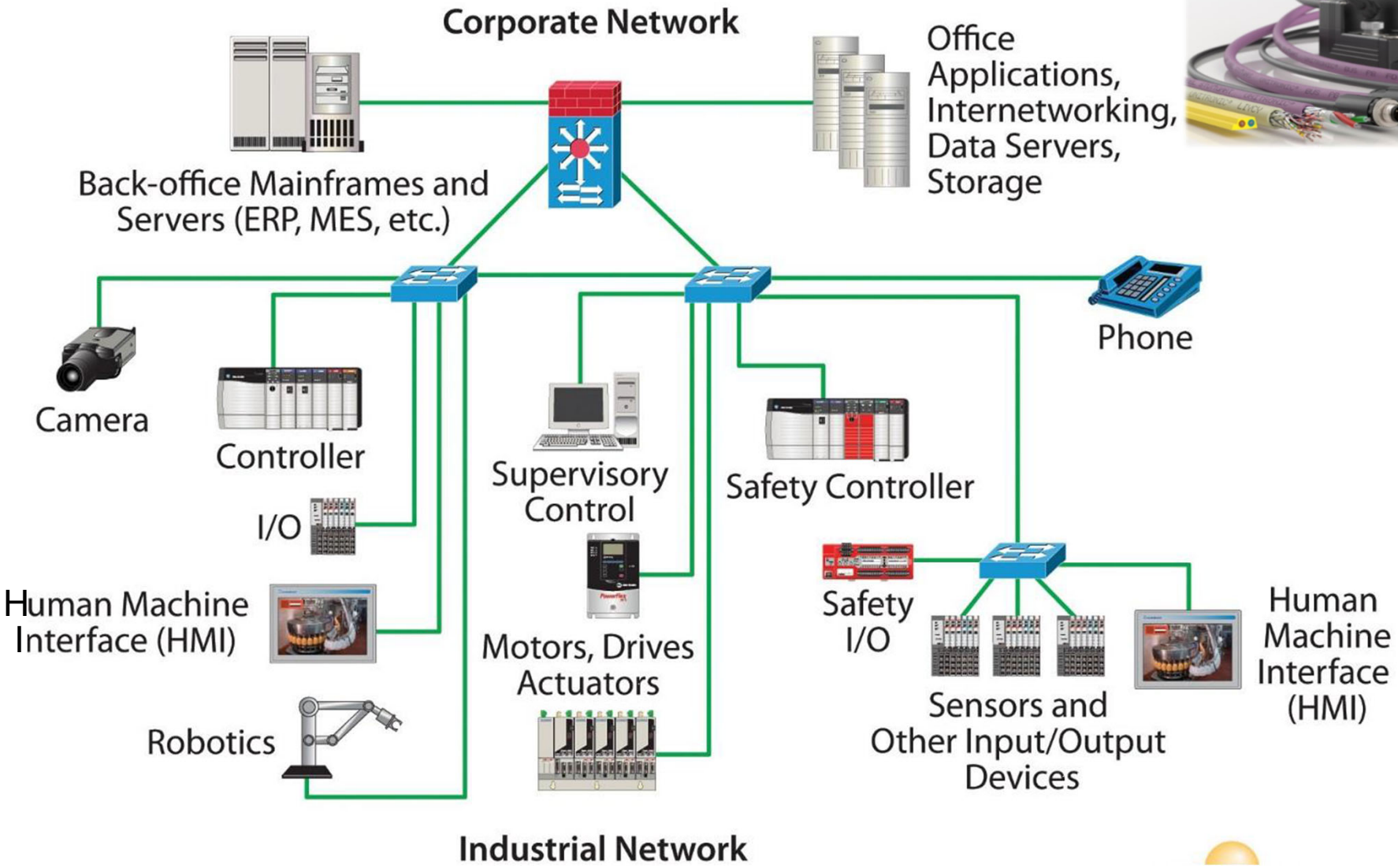
- A way to connect instruments in a manufacturing plant
- Fieldbus works on different types of network topologies
- Eight typical protocols
  - Type 1: FOUNDATION Fieldbus H1
  - Type 2: ControlNet
  - Type 3: PROFIBUS
  - Type 4: P-NET
  - Type 5: FOUNDATION fieldbus HSE (High-Speed Ethernet)
  - Type 6: SwiftNet (a protocol developed for Boeing, since withdrawn)
  - Type 7: WorldFIP
  - Type 8: INTERBUS-S

Fieldbus with reference to OSI model



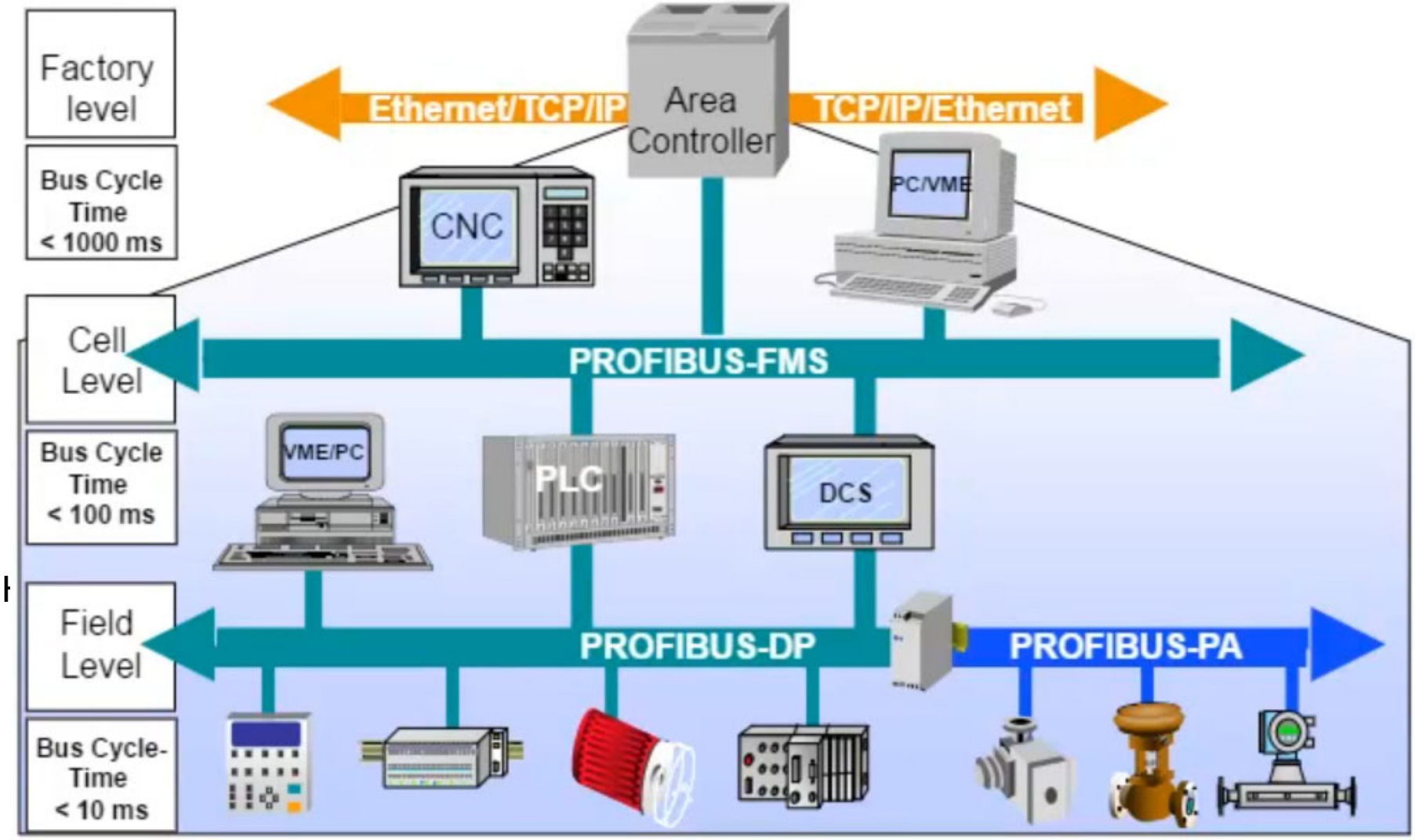
- Fieldbus standards are determined by the *physical media* of the cabling, and *layers 1, 2* and *7* of the OSI reference model.
- The *physical media* and the *physical layer (layer 1)* standards describe, in detail, the implementation of bit timing, synchronization, encoding/decoding, band rate, bus length and the physical connection of the transceiver to the communication wires.
- The *data link layer* (*layer 2*) standard is responsible for fully specifying how messages are assembled ready for transmission by the physical layer, error handling, message-filtering and bus arbitration.
- The *application layer (layer 7)* standard defines how the data communication layers are interfaced to the application that wishes to communicate. It describes message specifications, network management implementations and response to the request from the application of services.
- Layers 3 to 6 are not described in fieldbus standards.

Fieldbus control platform – an example



https://www.processindustryforum.com/wp-content/uploads/2015/08/Fieldbus-control-platform.png

Profibus – an example



Engineering Institute of Technology

Fieldbus video

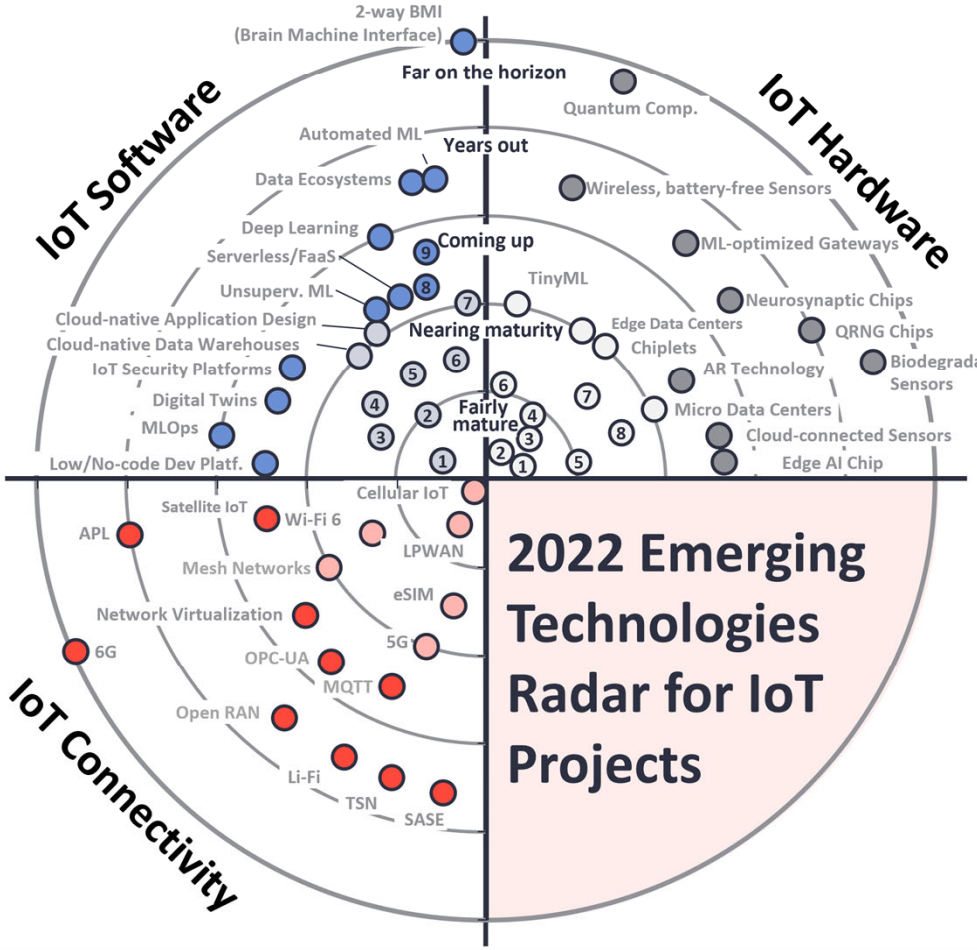
https://www.youtube.com/watch?v=ndc6at_d7uQ

IOT ANALYTICS

April 2022

Your Global IoT Market Research Partner

Emerging IoT Technologies Radar 2022



1. Cloud Computing
2. IoT Platforms
3. Containers
4. Supervised ML
5. IoT-based Streaming Analytics
6. Edge AI
7. Real-time Databases
8. IoT Marketplace
9. Edge Data and App Platforms

1. CPUs
2. MCUs
3. GPUs
4. Security Chips
5. Edge Gateway
6. FPGA
7. ASICs
8. Intelligent Sensors

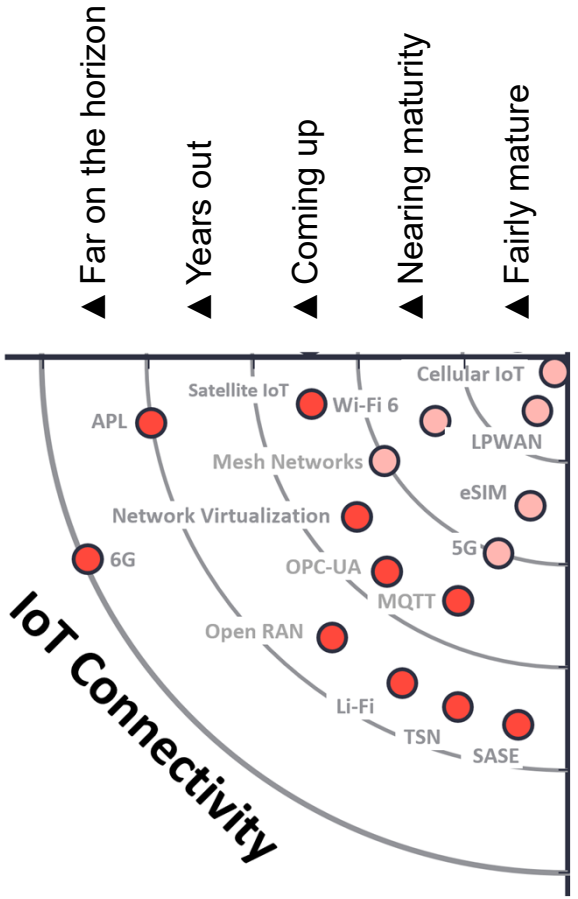
Disclaimer: The 2022 Emerging Technologies Radar for IoT projects represents the analyst view of the IoT Analytics' research team under the disclaimer that, depending on the context, the classification and perception of the project may be different in reality. IoT software refers to middleware, analytics, storage, platforms, and applications. The only relevant positioning of each topic is its distance to the center. The positioning on the axes plays no role.

Source: IoT Analytics Research—April 2022. We welcome republishing of images but ask for source citation with a link to the original post and company website.

https://iot-analytics.com/iot-technologies/??utm_source=browser_subscription&utm_medium=push_banner&utm_campaign=browser_subscription

IoT Connectivity Technologies (2022)

Technology	Description	Classification	Adoption Rate
Cellular IoT (2G/3G/4G)	Provides connectivity to IoT applications via traditional cellular networks	Mainstream	↗
LPWAN	Low-power, wide-area connectivity for IoT applications (e.g., Sigfox, LoRa, NB-IoT, and LTE-M)	Mainstream	↑
eSIM	A SIM-card embedded into mobile devices enables remote SIM provisioning, which allows storing multiple operator profiles simultaneously and switching between them remotely.	Nearing maturity	↑
Mesh Networks	A mesh network is a group of devices that act as a single Wi-Fi network, so there are multiple sources of Wi-Fi around your house instead of just a single router.	Nearing maturity	↗
5G	The fifth generation of cellular networks, commercially launched in 2019	Nearing maturity	↑
Wi-Fi 6	The newest version of the Wi-Fi protocol, also known as IEE 802.11ax	Nearing maturity	↑



↑ = very fast movers

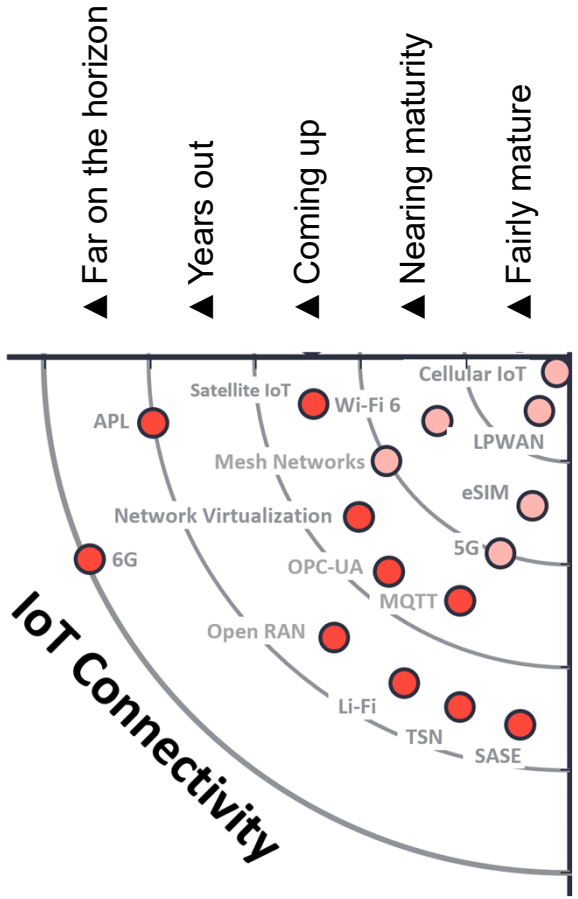
↗ = fast movers

→ = slow movers

IOT ANALYTICS
MARKET INSIGHTS FOR THE INTERNET OF THINGS

IoT Connectivity Technologies (2022)

Technology	Description	Classification	Adoption Rate
Network Virtualization	Abstracts network elements and resources into a logical virtual network that runs independently on top of a physical network	Coming up	→
MQTT	MQTT is a lightweight, publish-subscribe network protocol that transports messages between devices.	Coming up	→
OPC Unified Architecture (UA)	OPC UA is a machine-to-machine communication protocol for industrial automation from the OPC Foundation.	Coming up	↗
Satellite IoT	Provides connectivity to IoT applications via satellite networks	Coming up	↗
TSN	Time-Sensitive Networking is a set of standards defined by the IEEE for the time-sensitive transmission of data over deterministic Ethernet networks.	Years out	→
Li-Fi	Wireless communication technology that uses light to transmit data	Years out	→



↑ = very fast movers

↗ = fast movers

→ = slow movers

IOT ANALYTICS
MARKET INSIGHTS FOR THE INTERNET OF THINGS

INDUSTRIAL COMMUNICATIONS:

OPC:- Open Process Control

OPC UA:- Open Process Control Unified Architecture

References:

- "OPC Unified Architecture" by Wolfgang Mahnke, available via University of Auckland Library
- "VDW and OPC Foundation: OPC UA Information Model for CNC Systems". VDW – German Machine Tool Builders' Association. Release 1.0. July 05, 2017
- OPC Foundation: https://opcfoundation.org/

  and many many more on the Internet

Google

opc protocols

All Images Shopping News Videos More Tools

About 4,480,000 results (0.41 seconds)

https://opcfoundation.org › about › what-is-opc

What is OPC? - OPC Foundation

OPC is the interoperability standard for the secure and reliable exchange of data in the industrial automation space and in other industries.

People also ask

What is OPC protocol?

Does OPC use TCP?

What is OPC technology?

What is the difference between Modbus and OPC?

Feedback

https://en.wikipedia.org › wiki › Open_Platform_Com...

Open Platform Communications - Wikipedia

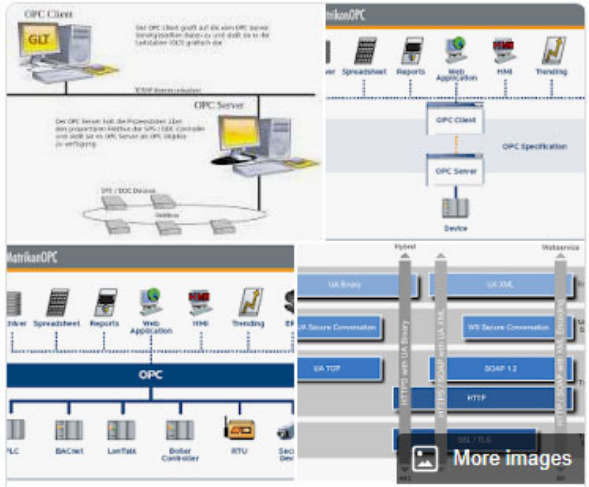
Open Platform Communications (**OPC**) is a series of standards and specifications for industrial telecommunication. They are based on Object Linking and ...

History · Design

https://en.wikipedia.org › wiki › OPC_Unified_Architec...

OPC Unified Architecture - Wikipedia

OPC Base Services are abstract method descriptions, which are **protocol** independent and provide the basis for **OPC** UA functionality. The transport layer puts ...



Open Platform Communications

Open Platform Communications is a series of standards and specifications for industrial telecommunication. They are based on Object Linking and Embedding for process control. An industrial automation task force developed the original standard in 1996 under the name OLE for Process Control.

Wikipedia

Feedback

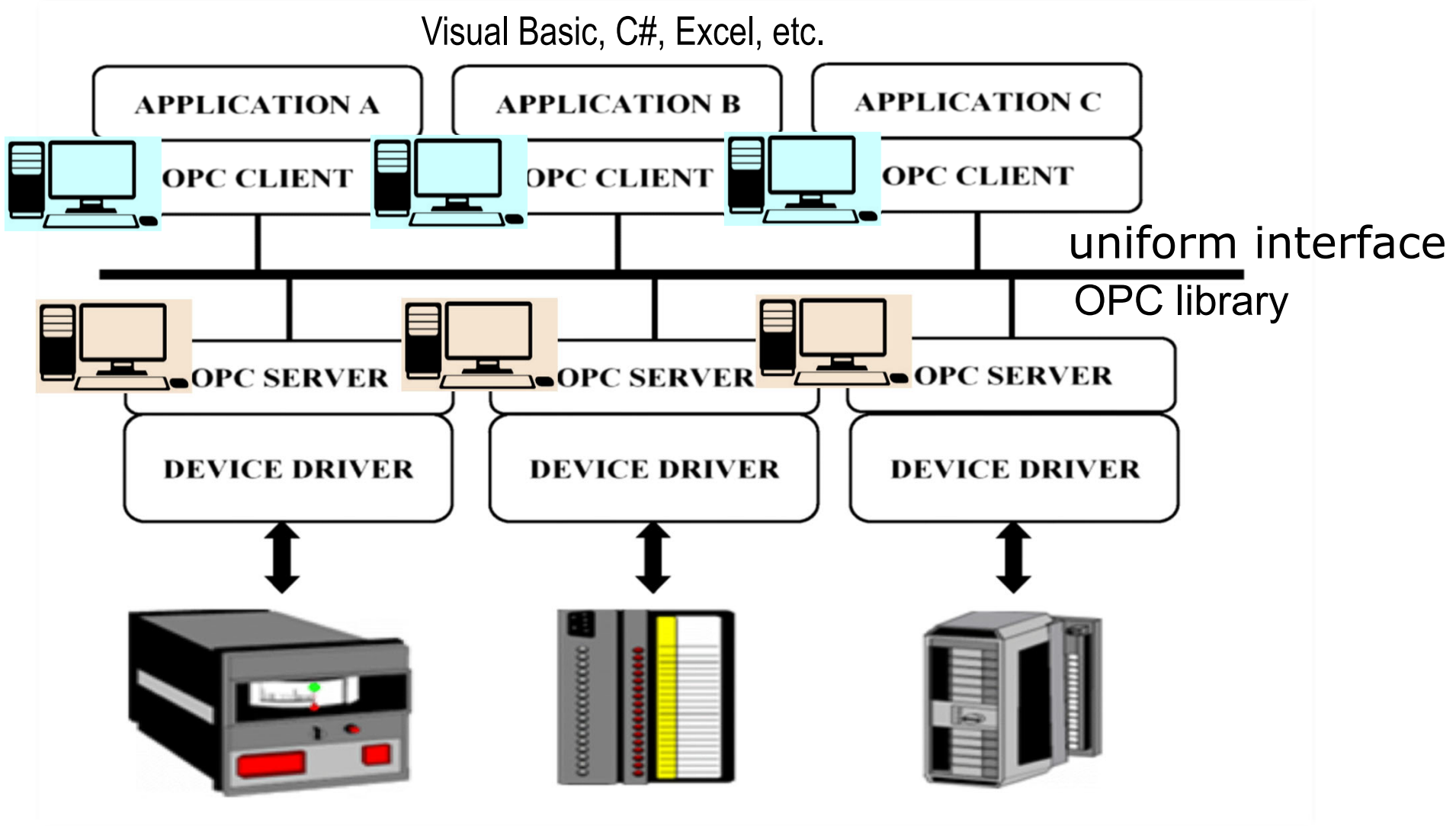
Snapshot of OPC – What

- OPC (OLE for Process Control now known as Open Process Control), also known as OPC Classic
- OPC is a standard, manufacturer-independent programming interface through which an automation application client, e.g. a human interface, can access the plant data coming from remote devices, e.g. PLCs, field bus devices or real-time databases.
- Built on Windows-based OLE, COM and DCOM technologies.
- OPC consists of three major components:
  - OPC - DA = Data Access (widespread, mature)
  - OPC - AE = Alarms and Events (some use)
  - OPC - HDA = Historical Data Access (little use)

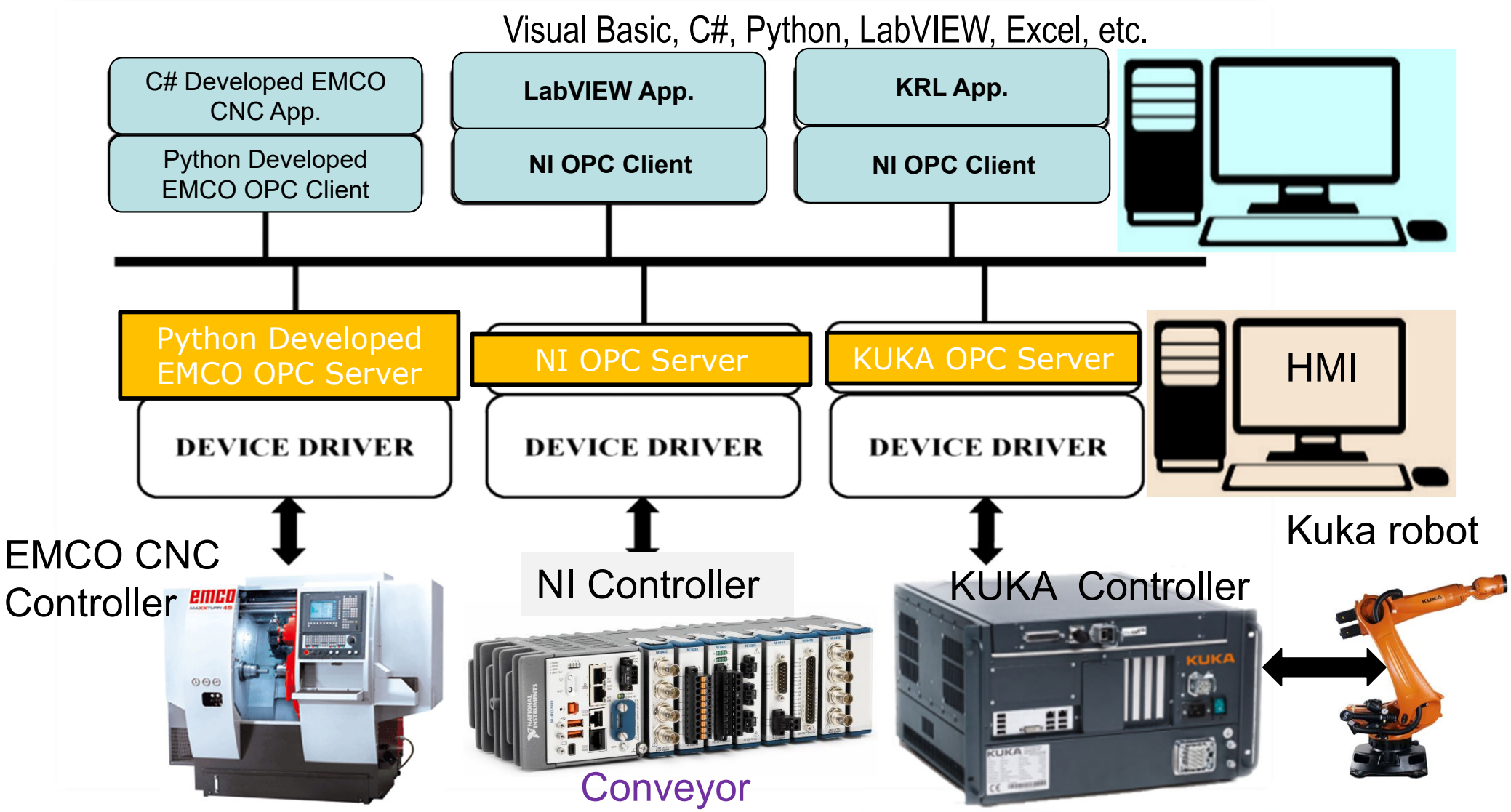
Snapshot of OPC – How

- Manufacturer of automation devices supplies an *OPC server* that runs on a PC, which communicates with its devices through a *proprietary* protocol.
- All OPC servers present the process variables in the *same format* to any clients as shared variables for a *uniform interface*.
- Interface has a set of commands collected in a software library (DLL) that can be called for accessing OPC servers, by client applications written in Visual Basic, C# or other Microsoft programming languages (even Excel).
- OPC library allows to read and write process variables, read alarms and events and acknowledge alarms, and retrieve historical data from data bases.

Snapshot of OPC - How

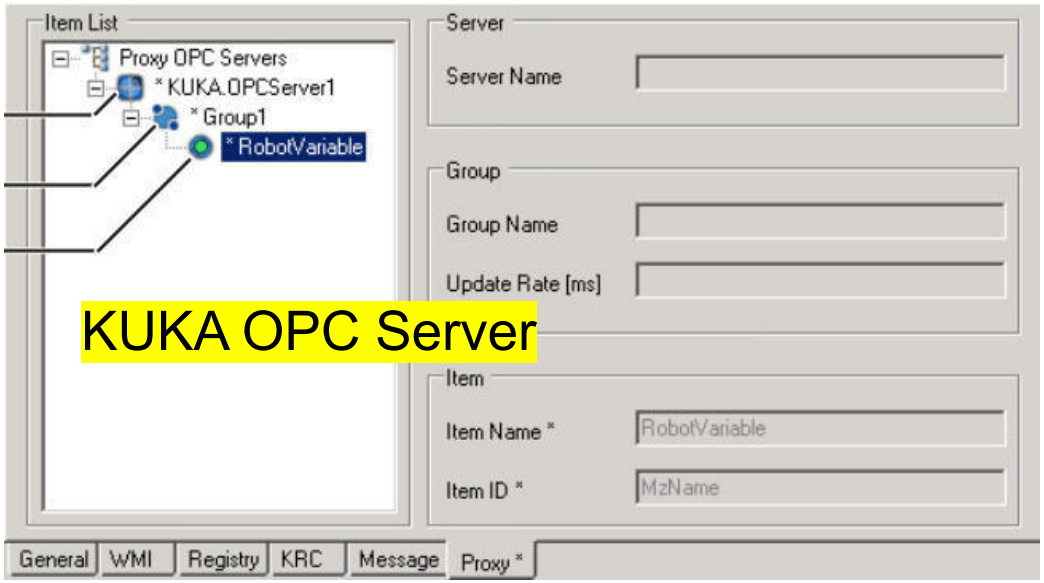


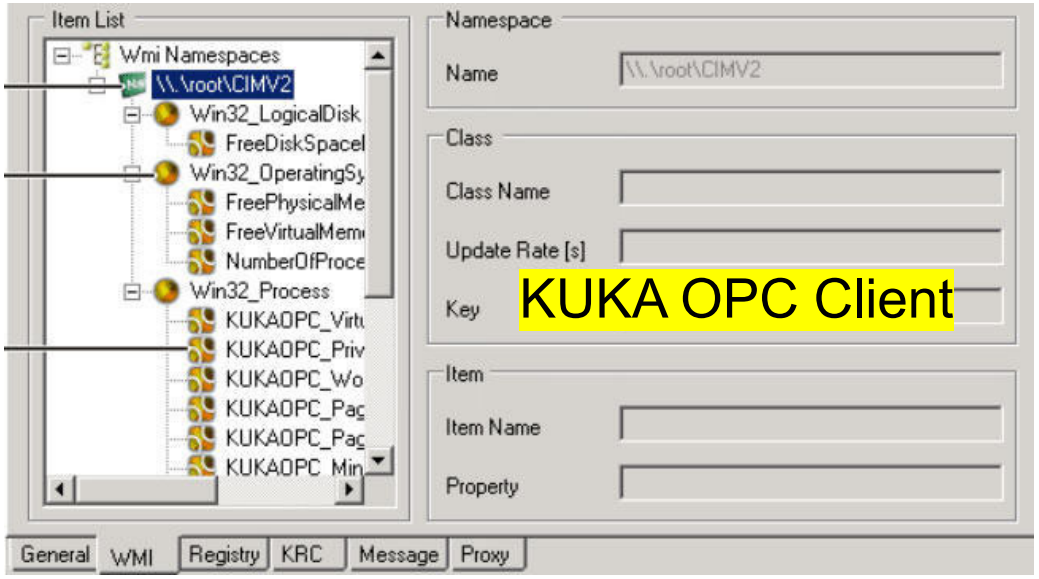
Examples


Visual Basic, C#, Python, LabVIEW, Excel, etc.
C# Developed EMCO CNC App.
Python Developed EMCO OPC Client
LabVIEW App.
NI OPC Client
KRL App.
NI OPC Client
Python Developed EMCO OPC Server
NI OPC Server
KUKA OPC Server
DEVICE DRIVER
DEVICE DRIVER
DEVICE DRIVER
HMI
EMCO CNC Controller
NI Controller
KUKA Controller
Kuka robot
Conveyor

KUKA OPC Server and Client

The KUKA OPC Server is a licensed software option for the KUKA controller which makes it possible to access KUKA system and user defined variables (tags) of the controller externally from anywhere in the network.



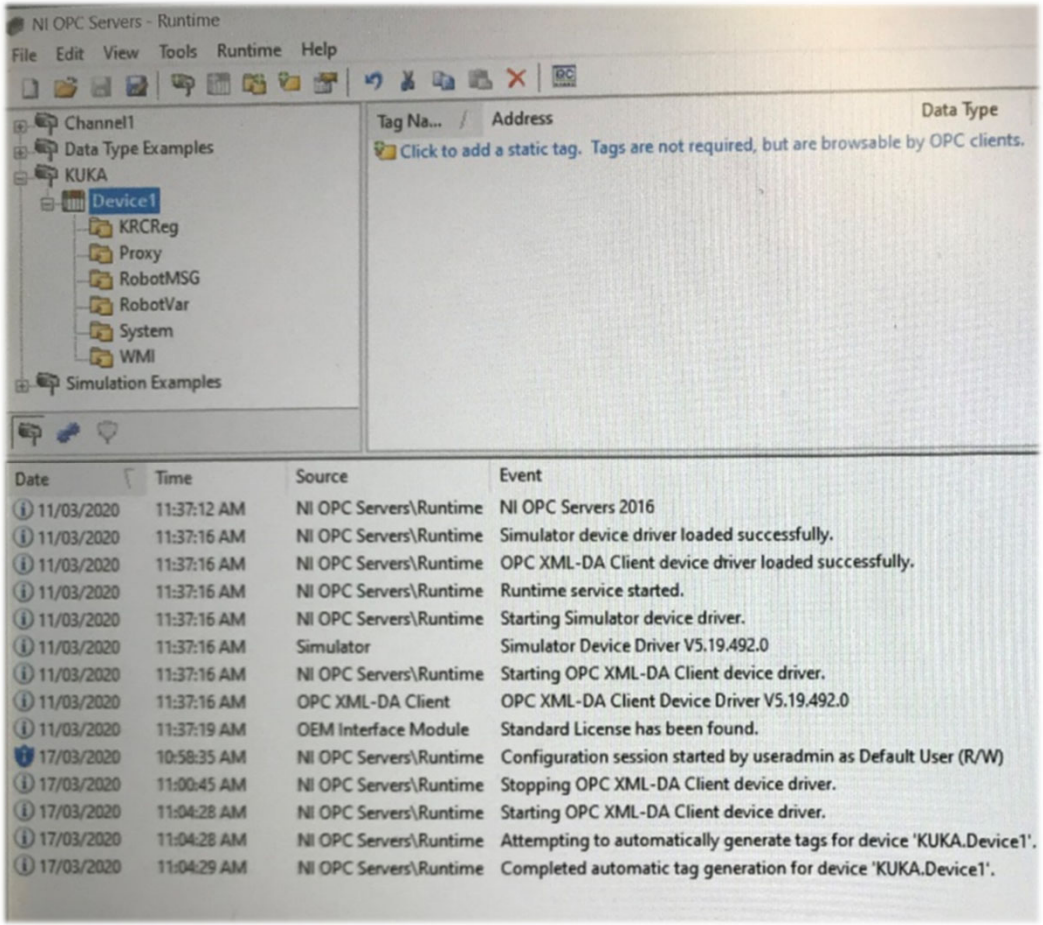


Automation platform

- Automation platforms such as ABB's 800XA platform act as OPC clients to collect data from PLCs or databases through third-party OPC servers.
- Several automation platforms act themselves as an OPC server to publish their data, events and historical data.

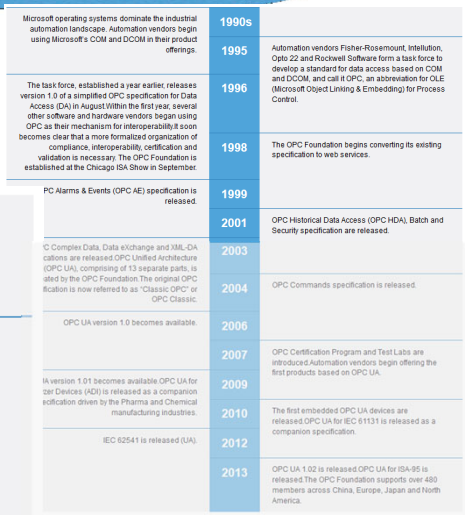
NI OPC Server

- NI OPC Servers bridge converts proprietary industrial protocols to the open OPC Classic and OPC Unified Architecture (UA) protocols.
- Enables NI LabVIEW software to communicate to many different PLCs and third-party devices through the OPC Client that is included with the LabVIEW Datalogging and Supervisory Control (DSC) Module.
- The combination of NI OPC Servers and LabVIEW provides a single platform for delivering high-performance measurements and control to industrial systems.



History of OPC

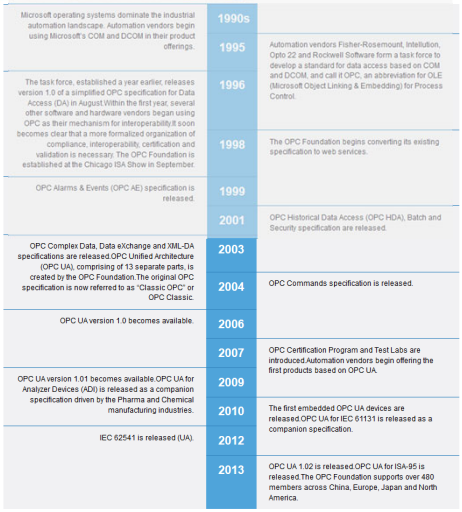
Microsoft operating systems dominate the industrial automation landscape. Automation vendors begin using Microsoft's COM and DCOM in their product offerings.	Win 3.x 1990s	
	1995 Win 9x	Automation vendors Fisher-Rosemount, Intellution, Opto 22 and Rockwell Software form a task force to develop a standard for data access based on COM and DCOM, and call it OPC, an abbreviation for OLE (Microsoft Object Linking & Embedding) for Process Control.
The task force, established a year earlier, releases version 1.0 of a simplified OPC specification for Data Access (DA) in August.Within the first year, several other software and hardware vendors began using OPC as their mechanism for interoperability.It soon becomes clear that a more formalized organization of compliance, interoperability, certification and validation is necessary. The OPC Foundation is established at the Chicago ISA Show in September.	1996	
	1998 Win 98	The OPC Foundation begins converting its existing specification to web services.
OPC Alarms & Events (OPC AE) specification is released.	1999 Win 2000	
	2001	OPC Historical Data Access (OPC HDA), Batch and Security specification are released.



https://opcfoundation.org/about/opc-foundation/history/

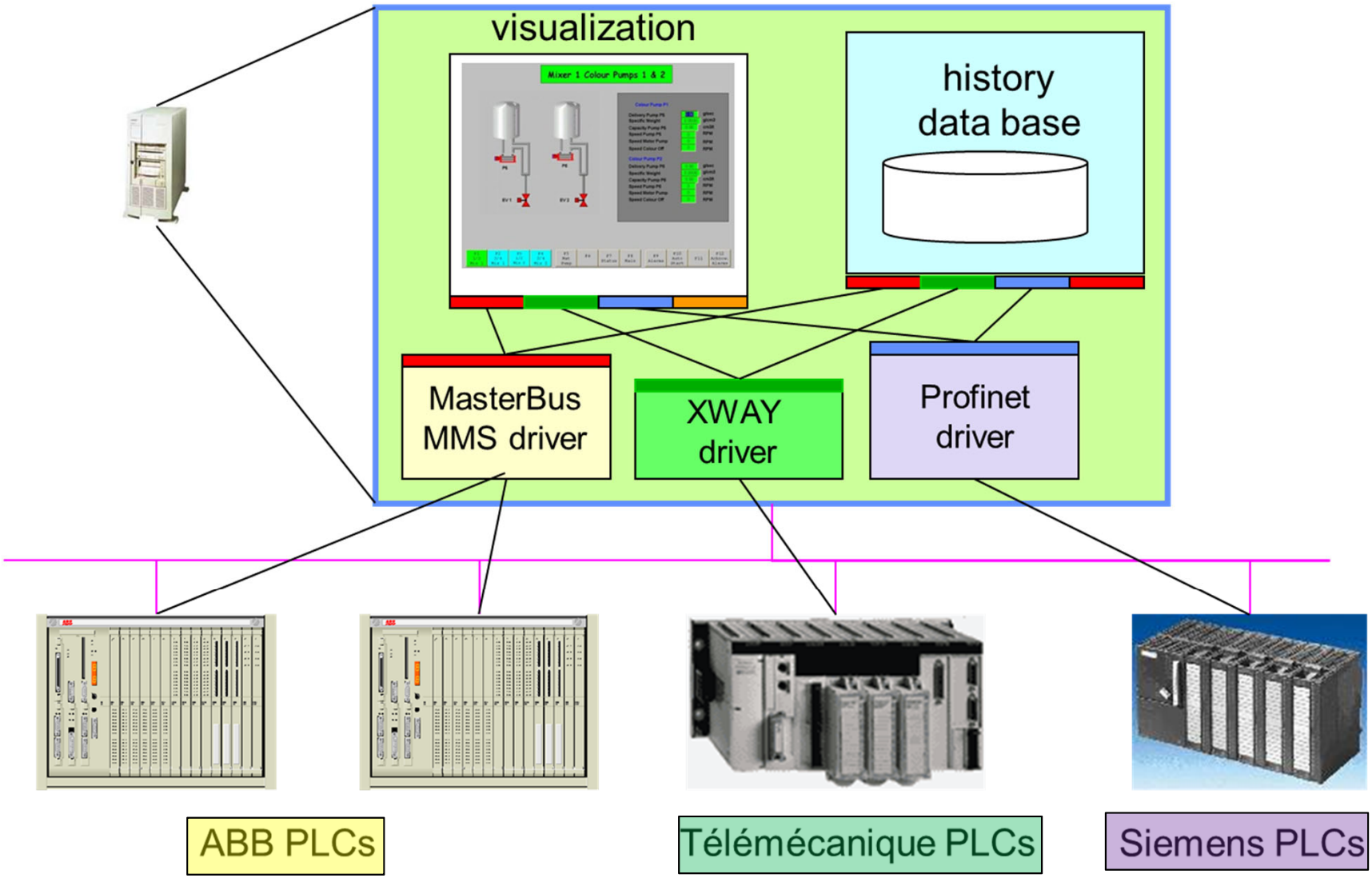
History of OPC / OPC UA *(Free from* Windows *)*

OPC Complex Data, Data eXchange and XML-DA specifications are released.OPC Unified Architecture (OPC UA), comprising of 13 separate parts, is created by the OPC Foundation.The original OPC specification is now referred to as “Classic OPC” or OPC Classic.	2003	
	2004	OPC Commands specification is released.
OPC UA version 1.0 becomes available.	2006	
	2007	OPC Certification Program and Test Labs are introduced.Automation vendors begin offering the first products based on OPC UA.
OPC UA version 1.01 becomes available.OPC UA for Analyzer Devices (ADI) is released as a companion specification driven by the Pharma and Chemical manufacturing industries.	2009	
	2010	The first embedded OPC UA devices are released.OPC UA for IEC 61131 is released as a companion specification.
IEC 62541 is released (UA).	2012	
	2013	OPC UA 1.02 is released.OPC UA for ISA-95 is released.The OPC Foundation supports over 480 members across China, Europe, Japan and North America.



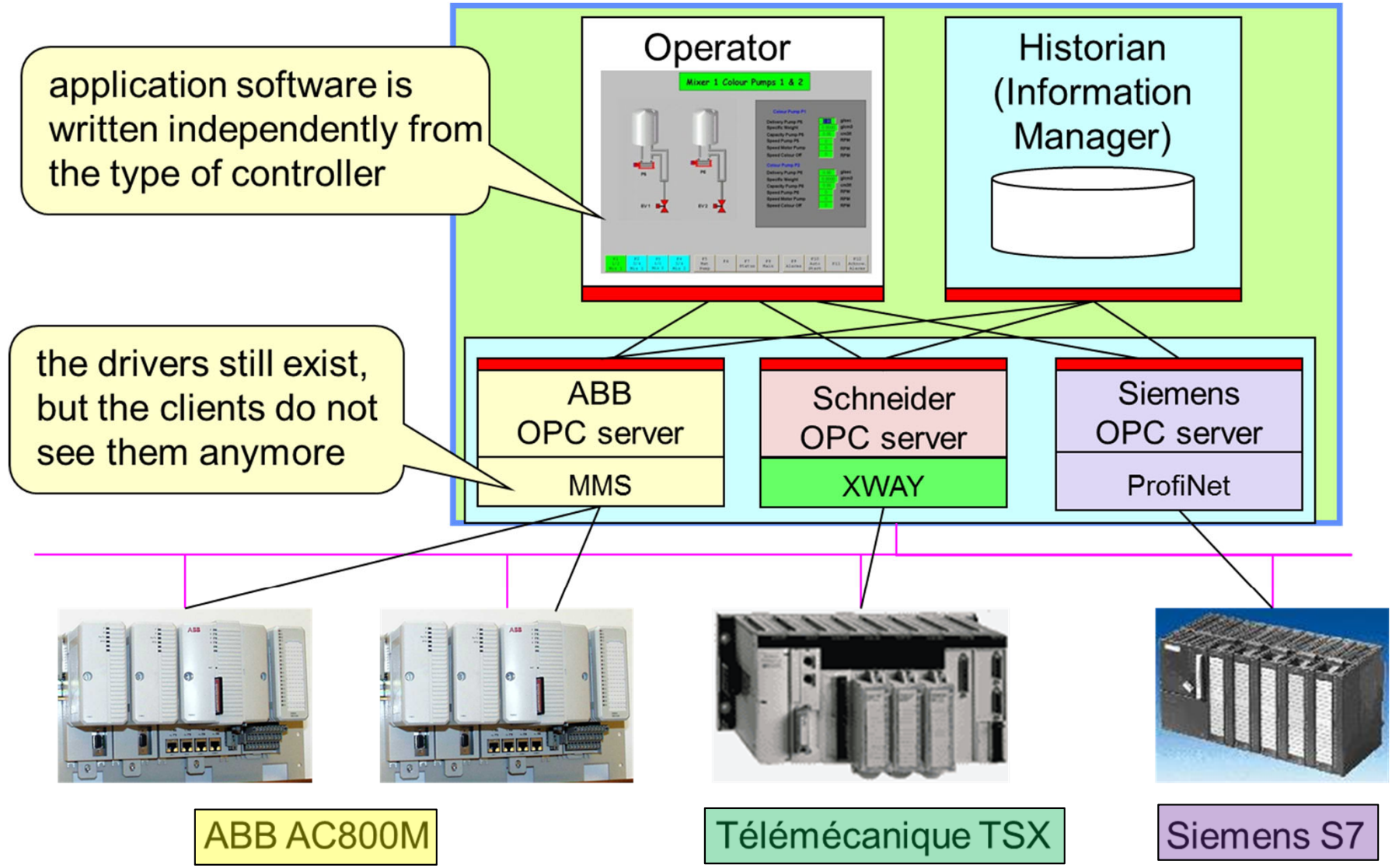
https://opcfoundation.org/about/opc-foundation/history/

Before OPC



Prof. Dr. H. Kirrmann. EPFL. 2012

After OPC (in a SCADA application)



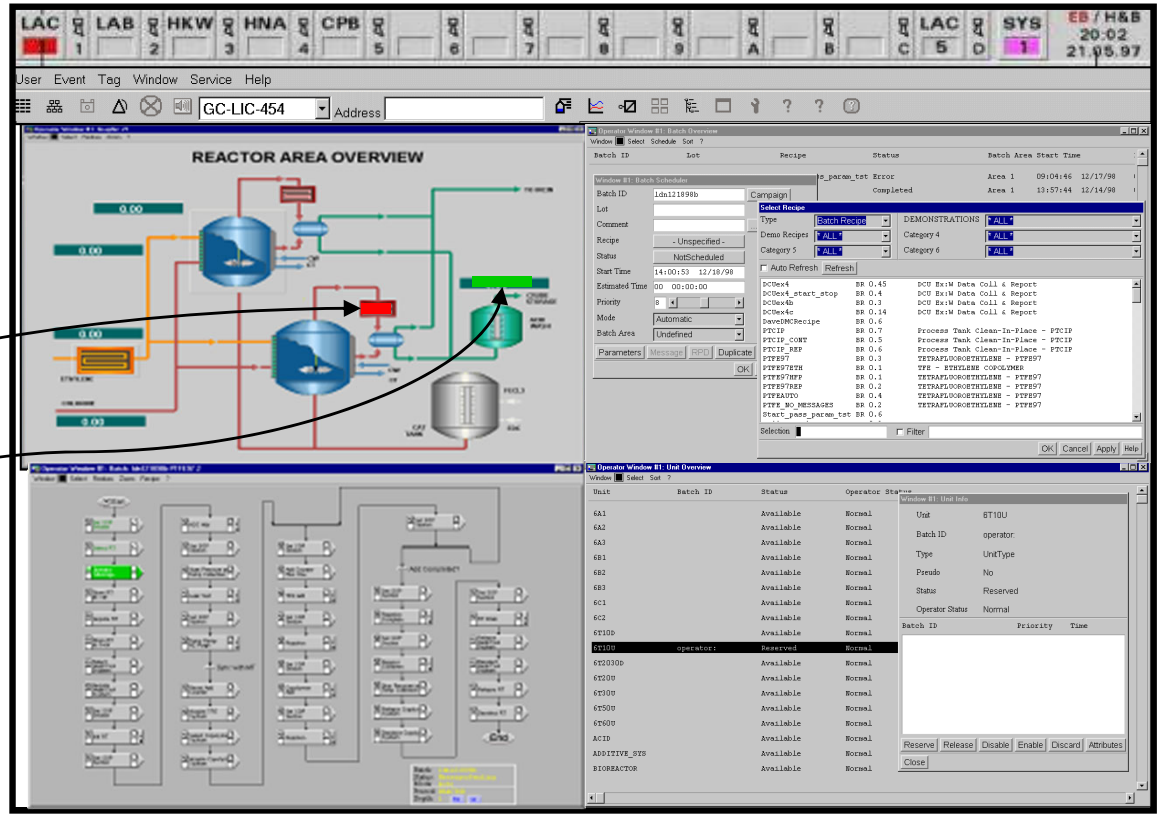
SCADA : Supervisory control and data acquisition

Prof. Dr. H. Kirrmann. EPFL. 2012

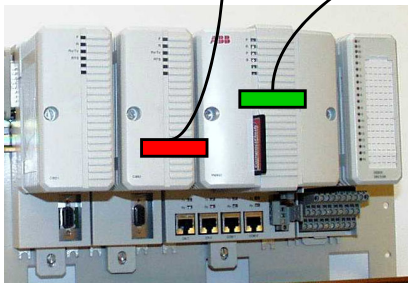
Specification 1: OPC DA (for Data Access)

- Process variables:
  - generated by the sensors or calculated in a PLC
  - sent upon a change, on demand or when a given time elapsed
- Main clients of OPC DA:
  - Visualization
  - (soft-) control

Client



Server end

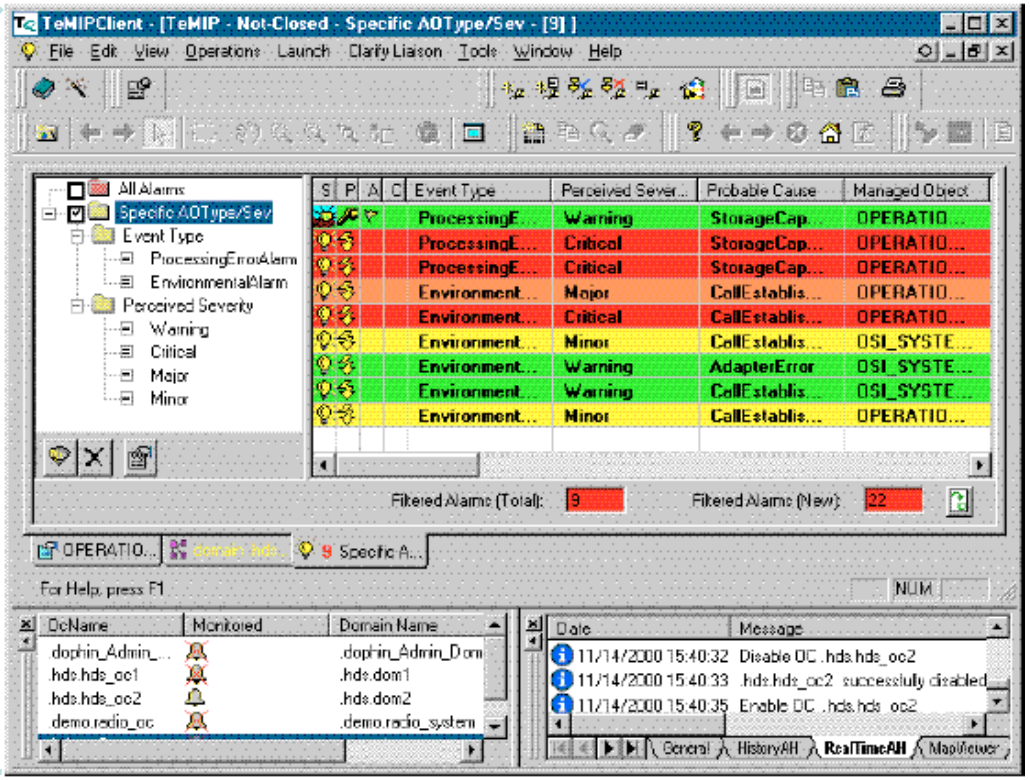


Prof. Dr. H. Kirrmann. EPFL. 2012

Specification 2: OPC AE (for Alarms and Events)

- Alarms:
  - abnormal states in the process that require attention, such as "low oil pressure"
- Events:
  - changes in the process that need to be logged, such as "production start"
- Main clients of OPC AE:
  - Alarms loggers
  - Event loggers

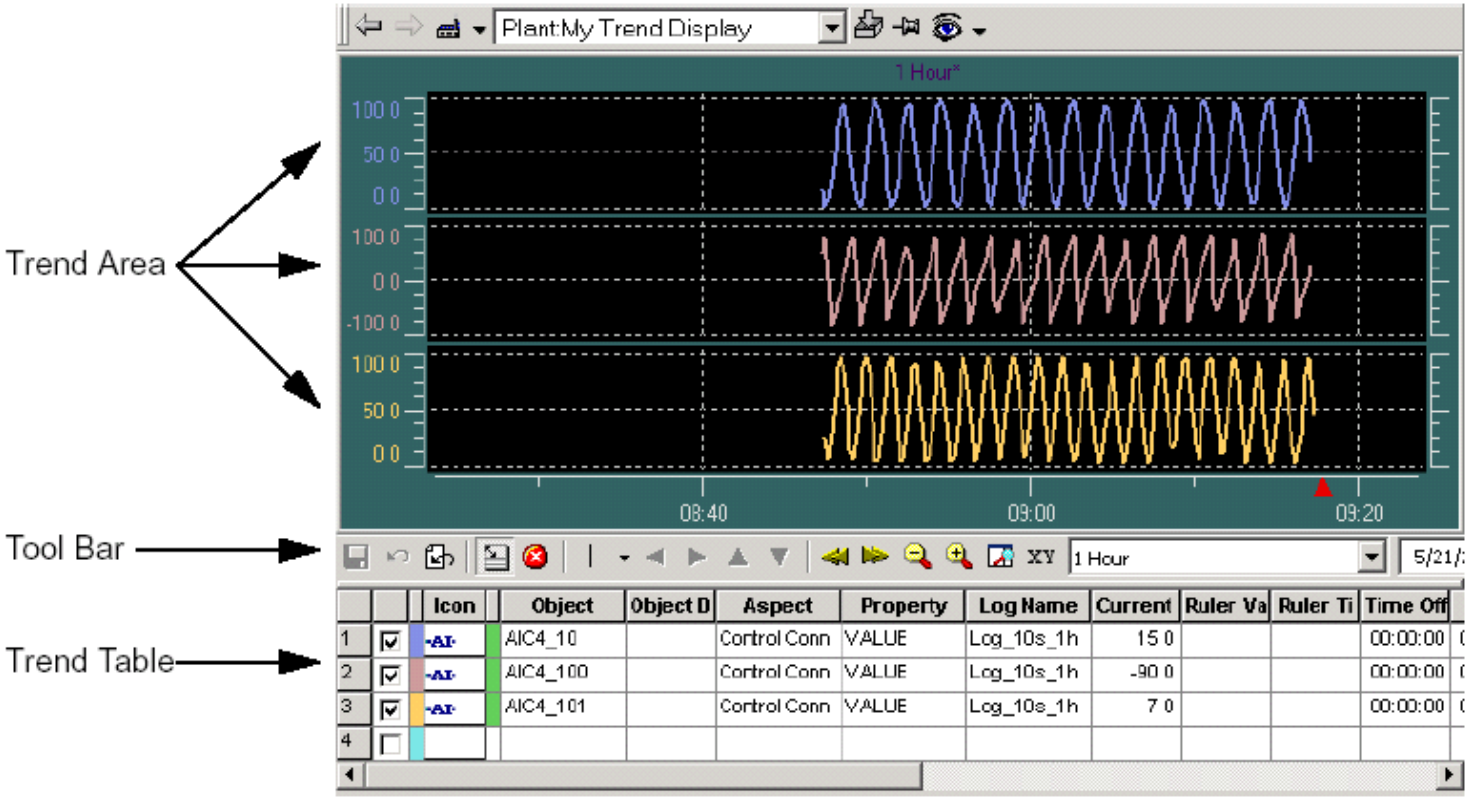
- determine the exact time of change (time stamping)
- categorized by priorities
- log for further use
- acknowledge alarms (events are not acknowledged)
- link to clear text explanation



Prof. Dr. H. Kirrmann. EPFL. 2012

Specification 3: OPC HDA (for Historical Data Access)

- Historical Data are process states and events, e.g. process variables, operator actions, recorded alarms,... that are stored as logs in a long-term storage for later analysis.
- OPC HDA specifies how historical data are retrieved from the logs in the long-term storage, filtered and aggregated.
- Typical OPC HDA client: Trend Displays and Historians.



Prof. Dr. H. Kirrmann. EPFL. 2012

INDUSTRIAL COMMUNICATIONS:

OPC:- Open Process Control

OPC UA:- Open Process Control Unified Architecture

References:

- "OPC Unified Architecture" by Wolfgang Mahnke, available via University of Auckland Library
- "VDW and OPC Foundation: OPC UA Information Model for CNC Systems". VDW – German Machine Tool Builders' Association. Release 1.0. July 05, 2017
- OPC Foundation: https://opcfoundation.org/

  and many many more on the Internet

INDUSTRIAL COMMUNICATIONS:

OPC UA:- Open Process Control Unified Architecture

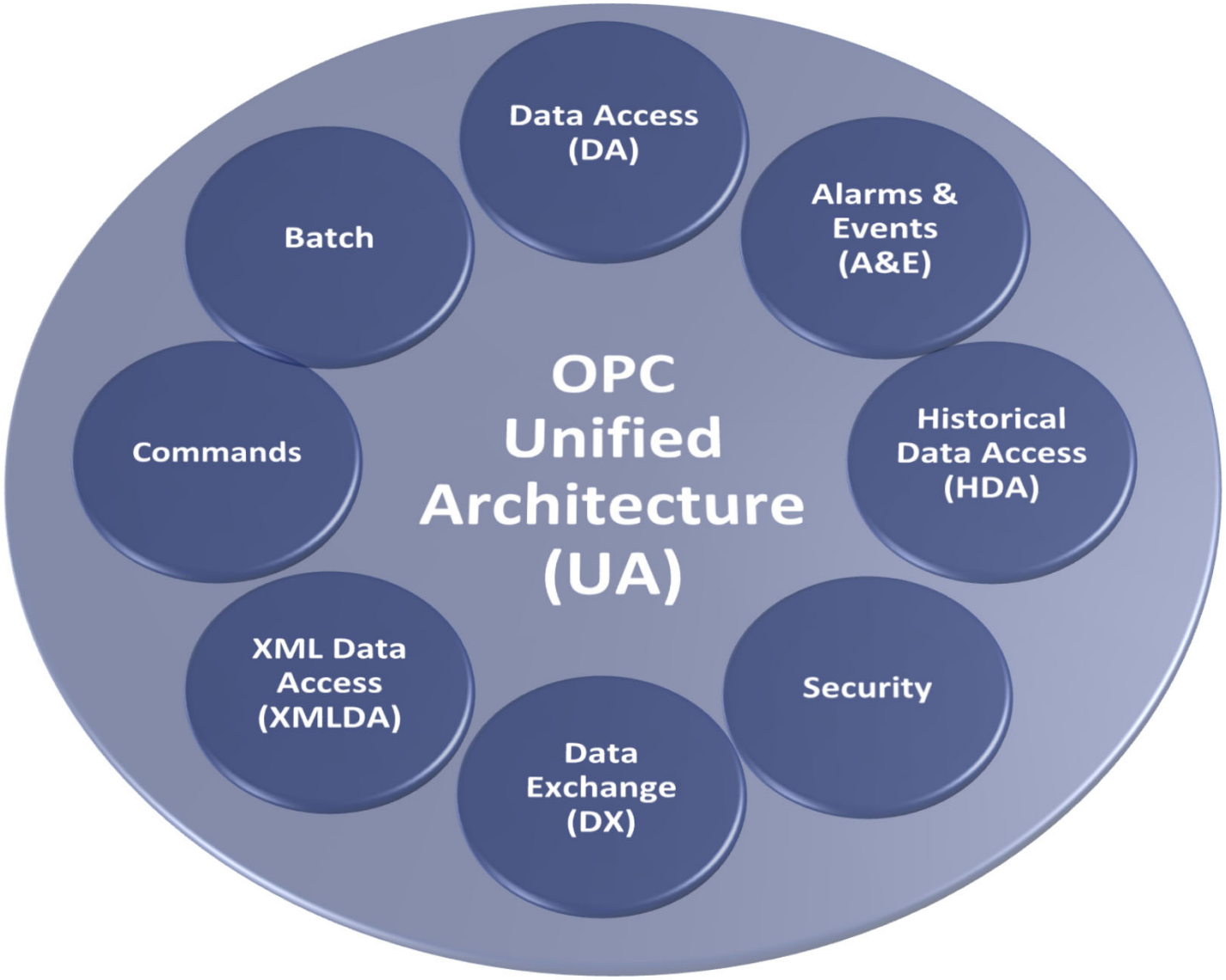
(Part I)

Beyond Microsoft: OPC UA (OPC Unified Architecture)

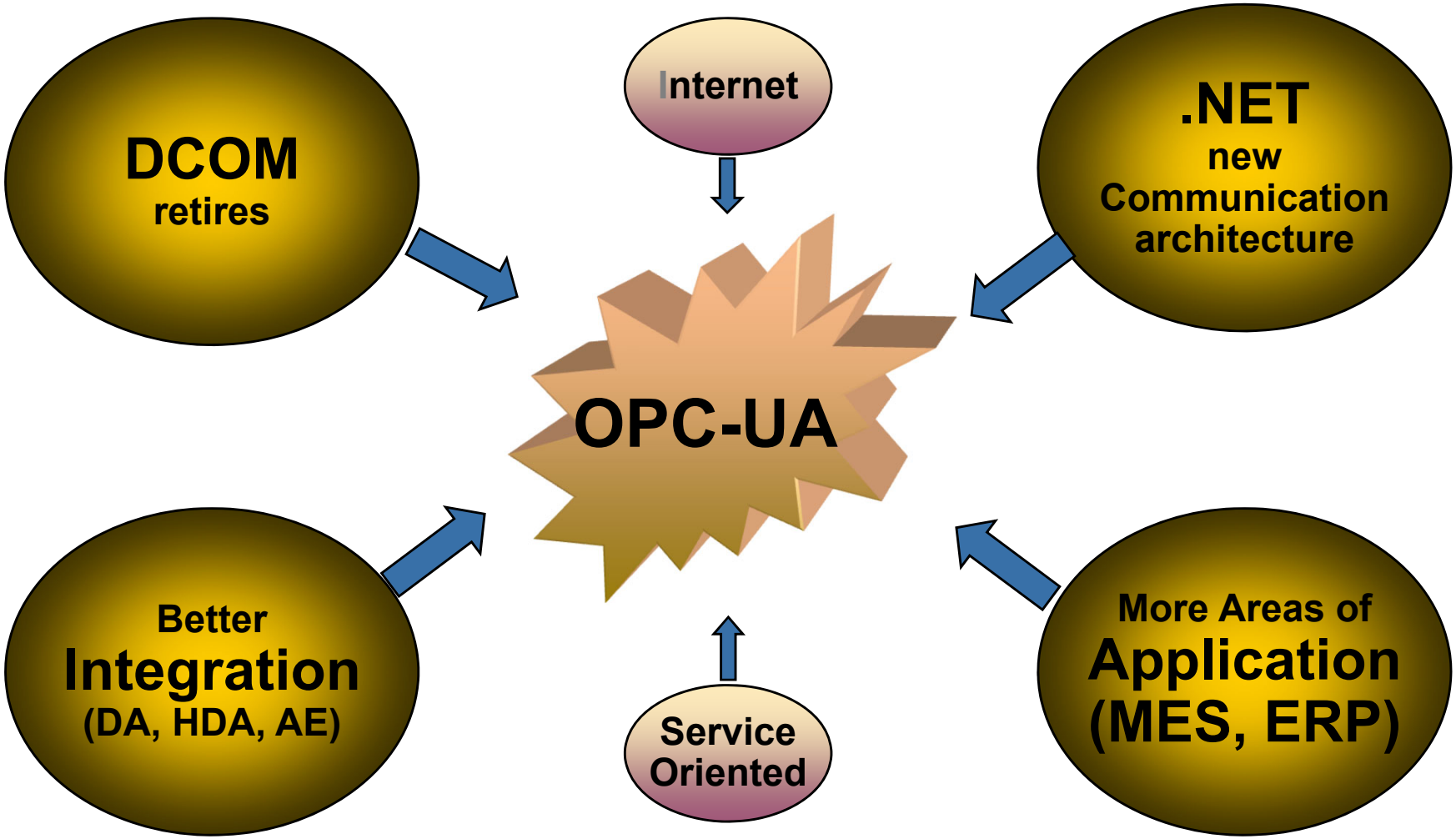
- More independence from Microsoft and use web technology
- Uses web services (via XML/SOAP/WSDL) for all kind of *transactions*: query, read, write, subscribe,...
- Standardizes the *interfaces* and the transmitted *data*
- More secure
- Simple TCP/IP connection
- Open Source

	2001	OPC Historical Data Access (OPC HDA), Batch and Security specification are released.
OPC Complex Data, Data eXchange and XML-DA specifications are released.OPC Unified Architecture (OPC UA), comprising of 13 separate parts, is created by the OPC Foundation.The original OPC specification is now referred to as "Classic OPC" or OPC Classic.	2003	
	2004	OPC Commands specification is released.
OPC UA version 1.0 becomes available.	2006	
	2007	OPC Certification Program and Test Labs are introduced.Automation vendors begin offering the first products based on OPC UA.
OPC UA version 1.01 becomes available.OPC UA for Analyzer Devices (ADI) is released as a companion specification driven by the Pharma and Chemical manufacturing industries.	2009	
	2010	The first embedded OPC UA devices are released.OPC UA for IEC 61131 is released as a companion specification.
IEC 62541 is released (UA).	2012	
	2013	OPC UA 1.02 is released.OPC UA for ISA-95 is released.The OPC Foundation supports over 480 members across China, Europe, Japan and North America.

OPC UA for Common Interfaces

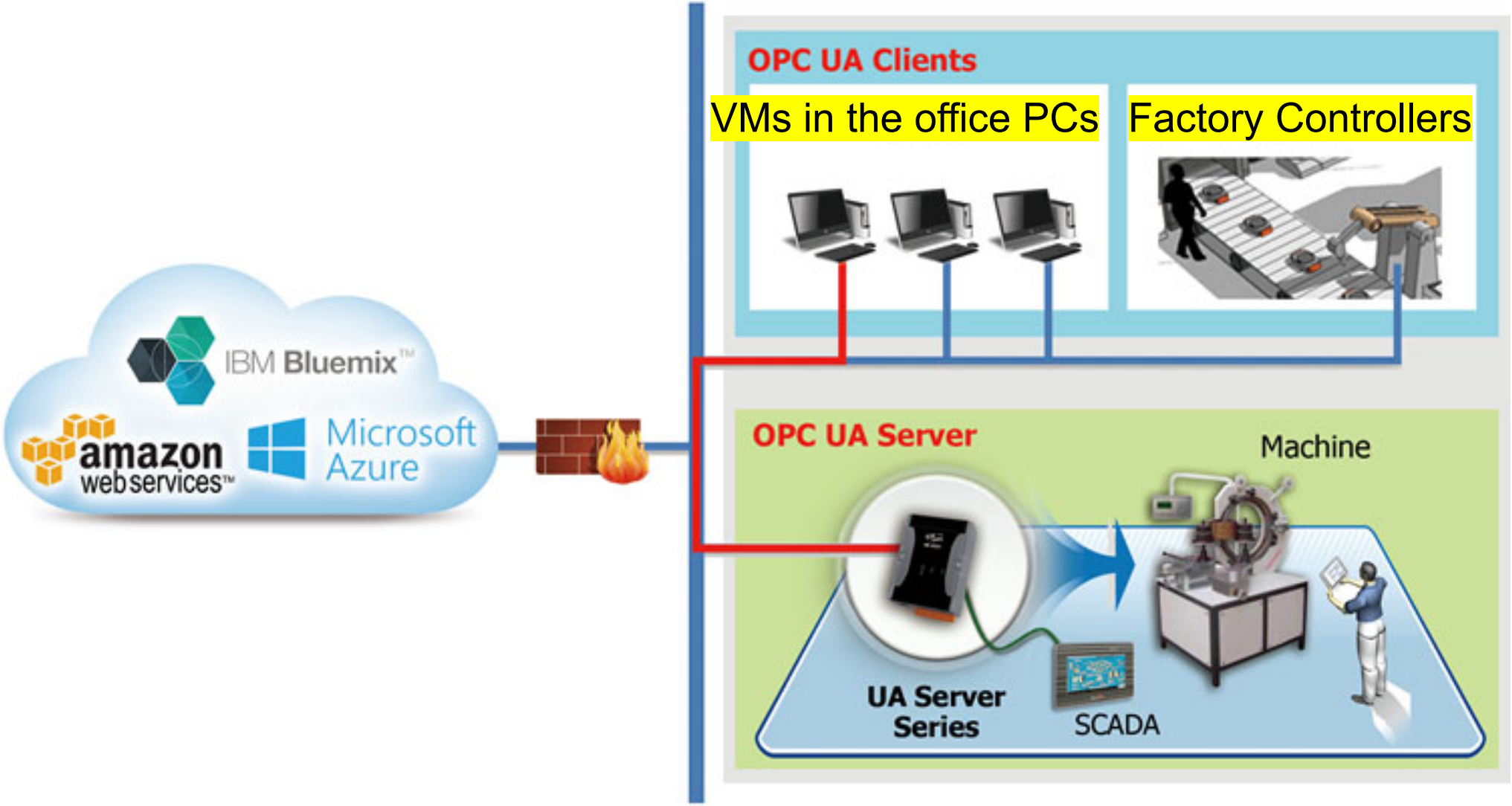


OPC UA Motivation



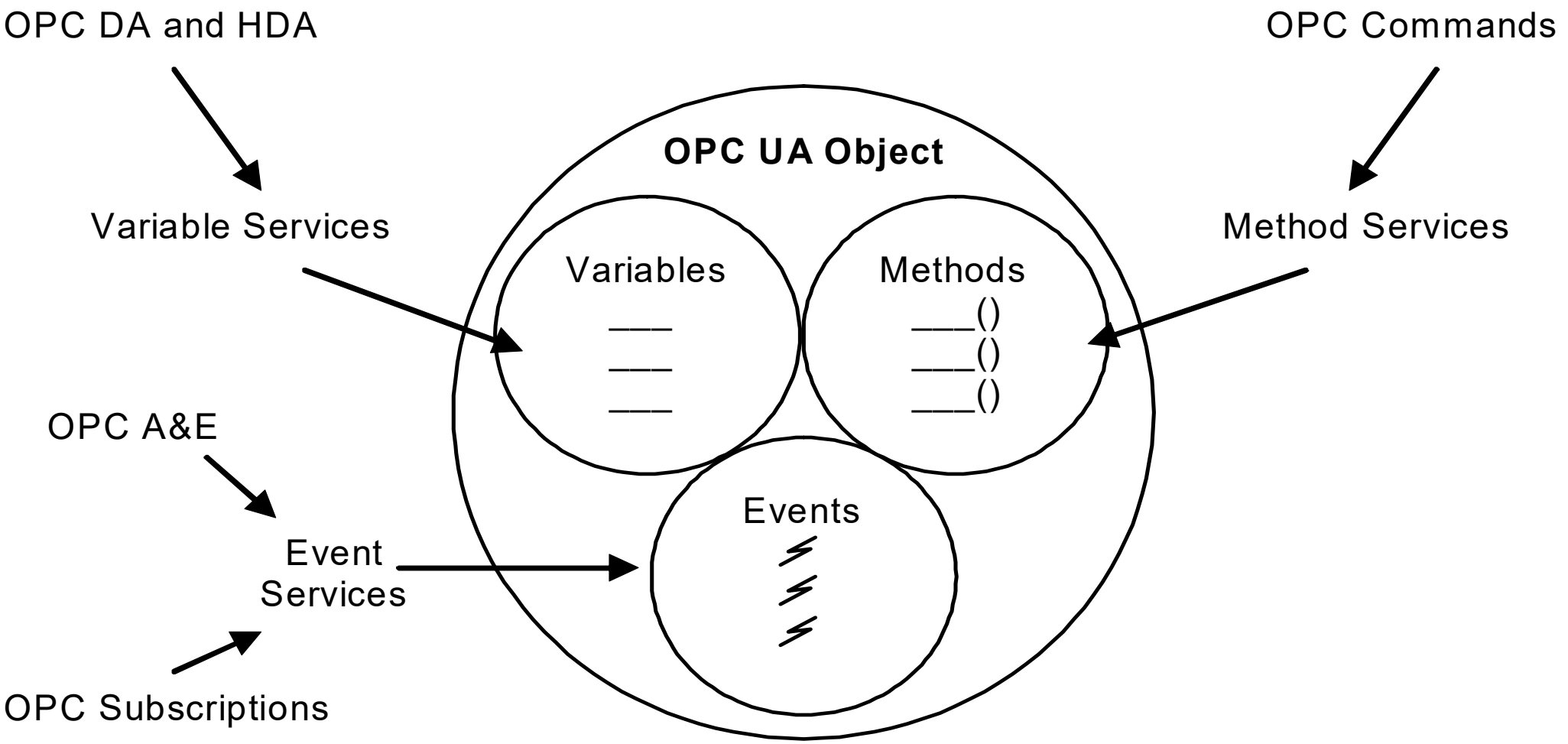
Thomas J.Burke. OPC Foundation

OPC UA Enabled Cloud Integration Architecture



http://www.icpdas.com/root/product/solutions/industrial_communication/m2m_iiot_server/ua-52x1.html

Unified Object Model



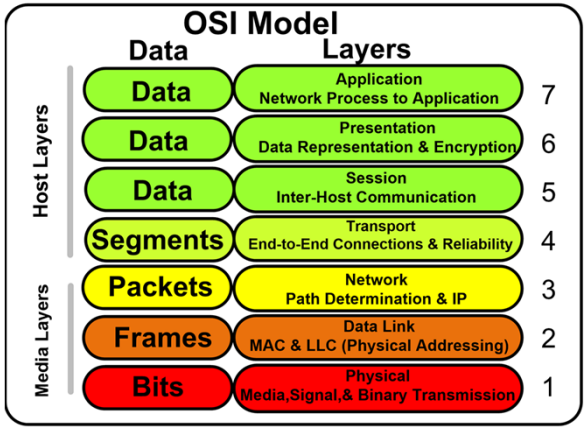
Thomas J.Burke. OPC Foundation

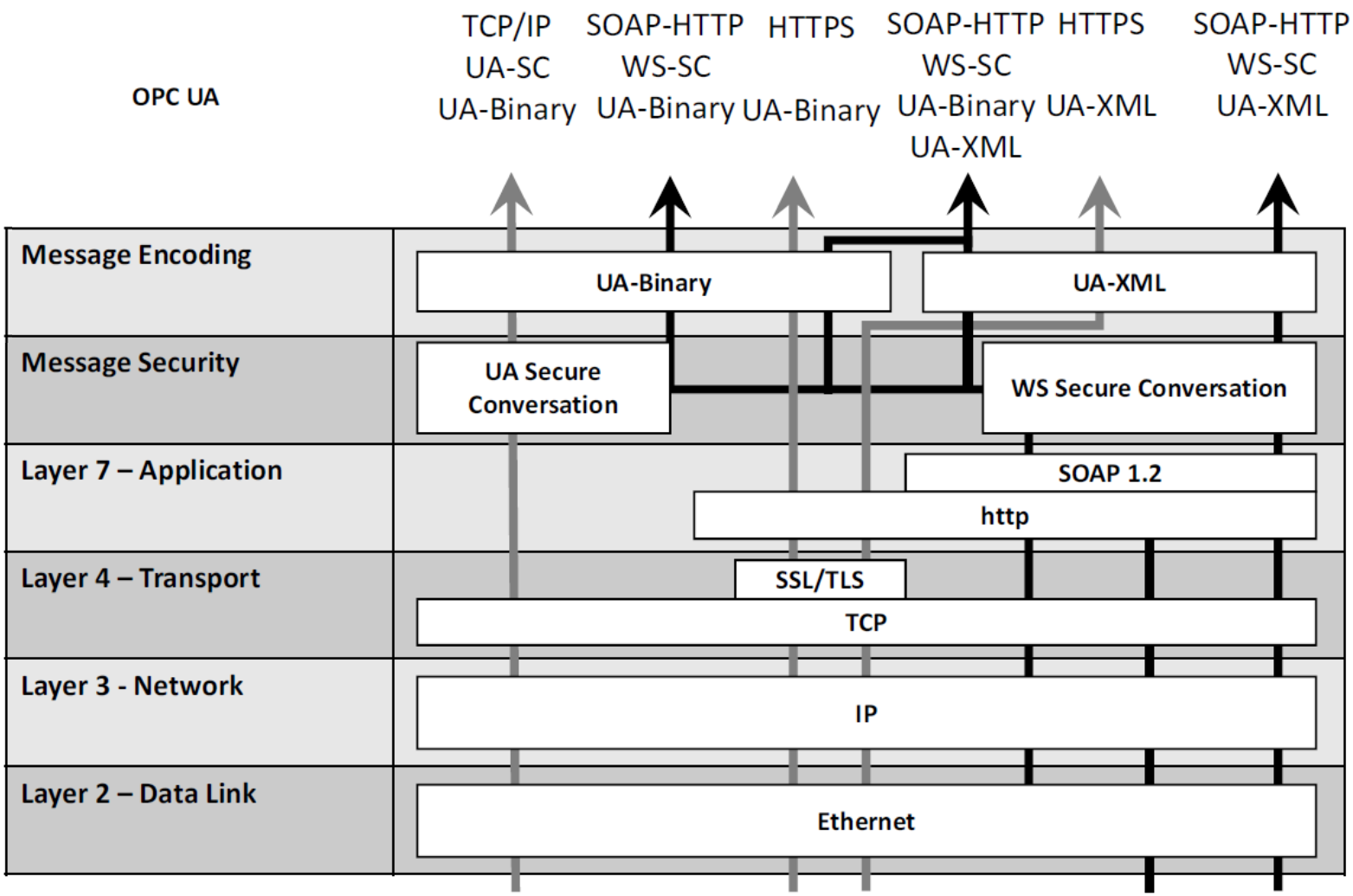
INDUSTRIAL COMMUNICATIONS:

OPC UA:- Open Process Control Unified Architecture

(Part II: terminology and taxonomy)

Categorization of OPC UA into the OSI Reference Model





"Utilizing OPC UA as comprehensive communication technology for Cyber Physical Production Systems". https://www.researchgate.net/publication/279530805

Common Interfaces = Less Programming

OPC Classic

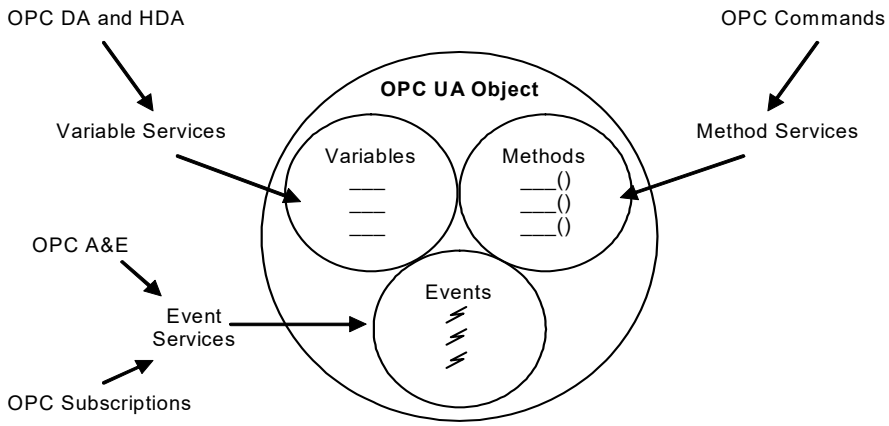
- DA:
  - IOPCSyncIO::Read
  - IOPCAsyncIO::Read
  - IOPCAsyncIO2::Read
  - IOPCAsyncIO3::Read
  - IOPCItemIO::Read
- A&E:
  - IOPCEventServer::GetConditionState
- HDA:
  - IOPCHDA_SyncRead::ReadRaw
  - IOPCHDA_SyncRead::ReadProcessed
  - IOPCHDA_SyncRead::ReadAtTime
  - IOPCHDA_SyncRead::ReadModified
  - IOPCHDA_SyncRead::ReadAttribute
  - IOPCHDA_AsyncRead::ReadRaw
  - IOPCHDA_AsyncRead::AdviseRaw
  - IOPCHDA_AsyncRead::ReadProcessed
  - IOPCHDA_AsyncRead::AdviseProcessed
  - IOPCHDA_AsyncRead::ReadAtTime
  - IOPCHDA_AsyncRead::ReadModified
  - IOPCHDA_AsyncRead::ReadAttribute

OPC Unified Architecture

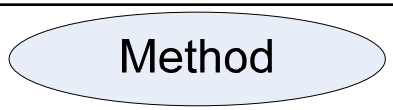
- Read
- ReadHistory

OPC UA Terminology

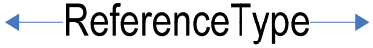
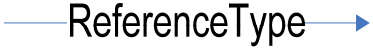
- UA "Tag" is now called a "Node". Nodes make up Information Models
- There are 8 classes of Nodes
  - Object, Variable, Method, ObjectType, VariableType, DataType, ReferenceType and View.
- Objects composed of
  - other Objects, Variables and Methods
- Variables:
  - Known as "Attributes" in OPC Classic (Value, Quality, Timestamp etc.)
- Methods:
  - Something to Execute
  - Parameters supported!
- References:
  - "Relationships" with other Nodes
- Events:
  - Notifications when "stuff" happens
- A set of Objects and the related information
  - referred to as its *AddressSpace*



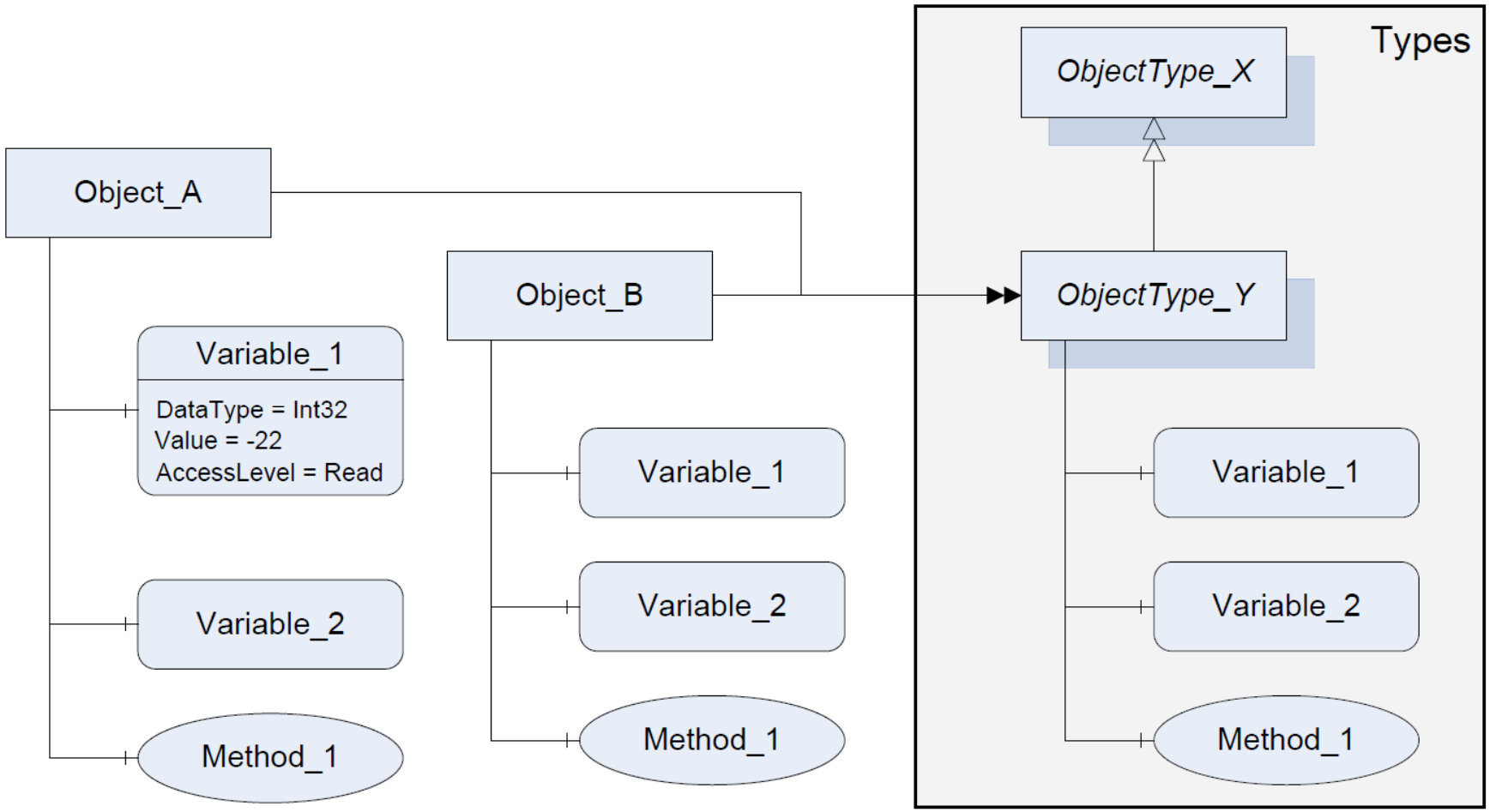
Graphical symbols for all NodeClasses

NodeClass	Graphical Representation	Comment
Object		Rectangle including text representing the string-part of the DisplayName of the Object. Can contain TypeDefinistion separated by "::" (e.g. Object1::Type1")
ObjectType		Shadowed rectangle including text representing the string-part of the DisplayName of the ObjectType. Abstract types use *italic*; concrete types not.
Variable		Rectangle with rounded corners including text representing the string-part of the DisplayName of the Variable. Can contain TypeDefinistion separated by "::" (e.g. Variable1::Type_a")
VariableType		Shadowed rectangle with rounded corners including text representing the string-part of the DisplayName of the VariableType. Abstract types use *italic*; concrete types not.
DataType		Shadowed hexagon including text representing the string-part of the DisplayName of the DataType. Abstract types use *italic*; concrete types not.
ReferenceType		Shadowed six-sided polygon including text representing the string-part of the DisplayName of the ReferenceType. Abstract types use *italic*; concrete types not.
Method		Oval including text representing the string-part of the DisplayName of the Method.
View		Trapezium including text representing the string-part of the DisplayName of the View.

Graphical notation for References

ReferenceType	Graphical Representation	Comment
Any symmetric ReferenceType		Symmetric *ReferenceTypes* are represented as lines between *Nodes* with closed and filled arrows on both sides pointing to the connected *Nodes*. Near the line has to be a text containing the string-part of the *BrowseName* of the *ReferenceType*.
Any asymmetric ReferenceType		Asymmetric *ReferenceTypes* are represented as lines between *Nodes* with a closed and filled arrow on the side pointing to the *TargetNode*. Near the line has to be a text containing the string-part of the *BrowseName* of the *ReferenceType*.
Any hierarchical ReferenceType		Asymmetric *ReferenceTypes* that are subtypes of *HierarchicalReferences* should be exposed the same way as asymmetric *ReferenceTypes* except that an open arrow is used.
HasComponent		The notation provides a shortcut for *HasComponent References* shown on the left. The single hashed line has to be near the *TargetNode*.
HasProperty		The notation provides a shortcut for *HasProperty References* shown on the left. The double hashed lines have to be near the *TargetNode*.
HasTypeDefinition		The notation provides a shortcut for *HasTypeDefinition References* shown on the left. The double closed and filled arrows have to point to the *TargetNode*.
HasSubtype		The notation provides a shortcut for *HasSubtype References* shown on the left. The double closed arrows have to point to the *SourceNode*.
HasEventSource		The notation provides a shortcut for *HasEventSource References* shown on the left. The closed arrow has to point to the *TargetNode*.

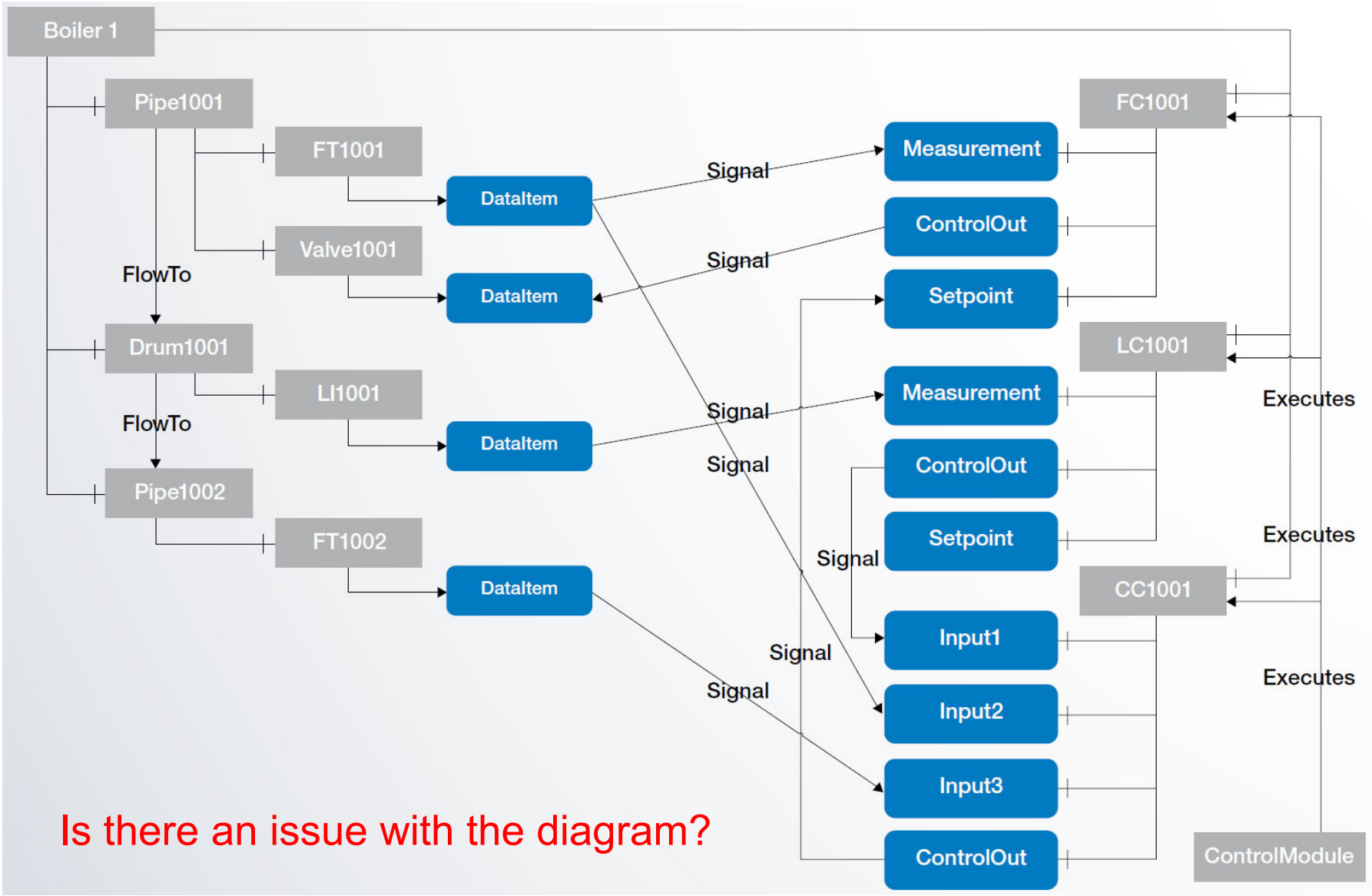
A Graphical Notation Example



Object_A and Object_B are instances of *ObjectType_Y* indicated by the HasTypeDefinition *References*. *ObjectType_Y* is derived from *ObjectType_X* indicated by the HasSubtype *Reference*. Object_A has components: Variable_1, Variable_2 and Method_1.

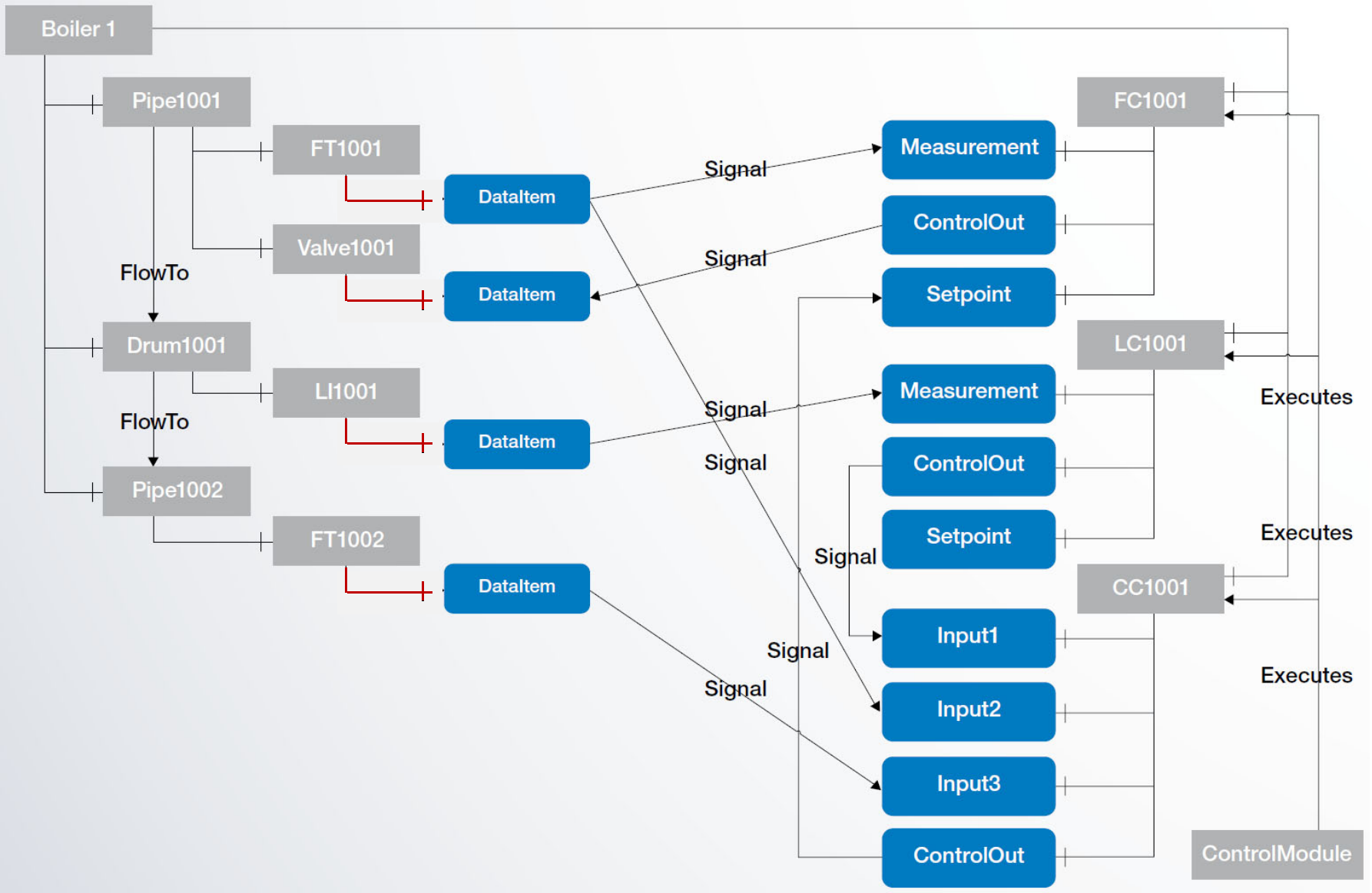
VDW and OPC Foundation: OPC UA Information Model or CNC Systems - Companion Specification. Release 1.0. July 05, 2017

UA modelling of a boiler



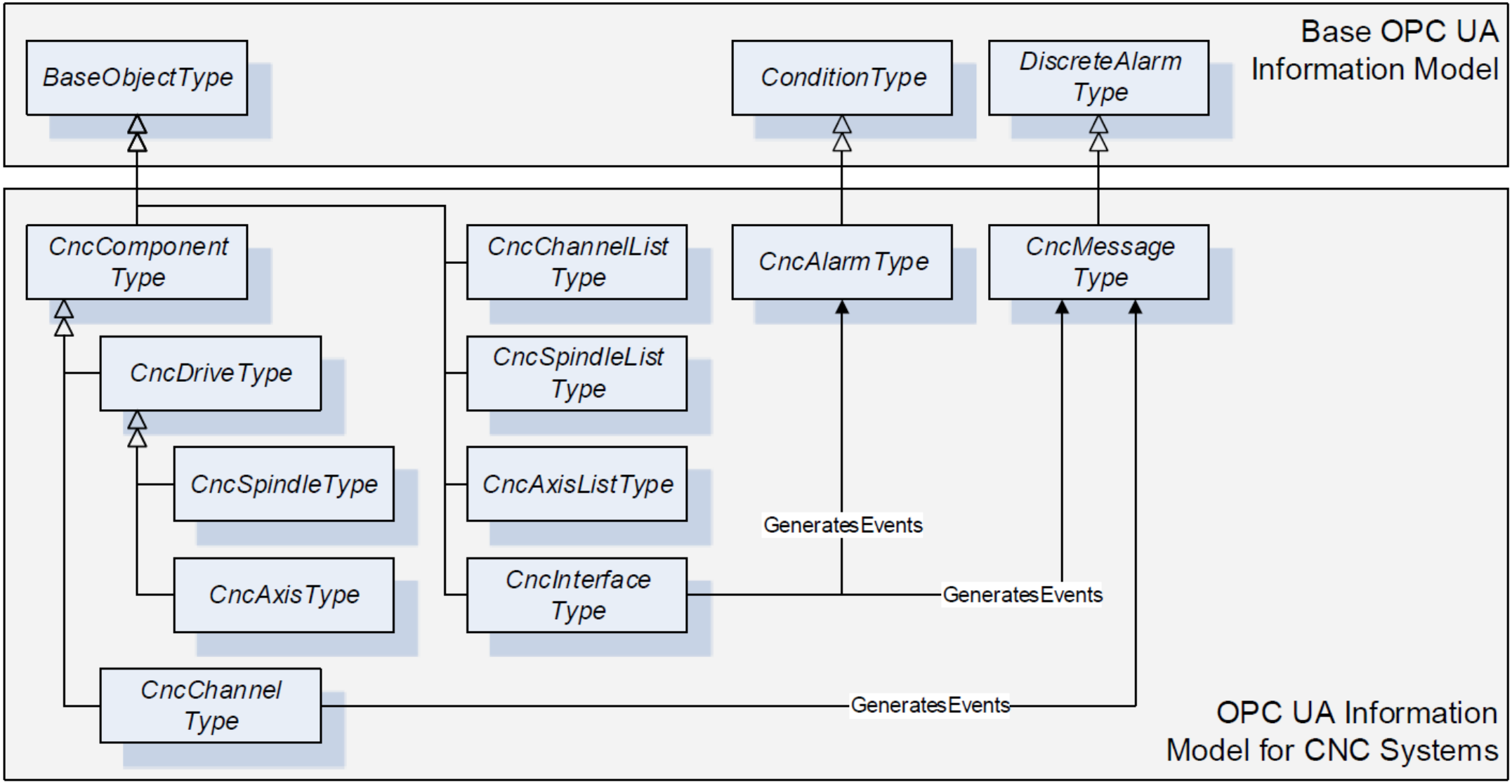
"OPC Unified Architecture - Interoperability for Industrie 4.0 and the Internet of Things". OPC Foundation

UA modelling of a boiler



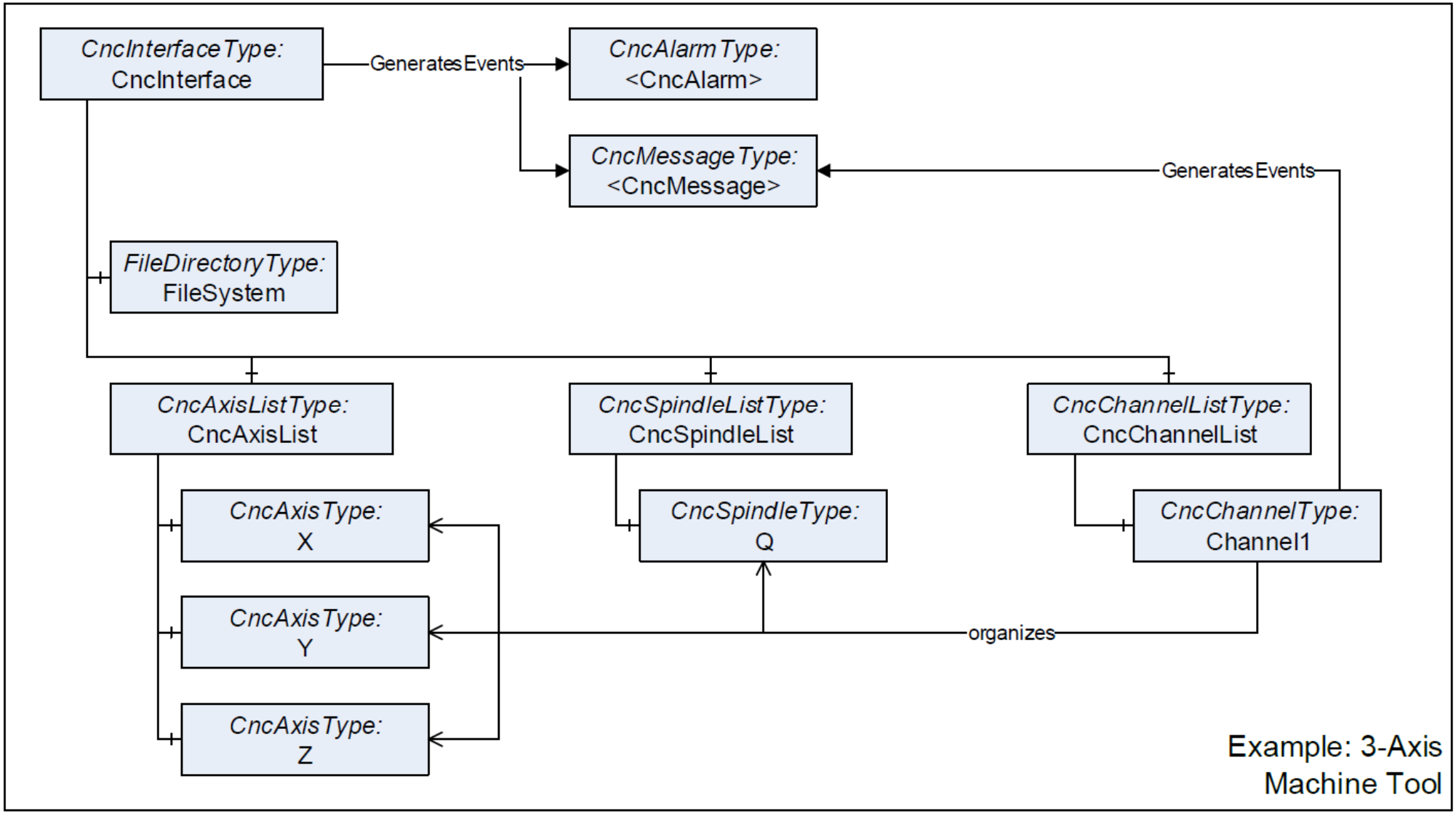
"OPC Unified Architecture - Interoperability for Industrie 4.0 and the Internet of Things". OPC Foundation

OPC UA Information Model for CNC Systems
(only the ObjectTypes and its References are presented)



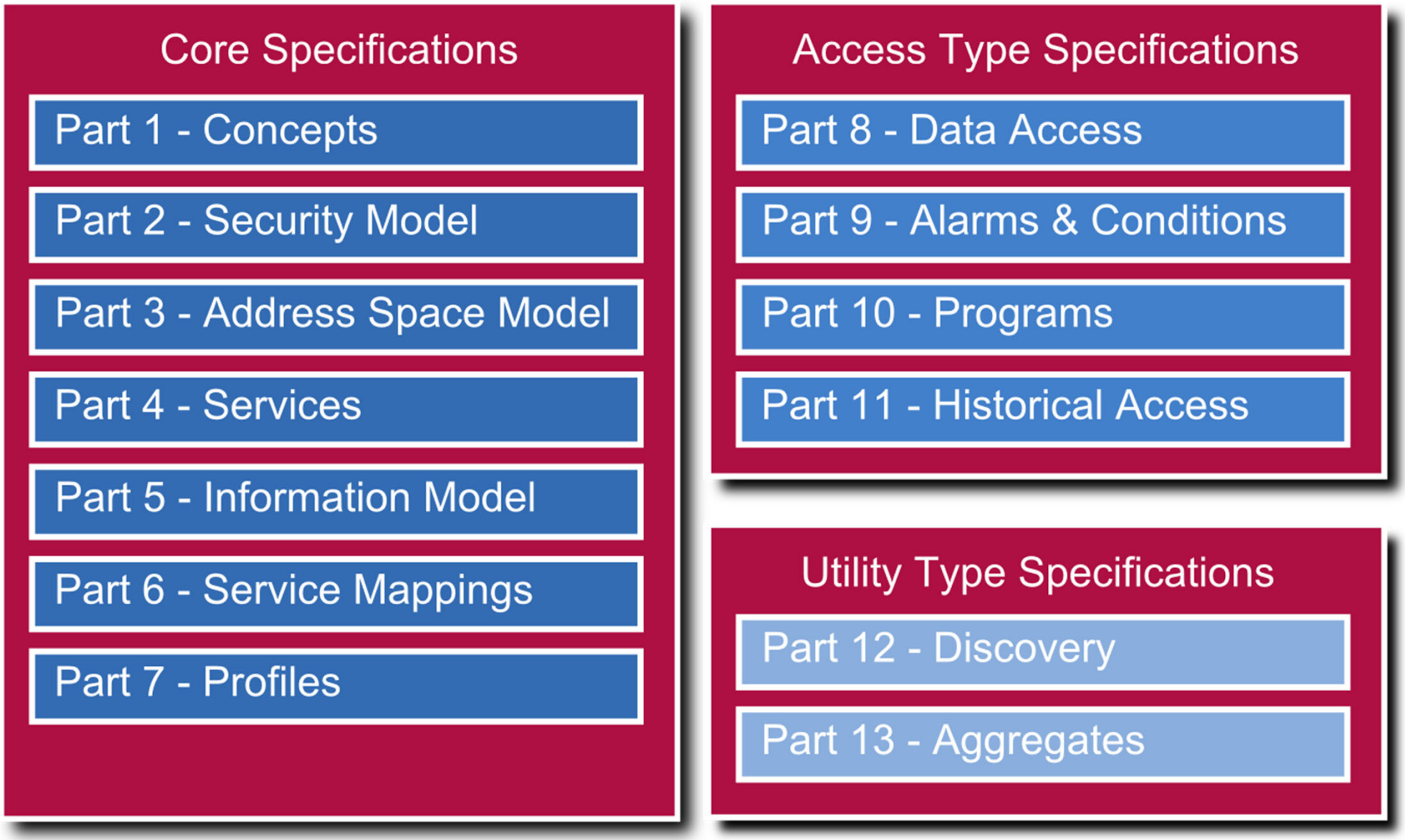
VDW and OPC Foundation: OPC UA Information Model or CNC Systems - Companion Specification. Release 1.0. July 05, 2017

A 3-axis CNC Machine Tool



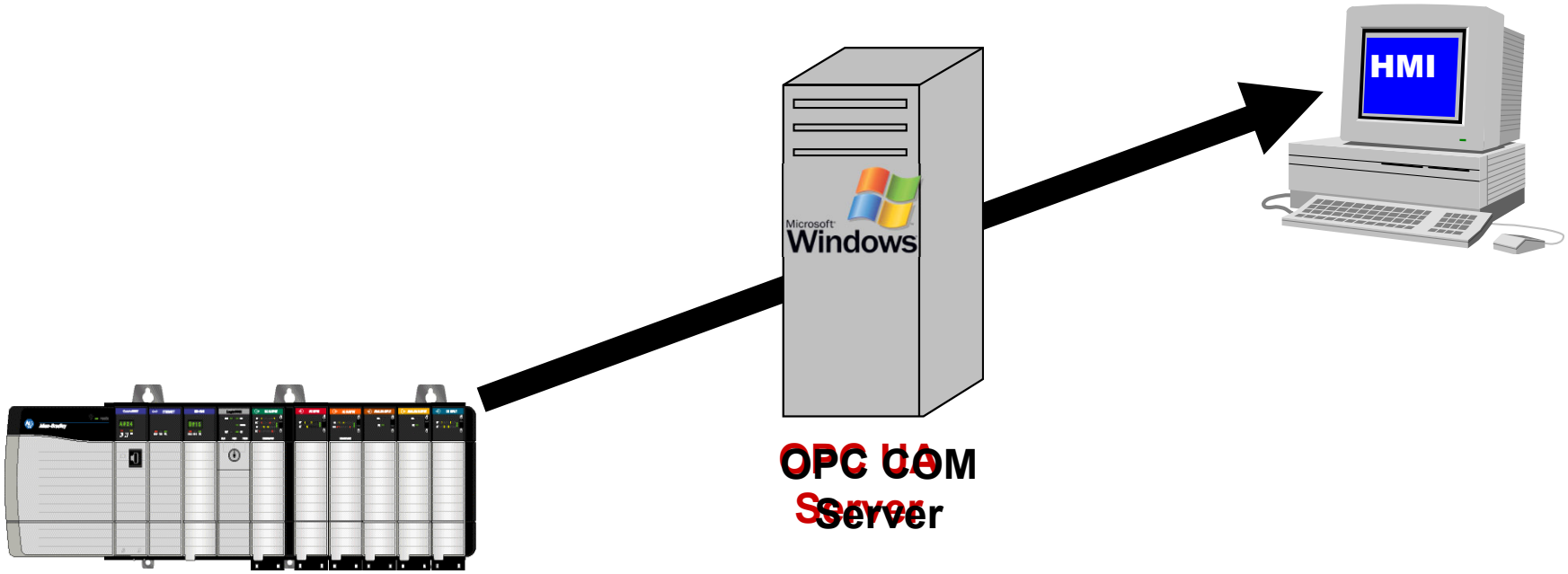
VDW and OPC Foundation: OPC UA Information Model or CNC Systems - Companion Specification. Release 1.0. July 05, 2017

IEC62541: OPC-UA specifications



OPC in device

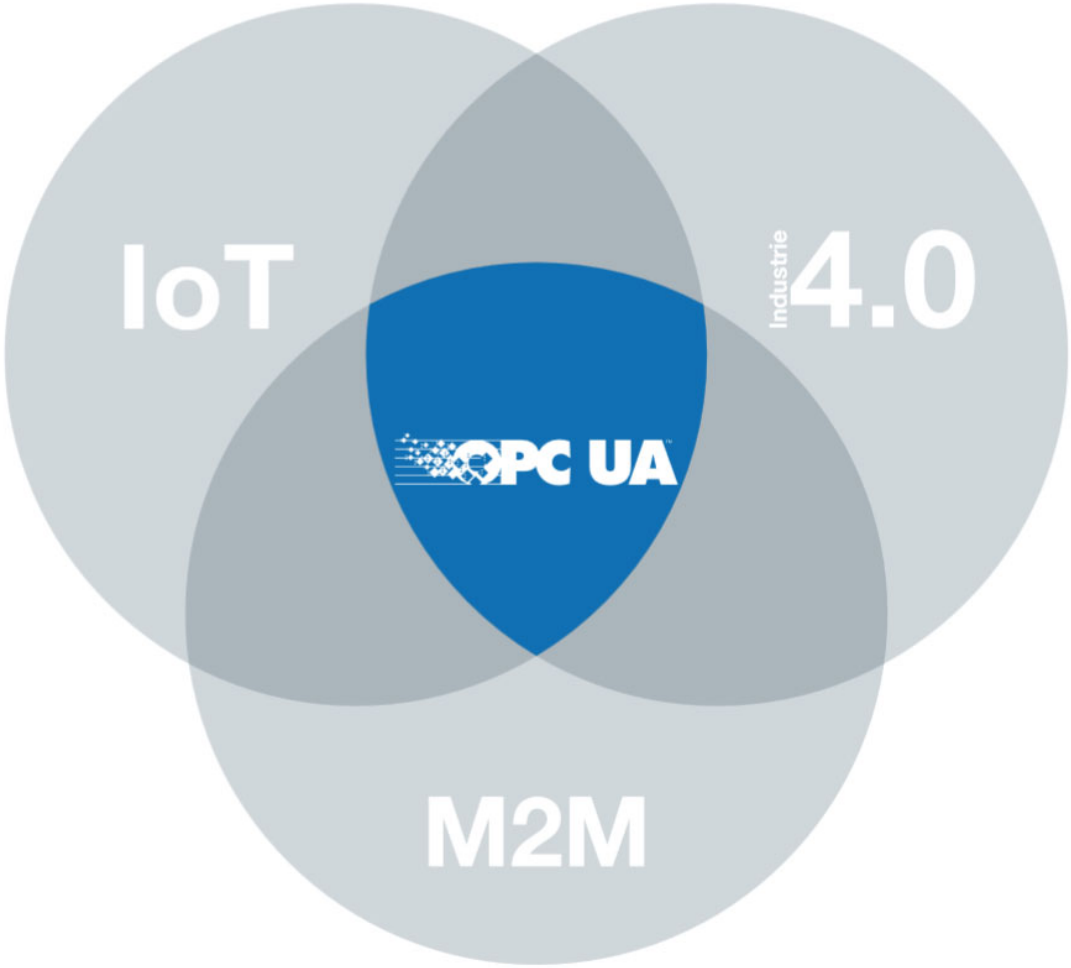
- PROBLEM: I want to supply an OPC interface to my device without the use of an external Windows PC



- SOLUTION: OPC-UA is cross-platform and embeddable

Thomas J.Burke. OPC Foundation

Remote Device Access (with OPC-UA) as the common intersection of M2M, IoT and Industry 4.0



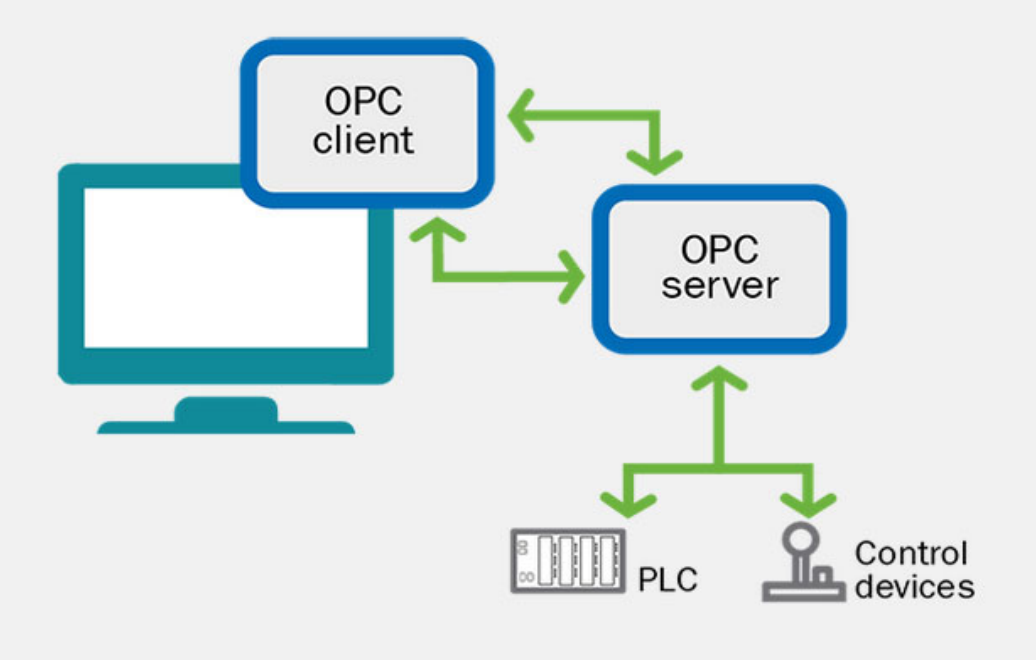
INDUSTRIAL COMMUNICATIONS:

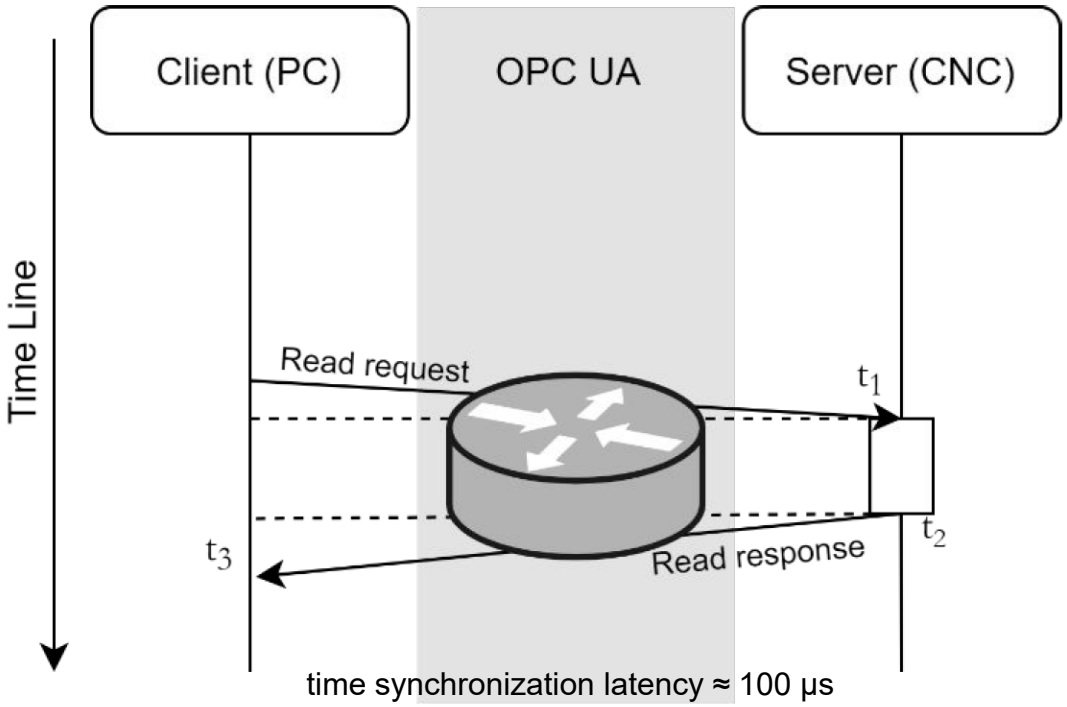
OPC UA:- Open Process Control Unified Architecture

(Part III: more on communications)

Client-Server mechanism

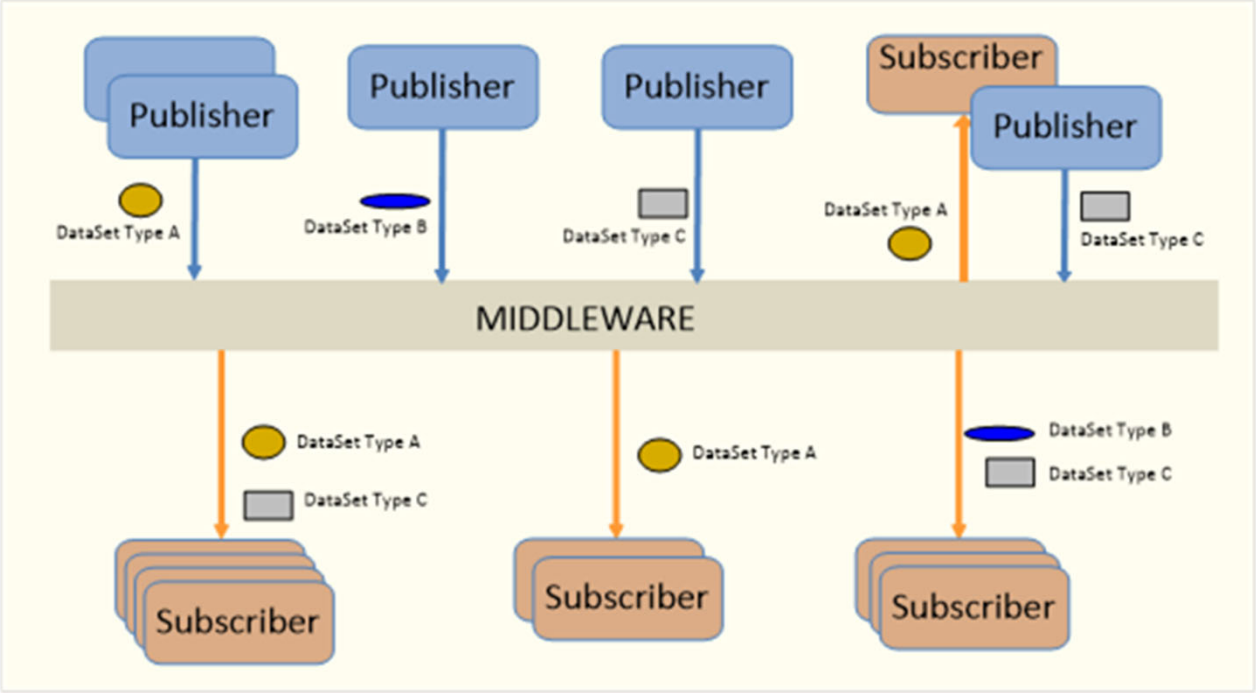
- Client is a requester, or consumer (*Read* service in OPC UA)
- Server is a respondent, or the producer (*Write* service in OPC UA)
- Bulk read/write operations





Publisher-Subscriber (PubSub) mechanism

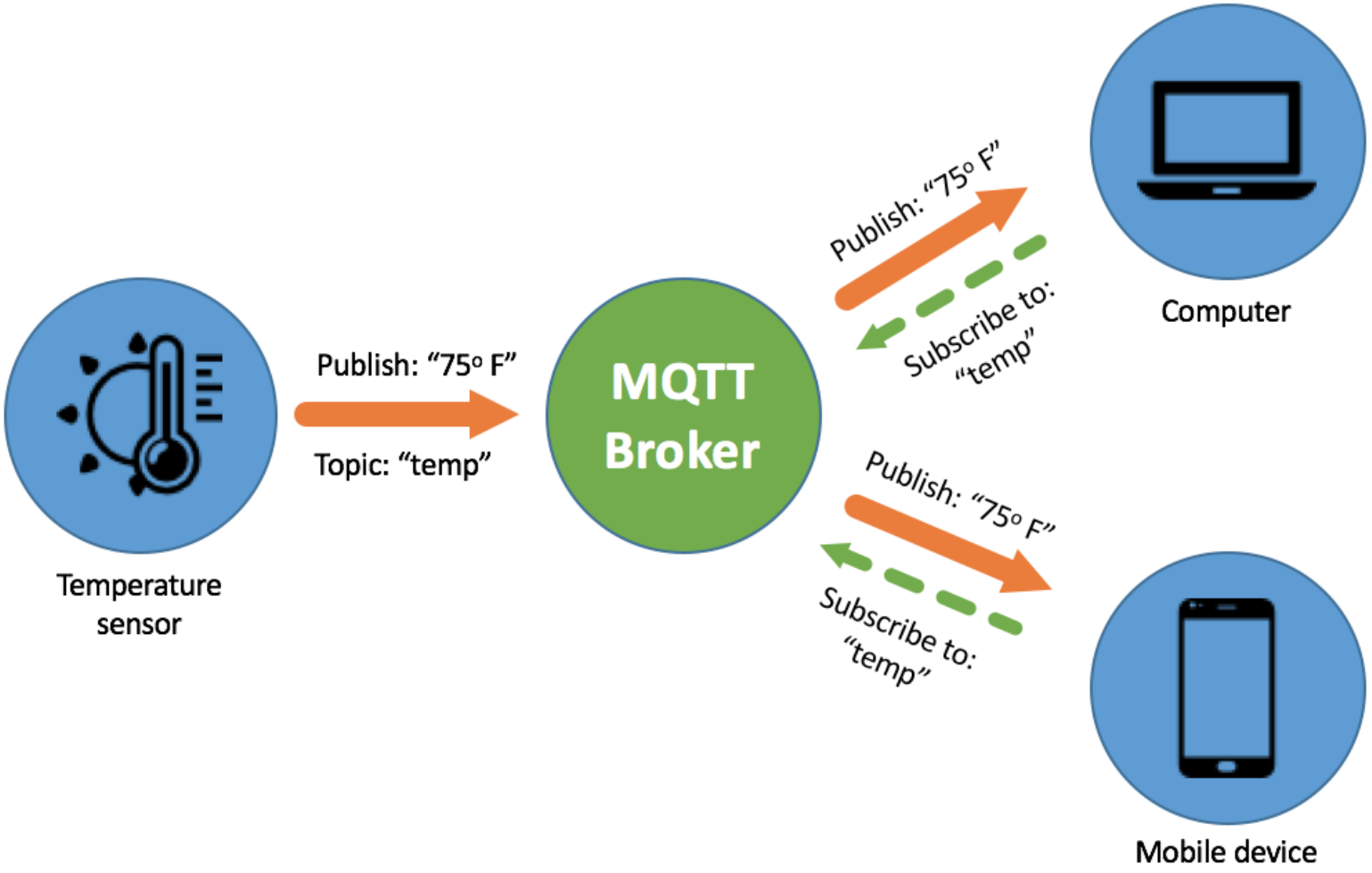
- OPC UA Part 14: PubSub (2018)
- Publishers send messages to a Message-Oriented Middleware (Broker) without knowledge of potential subscribers out there.
- Similarly, subscribers express interest in specific types of data and process messages that contain this data without knowledge of what publishers put out.
- Bulk read/write operations



Advantages of Pub/Sub over Request/Response

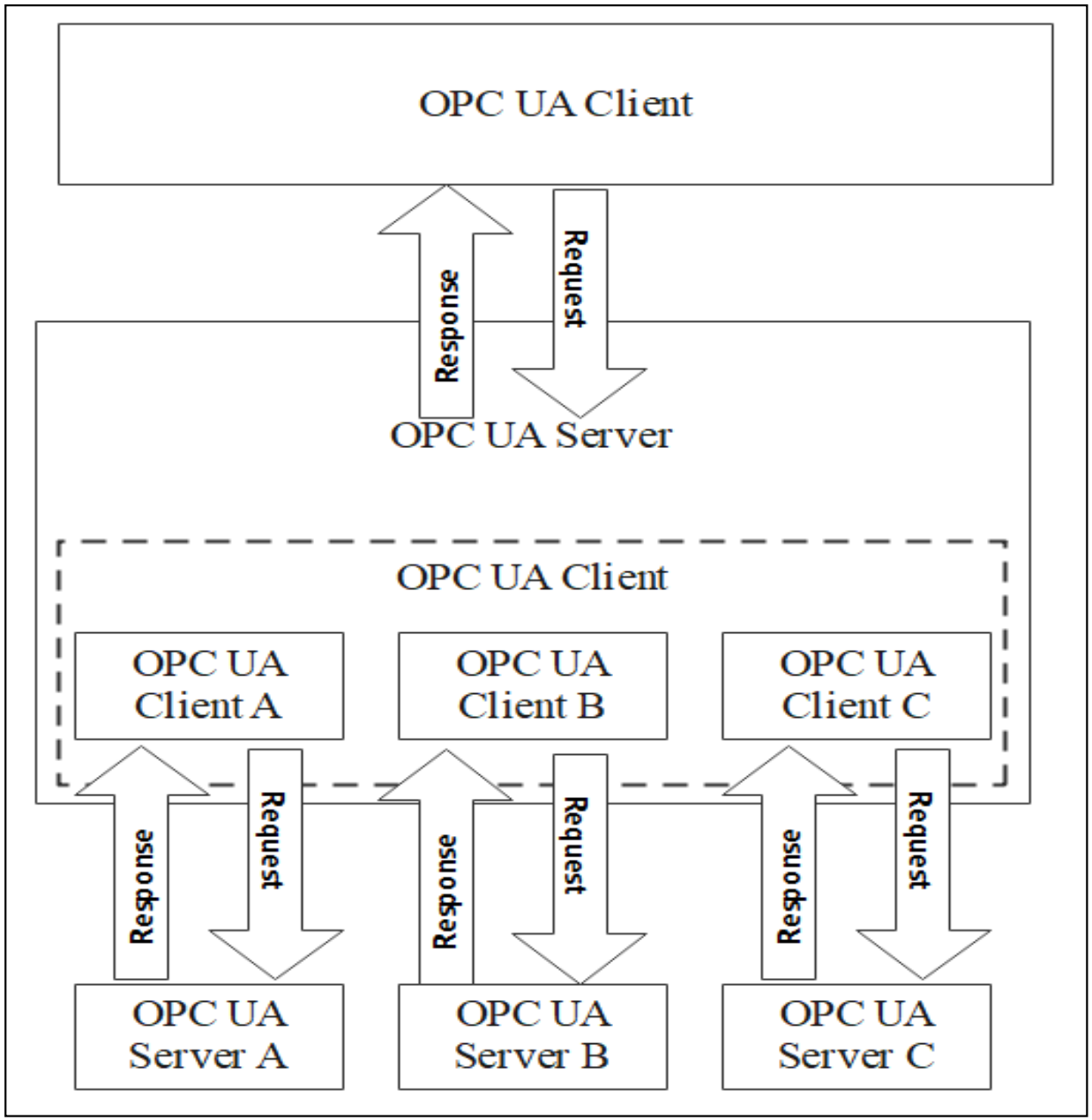
- The pub/sub enables public subscriptions from a large number of devices.
- Pub/sub for secured multicasting, many-to-one publishing, machine-to-machine communication, etc.
- A typical example for a publisher is a sensor node that can publish sensed information easily and needs no control functions – deeper level integration.
- Ability to deliver real-time guarantees.
- OPC UA pub/sub provides a framework that is capable of simultaneously supporting multiple protocols such as MQTT.
- Pub/sub support queuing while transferring data.

MQTT (Message Queuing Telemetry Transport)



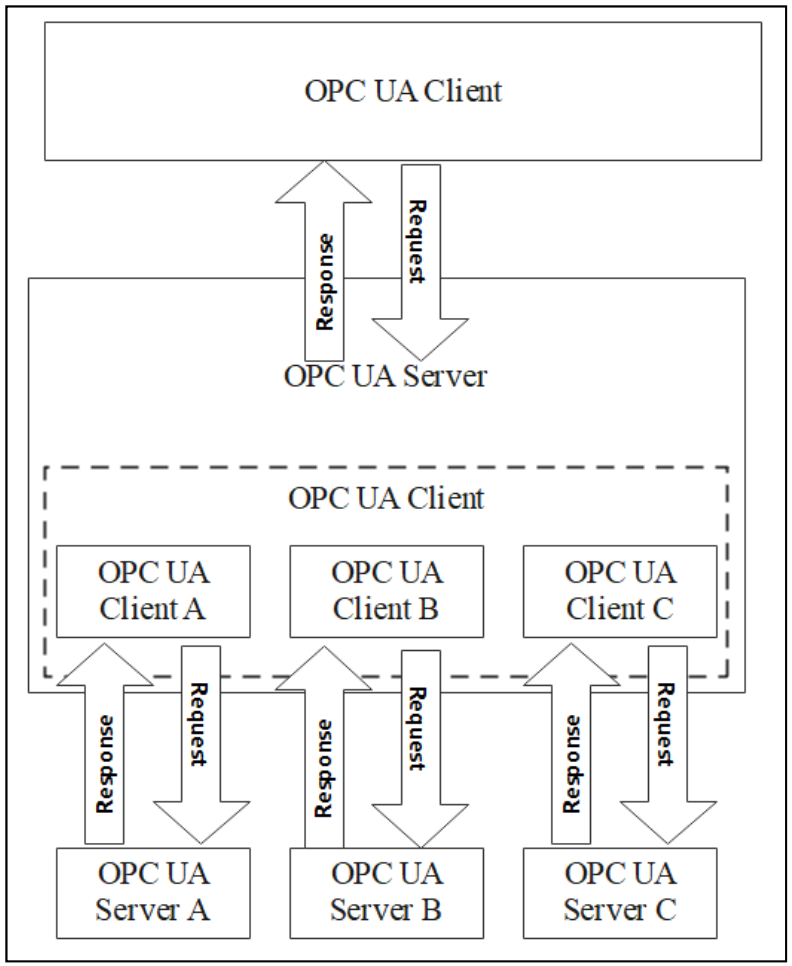
OPC UA Aggregate Server

- Multiple servers are pinned to a solitary server.
- Multiple client instances within the aggregate server will connect and retrieve all the data, and then these retrieved data, with the help of an embedded server, is ready to share complete data into a single client.
- Reduce load on clients by making all the data available through a single connection.

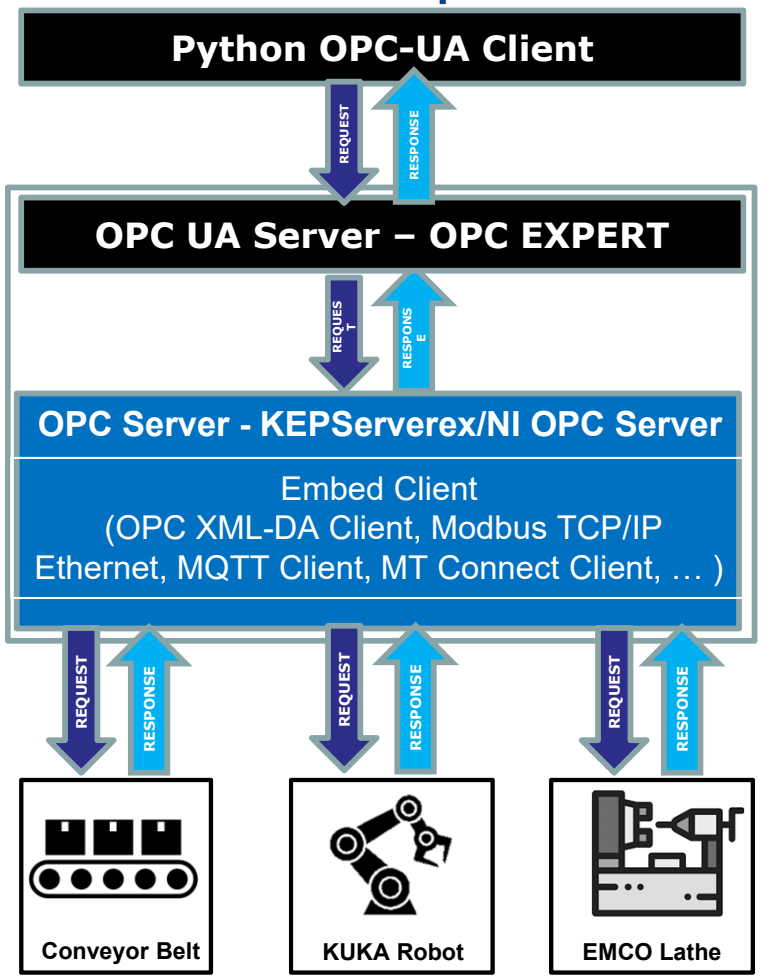

OPC UA Client
Response
Request
OPC UA Server
OPC UA Client
OPC UA Client A
OPC UA Client B
OPC UA Client C
Response
Request
Response
Request
Response
Request
OPC UA Server A
OPC UA Server B
OPC UA Server C

Aggregating OPC Server – an example

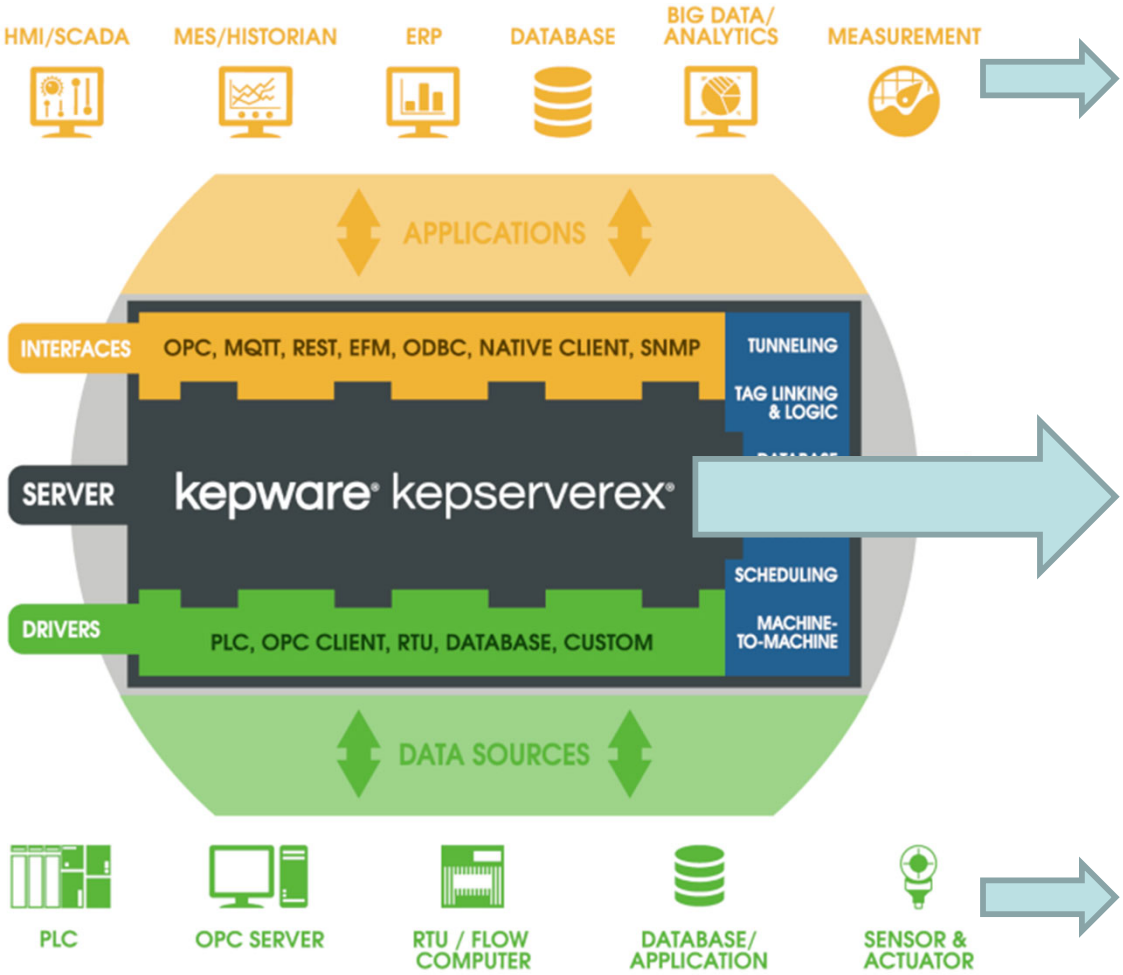
Aggregating Server



An example



KEPServerEX

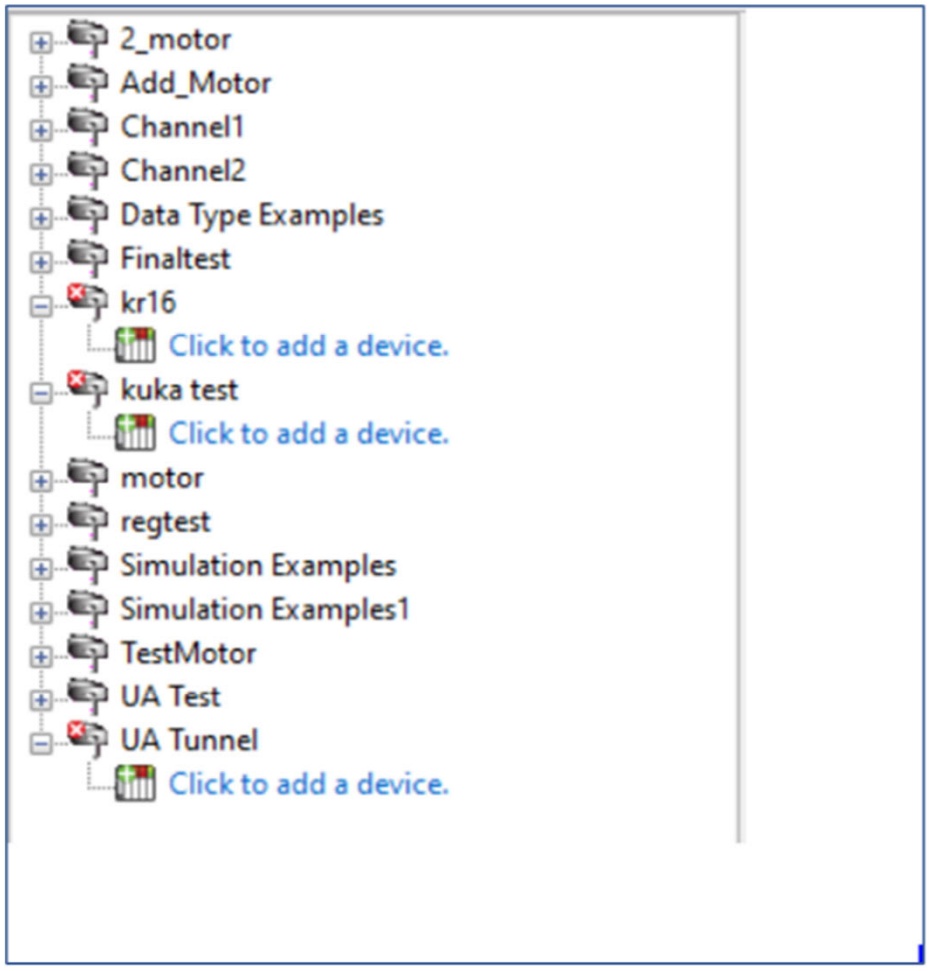


- Different applications need different interfaces to communicate with the devices

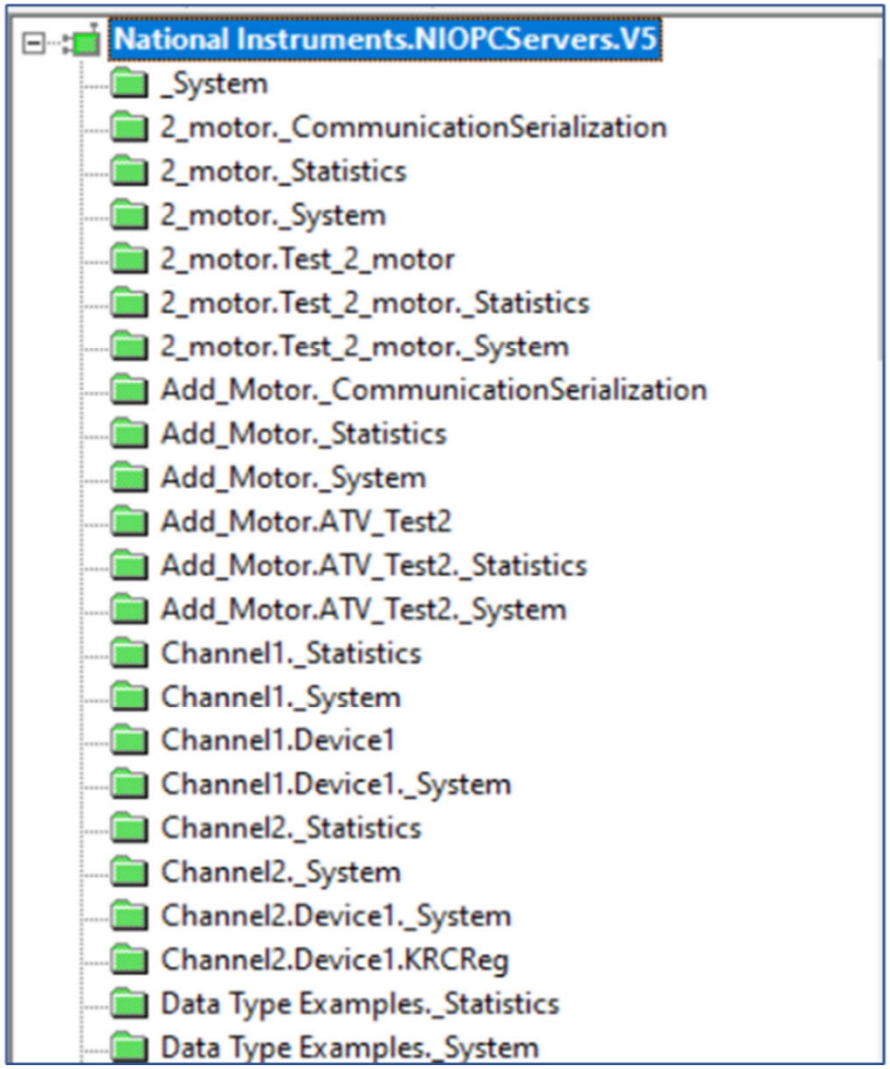
- Connecting Communication Platforms
- KEPServerex has many built-in drivers, such as OPC DA, OPC UA, MQTT, MTConnect
- The role of the KEPServerex is to convert the drivers of all connected devices into a single OPC server

- Different devices have different drivers and communication protocols

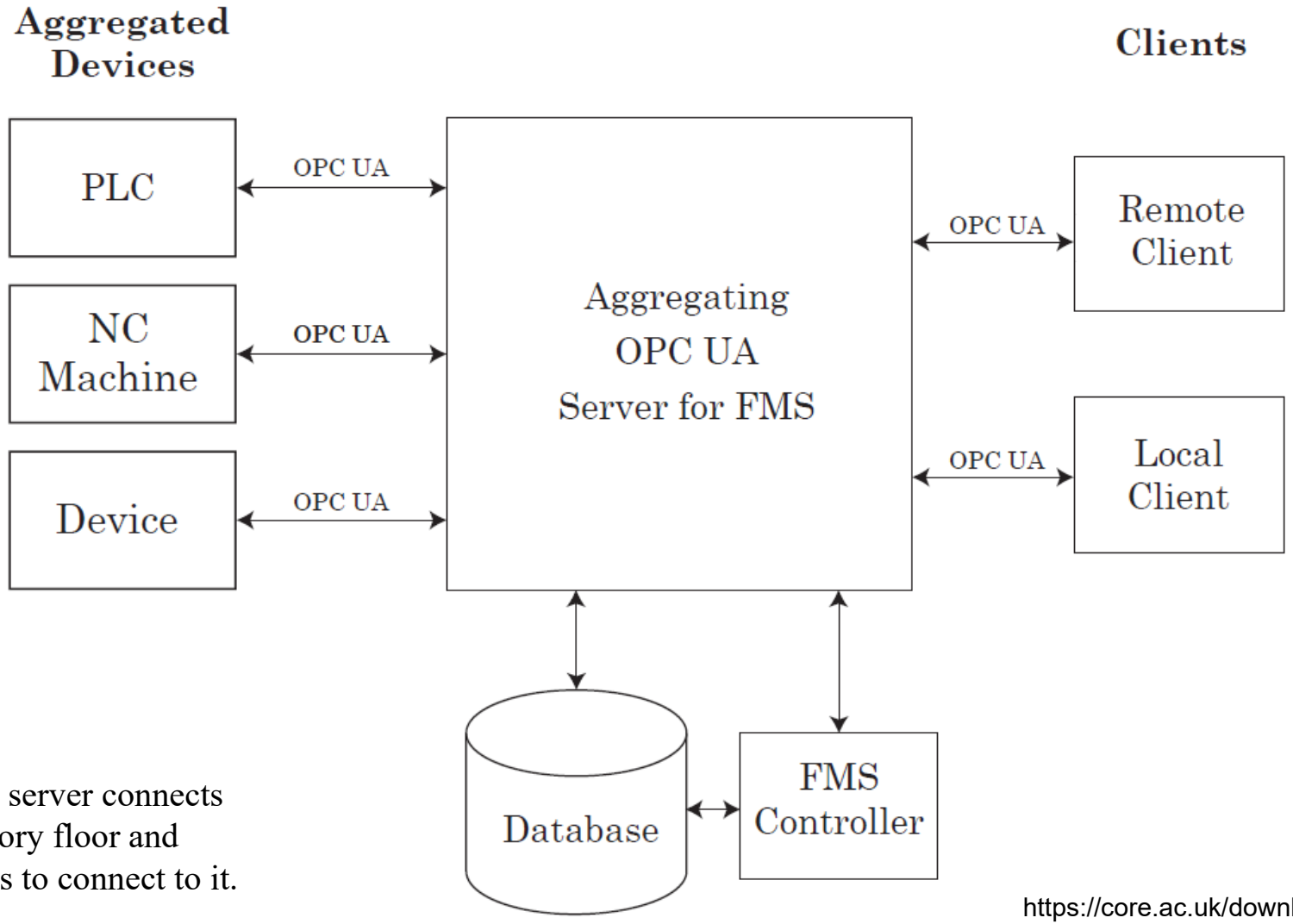
Aggregate Server (e.g. NI OPC Server)



Single Client Instance



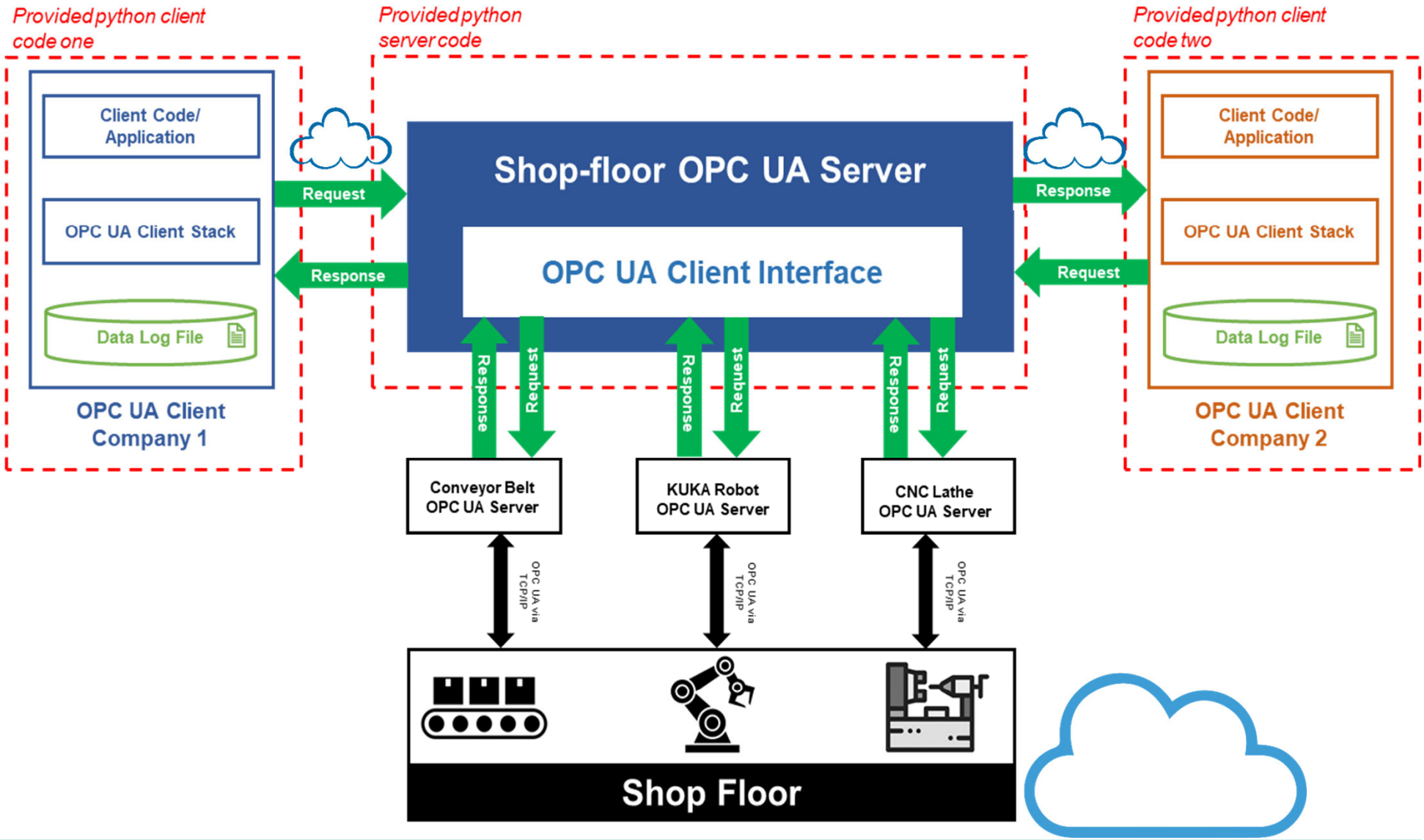
Aggregating OPC UA Server for Flexible Manufacturing Systems



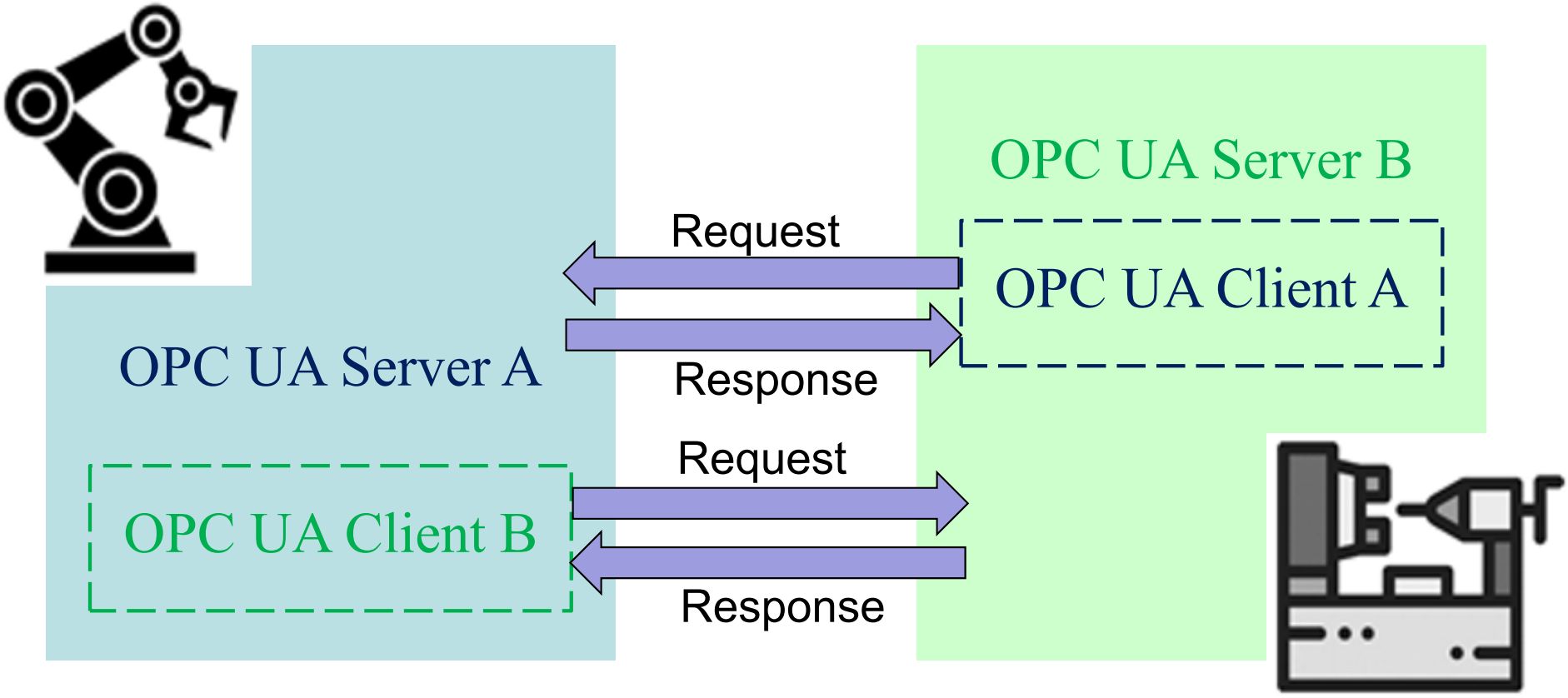
OPC UA aggregating server connects to devices on the factory floor and allows external clients to connect to it.

https://core.ac.uk/download/pdf/80716585.pdf

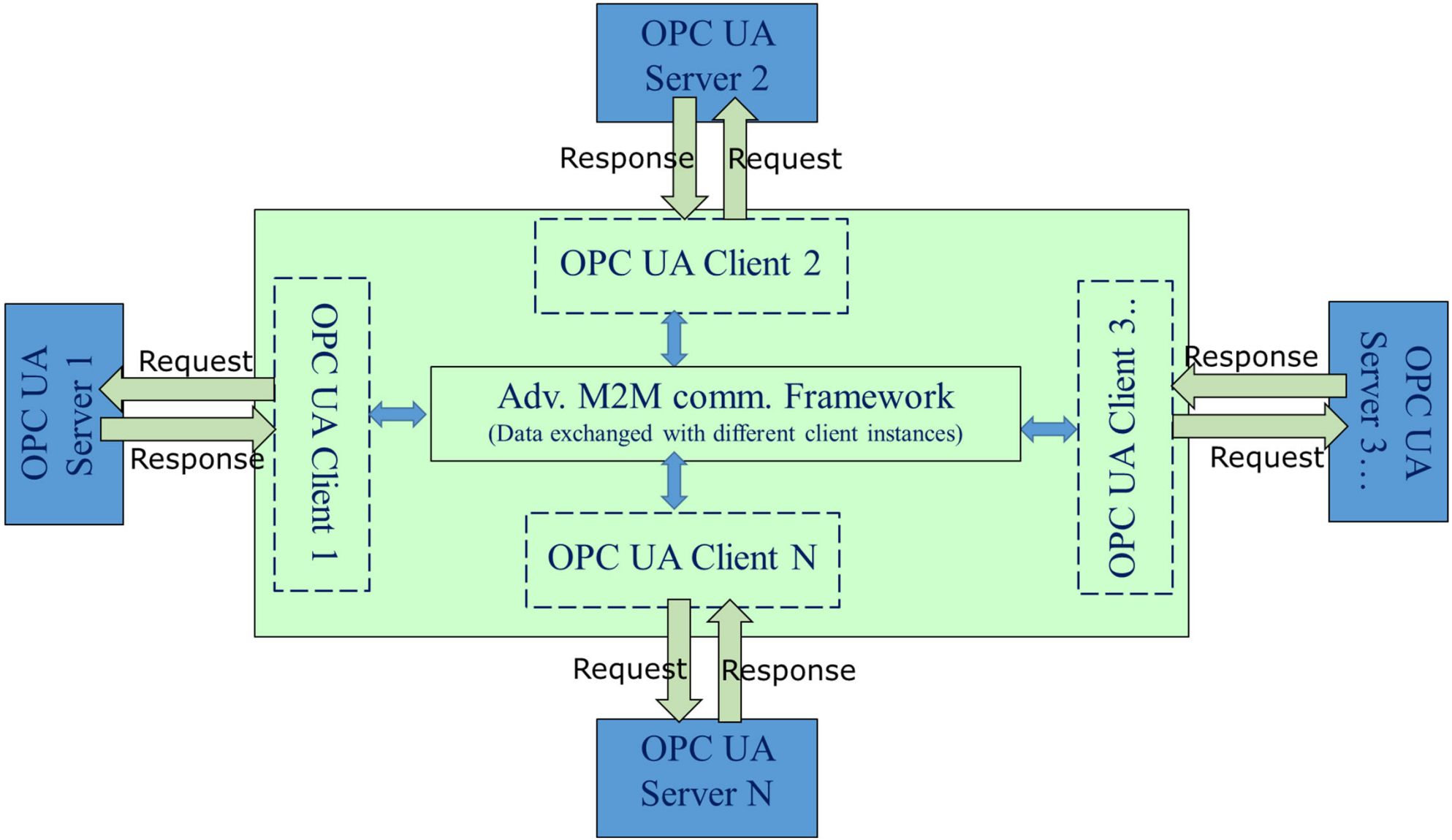
Aggregating OPC UA Server for a Smart Manufacturing Cell



More on M2M communications (server-to-server)



More on M2M communications (client-to-client)



Re-cap

- Industrial Communications
  - Fieldbus
  - Communications topologies
  - Open System Interconnection (OSI)
  - OPC Classic …
    - Benefits
    - Components
    - MS Windows technologies
  - OPC UA …
    - Web technology
    - Services
    - Terminology (…)
    - Information model (graphical and XML)
  - Different industrial communication modes with OPC UA