

**Análisis comparativo entre IEC 61131-3 e IEC 61499 y su papel en la evolución de la  
automatización industrial distribuida**

**Julio César Jurado Pérez**

**Trabajo de grado para optar el título de Ingeniero Mecatrónico**

**Director**

**Kelly Johanna Niño Sandoval**

**Ingeniería Mecatrónica**

**Universidad Santo Tomás, Seccional Bucaramanga**

**División de Ingenierías y Arquitectura**

**Facultad de Ingeniería Mecatrónica**

**2026**

### **Dedicatoria**

El presente trabajo fue realizado gracias al apoyo, la confianza y el amor que mis padres depositaron en mí durante todo este proceso. No ha sido un camino fácil; a lo largo de mi formación profesional se presentaron diversos inconvenientes y obstáculos que pusieron a prueba mi determinación. Sin embargo, su cariño, confianza y constante respaldo fueron la fuerza que me permitió seguir adelante.

Gracias a ellos, hoy estoy culminando una de las etapas más importantes de mi vida: la conclusión de mi carrera profesional. Por ello, les dedico estas palabras con profundo agradecimiento por estar siempre a mi lado, por creer en mí incluso en los momentos difíciles y por haberme criado con los valores y principios necesarios para nunca rendirme ante la adversidad.

De igual manera, quiero expresar mi más sincero agradecimiento a todos mis profesores. Gracias por cada consejo, enseñanza y experiencia compartida a lo largo de este camino. Sus aportes no solo contribuyeron a mi formación académica y profesional, sino también a mi crecimiento personal. Con sus enseñanzas ampliaron mi visión del mundo y me brindaron herramientas que me permitirán afrontar con mayor preparación los retos que encontraré como futuro profesional.

A todos ustedes, muchas gracias por la confianza, la fe y el esfuerzo que invirtieron en mí durante esta etapa de mi vida.

**Contenido**

|                                                                                        |    |
|----------------------------------------------------------------------------------------|----|
| 1. Introducción .....                                                                  | 8  |
| 2. Marco teórico .....                                                                 | 9  |
| 2.1 Fundamentos de la automatización industrial y evolución de sus arquitecturas ..... | 9  |
| 2.2 Estándar IEC 61131-3 .....                                                         | 11 |
| 2.3 Estándar IEC 61499.....                                                            | 12 |
| 2.4 Conceptos clave de IEC 61499.....                                                  | 13 |
| 2.4.1 Bloques de Función (Function Blocks) .....                                       | 14 |
| Los Bloques de Interfaz de Servicio (SIFB).....                                        | 14 |
| 2.4.2 Modelo orientado a eventos frente al ciclo de escaneo .....                      | 14 |
| 2.4.3 Aplicaciones, Dispositivos y Recursos.....                                       | 15 |
| 2.4.4 Distribución del control .....                                                   | 16 |
| 3. Metodología .....                                                                   | 16 |
| 4. Análisis de Datos.....                                                              | 18 |
| 4.1 Modelo de ejecución.....                                                           | 18 |
| 4.2 Arquitectura del sistema .....                                                     | 21 |
| 4.3 Reutilización y modularidad.....                                                   | 24 |
| 4.4 Interoperabilidad.....                                                             | 27 |
| 4.5 Implementación industrial .....                                                    | 31 |
| 6. Conclusiones .....                                                                  | 37 |
| Referencias.....                                                                       | 39 |

**Lista de tablas**

|                                                                 |    |
|-----------------------------------------------------------------|----|
| <b>Tabla 1.</b> <i>Comparación entre ambos estándares</i> ..... | 36 |
|-----------------------------------------------------------------|----|

**Lista de figuras**

|                                                                                                     |    |
|-----------------------------------------------------------------------------------------------------|----|
| <b>Figura 1.</b> <i>Diagrama del ciclo de escaneo de IEC 61131-3</i> .....                          | 19 |
| <b>Figura 2.</b> <i>Diagrama de activación por eventos en IEC 61499</i> .....                       | 20 |
| <b>Figura 3.</b> <i>Comparación arquitectónica entre IEC 61131-3 e IEC 61499</i> .....              | 24 |
| <b>Figura 4.</b> <i>Comparación de modularidad y reutilización en IEC 61131-3 e IEC 61499</i> ..... | 27 |
| <b>Figura 5.</b> <i>Comparativa de Interoperabilidad de IEC 61131-3 y 61499</i> .....               | 30 |
| <b>Figura 6.</b> <i>Limitantes de adopción de IEC 61499</i> .....                                   | 33 |

### Resumen

La evolución de los sistemas de automatización industrial ha generado la necesidad de contar con arquitecturas de control cada vez más flexibles, escalables e interoperables. En este contexto, los estándares internacionales han desempeñado un papel fundamental en la definición de modelos para el desarrollo de software en control industrial. El presente trabajo realiza un análisis comparativo entre los estándares IEC 61131-3 y IEC 61499, ampliamente reconocidos en el ámbito de la programación de controladores lógicos programables y sistemas de automatización distribuidos.

La investigación se desarrolló mediante un enfoque documental y comparativo, basado en la revisión de literatura técnica, documentos normativos y estudios académicos relacionados con ambos estándares. El análisis se centra en aspectos clave como el modelo de programación, la arquitectura de software, la capacidad de distribución del control, la interoperabilidad entre dispositivos y su nivel de adopción en entornos industriales.

Los resultados evidencian que IEC 61131-3 continúa siendo el estándar dominante en la industria debido a su madurez tecnológica, amplia adopción y fuerte soporte por parte de fabricantes de PLC. Sin embargo, IEC 61499 introduce un enfoque orientado a eventos y una arquitectura más distribuida que busca mejorar la portabilidad del software y la interoperabilidad entre sistemas. Finalmente, se concluye que, aunque ambos estándares presentan ventajas y limitaciones, la adopción de nuevas arquitecturas de control dependerá tanto de los avances tecnológicos como de las necesidades reales de la industria.

*Palabras clave:* automatización industrial, PLC, arquitectura de control, interoperabilidad, estándares industriales

### **Abstract**

The evolution of industrial automation systems has created the need for control architectures that are increasingly flexible, scalable, and interoperable. In this context, international standards have played a fundamental role in defining models for industrial control software development. This study presents a comparative analysis between IEC 61131-3 and IEC 61499, two widely recognized standards in the field of programmable logic controller programming and distributed automation systems.

The research was conducted using a documentary and comparative approach, based on the review of technical literature, normative documents, and academic studies related to both standards. The analysis focuses on key aspects such as programming models, software architecture, control distribution capabilities, interoperability between devices, and their level of adoption in industrial environments.

The results show that IEC 61131-3 remains the dominant standard in industry due to its technological maturity, widespread adoption, and strong support from PLC manufacturers. However, IEC 61499 introduces an event-driven approach and a more distributed architecture aimed at improving software portability and system interoperability. Finally, it is concluded that although both standards present advantages and limitations, the adoption of new control architectures will depend on technological developments and the practical needs of the industry.

*Keywords:* industrial automation, PLC, control architecture, interoperability, industrial standards

## 1. Introducción

La automatización industrial ha estado en constante cambio durante los últimos años, ajustándose a las necesidades de los sistemas de manufactura, buscando nuevas innovaciones donde se optimicen la productividad, flexibilidad e integración tecnológica, en esa búsqueda de progreso los estándares internacionales han sido pieza clave en proporcionar metodologías y estructuras para el desarrollo de sistemas de control industrial.

Dentro de estos estándares, IEC 61131-3 se ha consolidado como la principal referencia para la programación de Controladores Lógicos Programables (PLC), gracias a su confiabilidad, amplia adopción industrial y soporte por parte de los fabricantes, sin embargo, el avance de conceptos asociados a la Industria 4.0, los Sistemas Ciberfísicos (CPS) y el Internet Industrial de las Cosas (IIoT) ha impulsado la búsqueda de arquitecturas de control más distribuidas, interoperables y escalables.

Como respuesta a estas nuevas necesidades surge el estándar IEC 61499, el cual propone un enfoque basado en bloques de función y ejecución orientada a eventos, buscando facilitar el desarrollo de sistemas de automatización distribuidos y reducir la dependencia de plataformas propietarias, no obstante, a pesar de las ventajas que presenta desde el punto de vista conceptual, su adopción industrial continúa siendo limitada en comparación con IEC 61131-3.

Ante este panorama, surge la siguiente pregunta de investigación: ¿el estándar IEC 61499 representa un reemplazo, un complemento o una evolución del estándar IEC 61131-3 dentro de los sistemas modernos de automatización industrial?

Con el propósito de responder esta pregunta, el presente trabajo desarrolla un análisis comparativo de ambos estándares mediante una revisión documental de literatura técnica y académica especializada, para ello se consideran aspectos como el modelo de ejecución, la



arquitectura del sistema, el modularidad, la interoperabilidad y su nivel de implementación industrial, con el fin de identificar el papel que desempeña IEC 61499 frente a las necesidades actuales de la automatización industrial.

## **2. Marco teórico**

### **2.1 Fundamentos de la automatización industrial y evolución de sus arquitecturas**

Desde su introducción a finales de la década de 1960, los Controladores Lógicos Programables (PLC) han constituido un elemento central en la automatización industrial, proporcionando la capacidad de procesamiento digital y la robustez necesarias para gobernar la maquinaria en los entornos de fabricación [3, 18, 28, 33]. Tradicionalmente, la implementación de estos dispositivos se ha basado en arquitecturas de control centralizadas, en las cuales una única unidad principal asume las tareas de planificación, adquisición de señales y ejecución de la lógica [4, 12, 23]. Estas arquitecturas más clásicas se organizan en estructuras jerárquicas ya establecidas en el estándar ISA-95, en donde las operaciones a ejecutar se separan en capas delimitadas [11,24]. En el nivel inferior se hallan los elementos de campo, luego los de tiempo real, de seguido se encuentran los sistemas de control de supervisión y adquisición de datos (SCADA), cuya función es hacer recolección de información, monitoreo de procesos de forma continua y brindar interfaces humano-máquina (HMI); con esto se garantiza la buena visualización y la toma de decisiones de los operarios [11, 32]. Este enfoque centralizado ha sido bastante confiable y funcional para la industria desde hace décadas, ya que ofrece tiempos de respuesta acordes con los ciclos continuos de ejecución [3,15].

En la actualidad muchos sistemas de producción exigen una flexibilidad mayor para poder ser adaptados a ciclos de vida de productos más cortos y una mayor diversidad de variantes; el análisis de este documento observa varios desafíos relacionados con la estructura de arquitectura centralizada [3,19]. Ya que en estos últimos la estructura de los procesos se concentra en un controlador central y esto incrementa la vulnerabilidad del sistema ante fallos de un único punto (single-point of failure) [4,6,23]. A su vez, realizar la más mínima intervención en el software requiere de un análisis extenso para contemplar qué variables globales de código se podrían ver afectadas, además de que es obligatorio parar por completo el proceso del sistema por lo anterior mencionado [6,13]. También se debe tener en cuenta que es fundamental la combinación de redes y equipos de distintos proveedores y estos mismos exigen herramientas propietarias de su autoría, lo que genera dependencia hacia ciertas marcas o lo que se conoce como vendor lock-in [3,20]; por esas razones, el modelo cíclico con sus variables globales, su dificultad de exportar código a distintos dispositivos y las limitantes de sincronización entre estos, hace que su método de desarrollo sea bastante rígido [6, 9, 15, 18].

Para solucionar estos inconvenientes, se ha experimentado poco a poco con una estructura más descentralizada [6,30]. Con la evolución hacia la Industria 4.0 y los elementos que esta conlleva, como Sistemas Ciberfísicos (CPS), el Internet Industrial de las Cosas (IIoT), la computación en el borde (edge computing) y las plataformas virtualizadas en la nube, se ha llegado a una transición que busca que la toma de decisiones se haga de forma dispersa en una red compuesta por distintos dispositivos inteligentes y sistemas multiagente [3, 7, 21, 29,10,25].

Esta nueva vertiente en las arquitecturas se basa en los conceptos de modularidad e interoperabilidad.

La modularidad consiste en que un sistema se pueda dividir en componentes o elementos de software autónomos, lo que permite que cada uno de estos elementos pueda ser reutilizado o escalado de forma sencilla sin importar que haya algún cambio en el proceso, proporcionando lo que se conoce como diseño seguro (secure by design) [11,15,26]. La interoperabilidad tiene como objetivo que cada uno de los dispositivos físicos puedan comunicarse e integrarse en el sistema de forma sencilla y estandarizada, independiente del proveedor al cual pertenezcan.

Estas características proporcionan entornos perfectos para una arquitectura más maleable y adaptable a las necesidades del proceso, sentando los principios del estándar IEC 61499 [4,6].

## **2.2 Estándar IEC 61131-3**

El estándar IEC 61131-3 fue concebido como referencia en el diseño de software y la programación en PLC alrededor de 1993, esto debido a la fragmentación provocada por los lenguajes propietarios de cada fabricante [18,28]. Este define varios lenguajes de programación como, por ejemplo, el Texto Estructurado (ST), Lista de Instrucciones (IL), el Diagrama de Escalera (LD), el Diagrama de Bloques de Funciones (FBD), el Diagrama de Funciones Secuenciales (SFC), entre otros [3, 9, 33]. Todos ellos se emplean para armar Unidades de Organización de Programa (POUs); gracias a esto, los ingenieros han podido consolidar la lógica de programación de una forma sencilla y práctica [14, 33].

La estructura del estándar se sustenta en un modelo cíclico o de escaneo continuo, en donde el controlador opera en un bucle sin interrupción donde se adquiere una señal de entrada, se ejecuta secuencialmente la lógica del programa y por último se actualiza el estado de la variable de salida [1, 2, 3]. Esta lógica brinda tiempos de respuesta predecibles y se ha establecido en la industria

como una bastante confiable, tanto por los ingenieros debido a la familiaridad que tienen a la hora de programar como por las empresas y su larga trayectoria invirtiendo y capacitando en dicha infraestructura [18, 30].

A pesar de su histórico éxito, el estándar IEC 61131-3 presenta limitaciones cuando se intenta escalar hacia la manufactura distribuida [3, 4]. Su núcleo, al estar enfocado en arquitecturas centralizadas, presenta dificultades notables cuando se intenta adoptar en situaciones de sistemas distribuidos [15,18,23]. Fragmentar un programa en múltiples controladores genera problemas o se vuelve un trabajo de ingeniería demasiado tedioso. Es bastante complejo en ese escenario garantizar la toma de datos en tiempo real o la sincronización de los ciclos de escaneo de los distintos equipos; sumándole el hecho de la dependencia del intercambio de datos por variables globales, se entorpece demasiado la estructura de un código distribuido [8, 9, 13]. Todas estas limitantes de la estructura del estándar no permiten la fácil integración de componentes en la nube y la conexión con componentes colaborativos, por lo tanto, se buscó desarrollar una nueva forma para facilitar estas nuevas necesidades [12, 28, 31].

### **2.3 Estándar IEC 61499**

El estándar IEC 61499 aparece como una propuesta que busca desligarse de las limitantes del IEC 61131-3 con el control centralizado y las de compatibilidad impuestas por los fabricantes con sus equipos (vendor lock-in) [17, 18, 30]. Se define como una arquitectura abierta enfocada al desarrollo de sistemas de automatización industrial distribuidos [1, 6].

Su propuesta busca adaptarse a las nuevas exigencias de la Industria 4.0, brindando a los ingenieros herramientas avanzadas como la portabilidad del software, la interoperabilidad entre

dispositivos de múltiples proveedores, modularidad y configurabilidad de los equipos [4, 6, 17, 30].

El cambio más relevante que muestra frente a su antecesor es que no usa un sistema de ejecución cíclica, sino uno enfocado a eventos, en donde todo se encuentra en una red descentralizada de componentes de software modulares denominados Bloques de Función (FB). Estos bloques permanecen inactivos hasta que un evento específico del proceso los active [3, 18, 23, 30]. Esta nueva metodología facilita considerablemente la creación y programación de procesos asíncronos en redes, optimizando la configuración y los recursos utilizados [24,30].

Para garantizar que la arquitectura sea completamente distribuida, esta busca separar la lógica de software de la de hardware [3, 17]. Al dividir ambos componentes, los desarrolladores pueden diseñar la lógica digital sin tener tantas trabas con la estructura física de los equipos en la que se operará al final. Esto facilita que distintas partes del programa se puedan distribuir, mapear y ejecutar en múltiples nodos de la planta [6, 17, 18].

El desarrollo basado en bloques, la ejecución asíncrona por eventos y la independencia del hardware brindan a esta arquitectura una versatilidad bastante amplia [11, 17], permitiendo modificar secciones de código del sistema de forma local en la red sin necesidad de parar por completo el proceso, ofreciendo agilidad y practicidad para lograr conceptos de manufactura avanzada como el Plug & Produce [17, 27].

## **2.4 Conceptos clave de IEC 61499**

El estándar IEC 61499 se estructura principalmente mediante bloques de función, los cuales solo se activan cuando sucede un evento en el sistema en específico por el cual dicho bloque esté programado para reaccionar. Es importante mencionar que este estándar separa el programa

de los dispositivos físicos donde son ejecutados, es decir, el software del hardware, por lo que se pueden crear programas versátiles y adaptables según las necesidades.

#### ***2.4.1 Bloques de Función (Function Blocks)***

El elemento primordial para el estándar IEC 61499 son los bloques de función (FB). Estos internamente se dividen en dos secciones: una que se encarga del flujo de eventos y la otra de los datos [1, 6, 14, 18]. Para que ambas partes se sincronicen se utiliza la construcción lógica “WITH” mediante una asociación temporal; de esta manera, se actualizan los datos cuando ocurre el evento asociado [3, 18, 29].

Los bloques de función se dividen en tres tipos principales:

**Los Bloques de Función Básicos (BFB)** Almacenan lógica de control y realizan toma de decisiones de acuerdo con la situación del proceso [1, 14, 18, 23].

**Los Bloques de Función Compuestos (CFB)** Agrupan varios bloques de función en uno solo; esto ayuda a organizar mejor el sistema y favorece la reutilización de partes del programa [6, 15, 18, 27].

**Los Bloques de Interfaz de Servicio (SIFB)** permiten la comunicación con elementos externos como actuadores, sensores o redes de comunicación [3, 11, 21]. Cabe resaltar que el estándar incluye una herramienta conocida como adaptador, la cual agrupa varias señales en una misma interfaz para garantizar la buena conexión entre distintos bloques [1, 18, 29].

#### ***2.4.2 Modelo orientado a eventos frente al ciclo de escaneo***

El modelo del estándar IEC 61131-3 tiene de base el ciclo de escaneo continuo, en donde el controlador adquiere los datos de entrada, ejecuta el programa y actualiza las salidas sin

interrupción [2, 3, 18, 24]. A diferencia del estándar IEC 61499, cuyo modelo se orienta a los bloques de función y los eventos, donde el bloque se encontrará desactivado hasta que un evento en específico lo active [4, 18, 30]. Este último presenta ventajas significativas para sistemas modernos, ya que al no estar constantemente escaneando se optimiza el uso de recursos computacionales [7, 24], se enfocan estos recursos a actividades que requieran monitoreo continuo y se facilita la ejecución de procesos asíncronos y concurrentes. Esto permite que el programa pueda reaccionar de forma rápida y efectiva ante respuestas de diferentes nodos de la red [24, 30].

#### ***2.4.3 Aplicaciones, Dispositivos y Recursos***

Para poder trabajar con sistemas de automatización de gran tamaño, el estándar IEC 61499 organiza sus componentes en tres elementos principales: Aplicaciones, Dispositivos y Recursos [16, 18].

Una Aplicación es una red de bloques de función conectados entre sí, cuya finalidad es definir el comportamiento general del sistema. Una característica importante es que esta se diseña de forma independiente al hardware donde será ejecutada, permitiendo una mayor flexibilidad al momento de implementar la solución [4, 18, 30].

Un dispositivo representa el equipo físico dentro de la planta. Este puede corresponder a un PLC, un nodo de computación en el borde (edge node) o incluso un servidor en la nube [18, 31]. Además de ejecutar tareas de control, estos dispositivos cuentan con interfaces que les permiten comunicarse con otros equipos y con el proceso industrial [4, 16].

Finalmente, cada dispositivo puede contener uno o varios Recursos. Estos funcionan como espacios de ejecución independientes donde se alojan partes de la aplicación y se gestionan los eventos que deben ejecutarse localmente [2, 18, 22].

#### ***2.4.4 Distribución del control***

Una de las principales ventajas del estándar IEC 61499 es que permite repartir las tareas de control entre varios dispositivos de la red [4, 6, 18]. En lugar de que un único controlador se encargue de todo el proceso, distintas partes de la aplicación pueden ejecutarse en diferentes equipos.

Para que estos dispositivos puedan intercambiar información, el estándar incorpora mecanismos de comunicación basados en Bloques de Interfaz de Servicio (SIFB), los cuales gestionan el intercambio de datos entre los distintos nodos de forma automática [6, 11, 18].

Gracias a esta distribución, el sistema es más flexible y resistente a fallos. Si un dispositivo presenta algún problema, no necesariamente se verá afectado todo el proceso [4, 6, 21]. Además, resulta más sencillo agregar nuevos módulos o realizar modificaciones locales, facilitando la adaptación del sistema a nuevas necesidades y conceptos como el Plug & Produce [17, 30].

### **3. Metodología**

Esta investigación se realizará mediante un enfoque documental y comparativo, haciendo una búsqueda, revisión y análisis de literatura especializada con respecto a los estándares IEC 61131-3 y IEC 61499, dentro de la cual se incluyen documentos técnicos oficiales, artículos científicos, publicaciones académicas y reportes de aplicaciones industriales.

A partir de esta revisión se contemplaron los principios conceptuales, las arquitecturas y las características operativas de ambos estándares con el fin de establecer una comparación clara y estructurada.

La estrategia metodológica utilizada consiste en un análisis comparativo cualitativo, en el cual se examinan ambos estándares con base en ciertos criterios, con el objetivo de identificar las



fortalezas y debilidades que puede presentar cada uno. Los criterios que se tomarán como base son los siguientes:

**Arquitectura:** se analiza la estructura que propone cada estándar para el desarrollo del sistema de control, considerando aspectos como el modelo de programación, la organización de los bloques de función, el paradigma de ejecución y la distribución de la lógica de control dentro del sistema.

**Escalabilidad:** se evalúa la adaptabilidad de los estándares a sistemas más complejos, así como la facilidad que presentan para ampliar funcionalidades, integrar nuevos dispositivos o expandir la arquitectura de control sin comprometer el programa base.

**Interoperabilidad:** en este criterio se analiza la capacidad de cada estándar para permitir la integración y comunicación entre diversos dispositivos, plataformas y sistemas desarrollados por distintos fabricantes, así como su compatibilidad con arquitecturas de automatización distribuidas.

**Flexibilidad:** se investiga la capacidad de cada estándar para ser modificado, reconfigurado y adaptar la lógica de control frente a cambios en los procesos o en los requerimientos del sistema.

**Implementación industrial:** se revisa el nivel de adopción de cada estándar en la industria, considerando su aplicación en entornos reales, plataformas comerciales, herramientas de desarrollo y sistemas automatizados.

Con base en lo anteriormente mencionado y en el análisis desarrollado, se buscará responder a la siguiente pregunta de investigación:

¿El estándar IEC 61499 representa un reemplazo, un complemento o una evolución del estándar IEC 61131-3 dentro de los sistemas modernos de automatización industrial?

A partir de esta interrogante, se analizarán las diferencias arquitectónicas, operativas y conceptuales entre ambos estándares, considerando aspectos como la distribución del control, la interoperabilidad, la modularidad y su nivel de adopción industrial. Los resultados permitirán identificar el papel actual de IEC 61499 frente a IEC 61131-3 y evaluar su pertinencia dentro de los entornos de automatización asociados a la Industria 4.0.

## **4. Análisis de Datos**

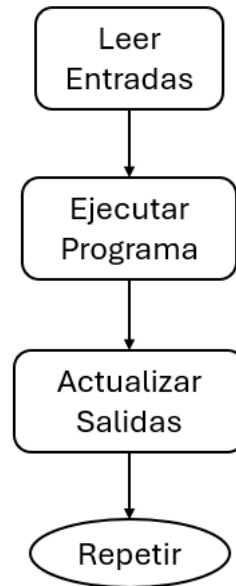
### **4.1 Modelo de ejecución**

Un modelo de ejecución es básicamente el núcleo operativo de todo sistema de automatización, pues determina la manera en que se planifican los recursos computacionales, se gestiona la concurrencia y se evalúan las restricciones de tiempo real en la planta. Realizar este análisis comparativo es esencial, ya que el comportamiento predecible, la escalabilidad y la capacidad de respuesta de un proceso industrial dependen directamente del paradigma bajo el cual el software interactúa con los sensores físicos, ejecuta los algoritmos y gestiona los actuadores [24, 30].

El modelo propuesto en la norma IEC 61131-3 se basa en una estructura de escaneo cíclico continuo (*scan cycle*). En este esquema centralizado el PLC opera mediante un bucle sin interrupción que se estructura en tres partes secuenciales: primero se adquiere y actualiza el estado de las señales de entrada, luego se ejecuta de forma secuencial la totalidad de la lógica del programa y, por último, se actualiza el estado de las variables de salida hacia los dispositivos de campo [3, 4]. La principal ventaja de este enfoque es la simplicidad arquitectónica que ofrece. Al trabajar con un único estado de los datos captados durante el desarrollo de cada ciclo de programa,

se puede garantizar una visión coherente del sistema, lo que proporciona una gran confiabilidad y tiempos de respuesta altamente predecibles en escenarios de control industrial convencionales [15, 18].

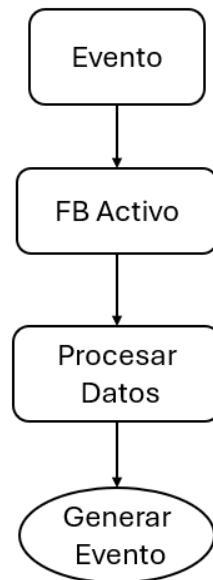
**Figura 1.** *Diagrama del ciclo de escaneo de IEC 61131-3*



Por otro lado, el estándar IEC 61499 propone un cambio hacia una ejecución orientada a eventos (*event-driven*), en donde el elemento fundamental es el bloque de función (FB). Este se estructura separando la interfaz de eventos (cabecera) de la interfaz de datos (cuerpo) [3, 32]. Bajo esta lógica, un FB permanece en estado de reposo junto con sus algoritmos internos hasta que se detecta un evento específico que lo active, momento en el cual se leen o actualizan únicamente los datos relacionados con dicho evento mediante restricciones temporales [1, 18]. De esa forma, distintas partes del sistema se pueden desplegar en diversos nodos, activándose únicamente cuando

se presenta un estímulo real, evitando la necesidad de un ciclo de escaneo continuo. Así se evita la repetición del procesamiento de entradas innecesarias y se brinda una mayor flexibilidad [3, 24].

**Figura 2.** Diagrama de activación por eventos en IEC 61499



Al analizar críticamente ambos enfoques, el impacto en la gestión de sistemas de tiempo real muestra diferencias importantes, el IEC 61131-3 ofrece un mejor control sobre los tiempos de ejecución cuando todo se realiza en un mismo controlador, pero presenta dificultades cuando el sistema se distribuye entre varios equipos, el IEC 61499 en cambio facilita la distribución del control y la comunicación entre dispositivos, pero puede resultar más complejo garantizar tiempos de respuesta más exactos debido a que trabaja mediante eventos. Por eso hay que analizar que es lo que se busca en el proceso a la hora de elegir uno u otro.

El modelo centralizado de IEC 61131-3 resulta más adecuado y eficiente para la automatización de lazos de control periódicos e intensivos en cálculos matemáticos directos, como los algoritmos de control PID, especialmente a nivel local [5, 15]. Por su parte, la ejecución por

eventos de IEC 61499 se presenta como una alternativa idónea para coordinar aplicaciones de control altamente secuenciales y para la organización de sistemas ciberfísicos (CPS) distribuidos, donde la interoperabilidad, la escalabilidad y la computación en el borde (*edge computing*) tienen mayor relevancia que el control centralizado estrictamente isócrono [4, 30].

## 4.2 Arquitectura del sistema

La arquitectura del sistema constituye uno de los elementos más determinantes en el diseño de los sistemas de automatización industrial, ya que define la forma en que los componentes de hardware y software se organizan e interactúan dentro de la planta productiva. En este sentido, no solo determina la estructura lógica del sistema, sino también la manera en que se distribuyen los recursos computacionales, la comunicación entre dispositivos y la gestión del flujo de información dentro del proceso industrial [11, 30]. En el contexto actual de la Industria 4.0 y los Sistemas Ciberfísicos (CPS), donde la interconexión entre equipos y el procesamiento de datos ocurre a través de múltiples nodos que abarcan desde el borde de la red (*edge*) hasta infraestructuras en la nube, una arquitectura rígida puede convertirse rápidamente en un obstáculo operativo. Por esta razón, disponer de un modelo arquitectónico adecuado resulta fundamental para garantizar la flexibilidad del sistema, la interoperabilidad entre dispositivos y la posibilidad de reconfigurar los procesos productivos sin afectar significativamente la operación de la planta [11, 21, 30].

El estándar IEC 61131-3 promueve tradicionalmente una arquitectura de carácter centralizado, en la cual la mayor parte de la lógica de control se ejecuta dentro de un único Controlador Lógico Programable (PLC) o bajo una estructura jerárquica claramente definida [4, 18]. Dentro de este modelo, la comunicación entre los distintos módulos del programa se basa principalmente en el uso de variables globales y rutas de acceso directas, las cuales funcionan

como el mecanismo principal para transferir información entre las diferentes secciones del sistema de control [9]. Si bien esta organización resulta relativamente sencilla de implementar y comprender, también introduce un alto grado de acoplamiento entre los módulos del programa. Desde una perspectiva arquitectónica, esto puede generar dificultades cuando se intenta escalar el sistema o distribuir la lógica de control entre varios dispositivos. En muchos casos, fragmentar el código de una máquina en diferentes controladores o expandir el sistema hacia nuevas capas de automatización requiere realizar modificaciones manuales extensas, además de depender de protocolos de comunicación específicos del fabricante. Como resultado, la capacidad del sistema para adaptarse a nuevas necesidades de producción puede verse limitada [4, 6, 15].

Por otro lado, el estándar IEC 61499 plantea una forma diferente de organizar los sistemas de automatización, enfocándose en la distribución del control y el modularidad. En lugar de concentrar toda la lógica en un único controlador, utiliza redes de bloques de función que pueden ejecutarse en distintos dispositivos de la planta [1, 6].

Una de las características más importantes de esta arquitectura es que separa la lógica del programa del hardware donde se ejecuta. Para lograrlo, el estándar organiza el sistema mediante los modelos de Aplicación, Dispositivo y Recurso [4, 11]. Gracias a esto, una aplicación puede diseñarse inicialmente como una única red lógica y luego distribuirse entre varios dispositivos según las necesidades del proceso [11, 23].

De esta manera, la comunicación entre los distintos equipos se realiza mediante interfaces de servicio, permitiendo que la lógica de control permanezca independiente del hardware utilizado. Esto facilita que diferentes partes del sistema puedan ejecutarse en PLC, dispositivos de computación en el borde o incluso servidores en la nube sin necesidad de modificar significativamente el programa [6, 11, 21].

Al comparar ambas arquitecturas se pueden observar diferencias importantes en aspectos como la escalabilidad, el mantenimiento y la flexibilidad. El modelo centralizado de IEC 61131-3 sigue siendo una opción muy utilizada en la industria debido a su simplicidad de implementación, facilidad de diagnóstico y confiabilidad en aplicaciones de control local. Además, cuenta con una amplia adopción industrial y una infraestructura tecnológica ya consolidada [5, 18].

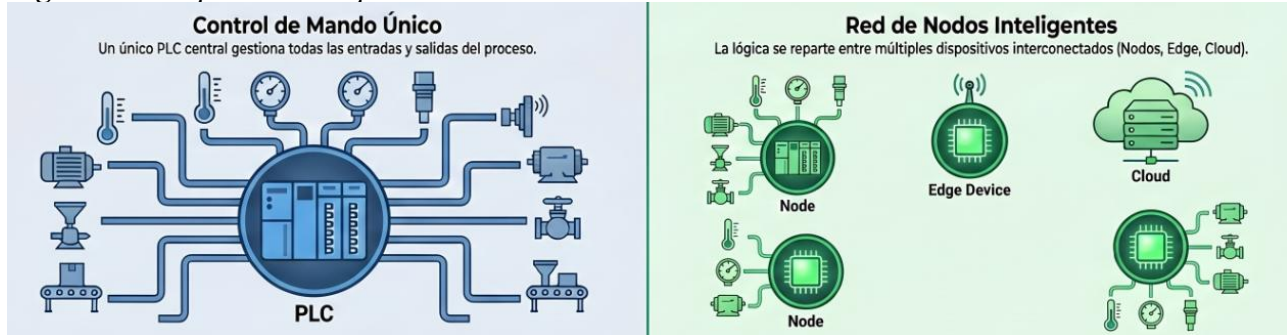
Sin embargo, cuando se necesitan ampliaciones o modificaciones importantes en el sistema, normalmente es necesario intervenir el controlador principal, lo que puede afectar temporalmente el funcionamiento del proceso [6].

Por el contrario, la arquitectura distribuida de IEC 61499 facilita la distribución del control entre varios dispositivos y permite realizar modificaciones en partes específicas del sistema sin detener toda la infraestructura de automatización. Esto le brinda una mayor flexibilidad para adaptarse a los requerimientos de los sistemas industriales modernos.

Asimismo, al promover una mayor independencia entre la lógica de control y el hardware utilizado, esta arquitectura reduce la dependencia tecnológica hacia un único proveedor y favorece la interoperabilidad entre distintos dispositivos y plataformas [4, 21, 30]. No obstante, su implementación práctica también presenta ciertos desafíos. La gestión de redes distribuidas introduce un mayor nivel de abstracción en el sistema, lo que puede dificultar la depuración de errores a escala global. Además, la adopción de este paradigma aún se ve limitada por la relativa escasez de profesionales con experiencia en arquitecturas de control distribuidas basadas en IEC 61499 [18, 30], mientras que la arquitectura centralizada de IEC 61131-3 continúa siendo una solución sólida y eficiente para sistemas de automatización tradicionales y fuertemente acoplados, el enfoque distribuido propuesto por IEC 61499 proporciona una base estructural más adecuada para los entornos de manufactura modernos, caracterizados por la interconexión de dispositivos,

la escalabilidad del sistema y la necesidad de adaptación continua de los procesos industriales [6, 30].

**Figura 3.** Comparación arquitectónica entre IEC 61131-3 e IEC 61499



### 4.3 Reutilización y modularidad

En la ingeniería de sistemas de automatización industrial, la reutilización de software y la modularidad representan principios clave para optimizar el desarrollo de soluciones de control. A medida que los procesos industriales demandan mayor flexibilidad y ciclos de vida de producto cada vez más cortos, resulta cada vez más importante construir sistemas a partir de componentes de software independientes que puedan reutilizarse, probarse y reconfigurarse en distintos contextos. Este enfoque no solo reduce los tiempos de desarrollo, sino que también facilita el mantenimiento del sistema y mejora su capacidad de evolución a largo plazo frente a nuevas necesidades de producción [3, 15, 30].

Dentro del estándar IEC 61131-3, la modularidad se implementa principalmente a través de las Unidades de Organización de Programas (POU), entre las cuales destacan las Funciones y los Bloques de Función (FB). Estos elementos permiten encapsular fragmentos de lógica que pueden reutilizarse dentro de un programa de PLC, e incluso organizarse en librerías de software que pueden ser utilizadas en distintos proyectos [3, 33]. Gracias a este mecanismo, los ingenieros



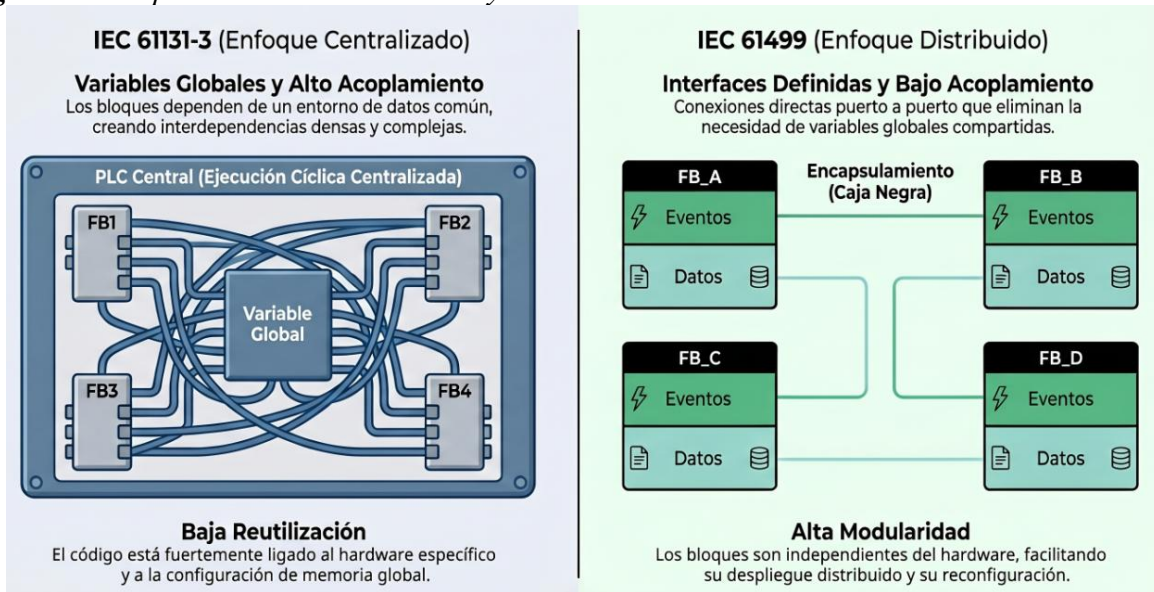
pueden instanciar múltiples veces una misma lógica de control sin necesidad de reescribir el código, lo que proporciona un nivel básico de reutilización dentro de un mismo sistema de automatización [15]. Sin embargo, en la práctica esta modularidad también presenta algunas limitaciones importantes. El uso de variables globales y rutas de acceso directas hace que muchos módulos del programa dependan unos de otros, por lo que reutilizar una parte del código en otro proyecto puede volverse una tarea complicada y normalmente requiere realizar modificaciones adicionales [9, 18]. Además, muchas de estas librerías terminan asociadas a las herramientas de desarrollo propias de cada fabricante de PLC, lo que dificulta mover el programa entre distintas plataformas y genera dependencia hacia un proveedor específico, situación conocida como vendor lock-in [3, 18].

Por otro lado, el estándar IEC 61499 fue desarrollado con una filosofía más orientada a la modularidad y a los sistemas distribuidos. En este modelo, la lógica de control se organiza mediante redes de bloques de función, donde cada bloque cumple una tarea determinada y se comunica con los demás mediante eventos y datos [1, 6]. Gracias a esto, cada bloque puede utilizarse de forma más independiente, facilitando su reutilización y permitiendo que diferentes partes del sistema se integren de manera más sencilla [17].

Además, el estándar incorpora herramientas que ayudan a organizar mejor la estructura del programa. Entre ellas se encuentran los Bloques de Función Compuestos (CFB), que permiten agrupar varios bloques dentro de uno solo, y los Adaptadores, los cuales simplifican la conexión entre diferentes partes del sistema [1, 13]. De esta manera, los componentes de software pueden reutilizarse con mayor facilidad incluso cuando se encuentran distribuidos en distintos dispositivos dentro de la red [15, 17, 30].

Al comparar ambos enfoques, se puede observar que el estándar IEC 61131-3 ofrece un nivel de reutilización adecuado para aplicaciones de control local donde la estructura del sistema no suele cambiar con frecuencia. No obstante, su fuerte acoplamiento entre módulos dificulta la evolución de arquitecturas más complejas o distribuidas [9, 15]. Por otro lado, el modelo propuesto por IEC 61499 facilita en mayor medida el desarrollo de sistemas verdaderamente modulares al promover un bajo acoplamiento entre componentes y una alta cohesión interna. Esta característica reduce considerablemente el esfuerzo de ingeniería necesario para modificar, ampliar o mantener aplicaciones de automatización que se ejecutan en múltiples dispositivos [9, 13]. Mientras que las librerías basadas en IEC 61131-3 suelen requerir ajustes manuales cuando se migran entre plataformas, los componentes desarrollados bajo IEC 61499 pueden reutilizarse de forma más transparente en distintos entornos de ejecución [18, 23].

En última instancia, esta diferencia refleja una evolución más amplia dentro de la ingeniería de software industrial hacia modelos de desarrollo basados en componentes. Mientras que el estándar IEC 61131-3 presenta ciertas limitaciones para implementar de forma estricta los principios de la ingeniería de software basada en componentes (Component-Based Software Engineering, CBSE), el estándar IEC 61499 adopta de manera más natural estos principios al permitir la construcción de sistemas a partir de módulos altamente encapsulados y reutilizables. De esta forma, proporciona una base más adecuada para el desarrollo de aplicaciones industriales escalables, interoperables y capaces de integrarse dentro de arquitecturas modernas orientadas a servicios (SOA) [11, 29, 30].

**Figura 4.** Comparación de modularidad y reutilización en IEC 61131-3 e IEC 61499

#### 4.4 Interoperabilidad

En los sistemas de automatización industrial modernos, especialmente bajo el paradigma de la Industria 4.0, la interoperabilidad se ha convertido en un requisito fundamental dentro del diseño arquitectónico de las plantas productivas. En términos generales, la interoperabilidad hace referencia a la capacidad de distintos dispositivos, plataformas de control y herramientas de ingeniería de compartir información y coordinar sus operaciones de forma transparente, incluso cuando provienen de fabricantes diferentes [11, 17]. Este aspecto resulta especialmente relevante en entornos industriales actuales, donde los sistemas de producción suelen estar conformados por múltiples capas tecnológicas que abarcan desde sensores y actuadores hasta sistemas de supervisión y plataformas de análisis de datos. Cuando esta integración no se logra adecuadamente, es común que la información quede fragmentada en lo que se conoce como *data silos*, lo que dificulta el análisis global del proceso y limita la capacidad de tomar decisiones en tiempo real [20]. A medida que las plantas industriales se vuelven más interconectadas y heterogéneas, la falta

de interoperabilidad no solo incrementa los costos de integración, sino que también restringe la escalabilidad del sistema y complica la adopción de arquitecturas distribuidas más avanzadas [6, 20].

El estándar IEC 61131-3 representó en su momento un paso importante hacia la estandarización del desarrollo de software para Controladores Lógicos Programables (PLC), ya que definió un conjunto común de lenguajes de programación ampliamente utilizados dentro de la automatización industrial [18, 33]. Gracias a esta normalización, fue posible reducir parcialmente la fragmentación del mercado y facilitar el aprendizaje de los lenguajes de programación de PLC por parte de los ingenieros. Sin embargo, a pesar de compartir una sintaxis relativamente común, los problemas de interoperabilidad entre plataformas de distintos fabricantes continúan siendo significativos [3]. Esto ocurre principalmente porque el estándar no define de manera estricta cómo deben implementarse las interfaces entre el software de control y el hardware subyacente, ni cómo deben integrarse las redes industriales dentro de los entornos de programación. Como consecuencia, muchos fabricantes han desarrollado entornos de desarrollo integrados (IDE) con características propietarias que dificultan la migración directa de programas entre distintas plataformas [3, 17]. En la práctica, esto significa que un programa desarrollado para un PLC de un proveedor específico rara vez puede trasladarse a otro dispositivo de una marca diferente sin requerir modificaciones sustanciales en el código o en la configuración de la comunicación. Esta situación ha contribuido a consolidar un fenómeno conocido en la industria como *vendor lock-in*, donde los usuarios terminan dependiendo tecnológicamente de un único proveedor [17, 30].

Para intentar solucionar estas limitaciones, el estándar IEC 61499 propone un modelo basado en aplicaciones distribuidas que busca mejorar la portabilidad del software y facilitar la

interoperabilidad entre dispositivos de diferentes fabricantes. Para ello, utiliza redes de bloques de función que pueden distribuirse entre distintos nodos dentro de un sistema de control [6, 23].

Una de las características más importantes de esta arquitectura es que separa la lógica de la aplicación del hardware donde finalmente será ejecutada [4, 11]. Gracias a esto, los desarrolladores pueden diseñar el programa sin depender directamente de un equipo específico.

Además, para gestionar la comunicación con sensores, actuadores y redes industriales, el estándar utiliza los Bloques de Interfaz de Servicio (SIFB). Estos bloques se encargan de manejar las funciones relacionadas con el dispositivo físico, permitiendo que la lógica principal de la aplicación permanezca independiente del entorno donde se ejecuta [16, 24].

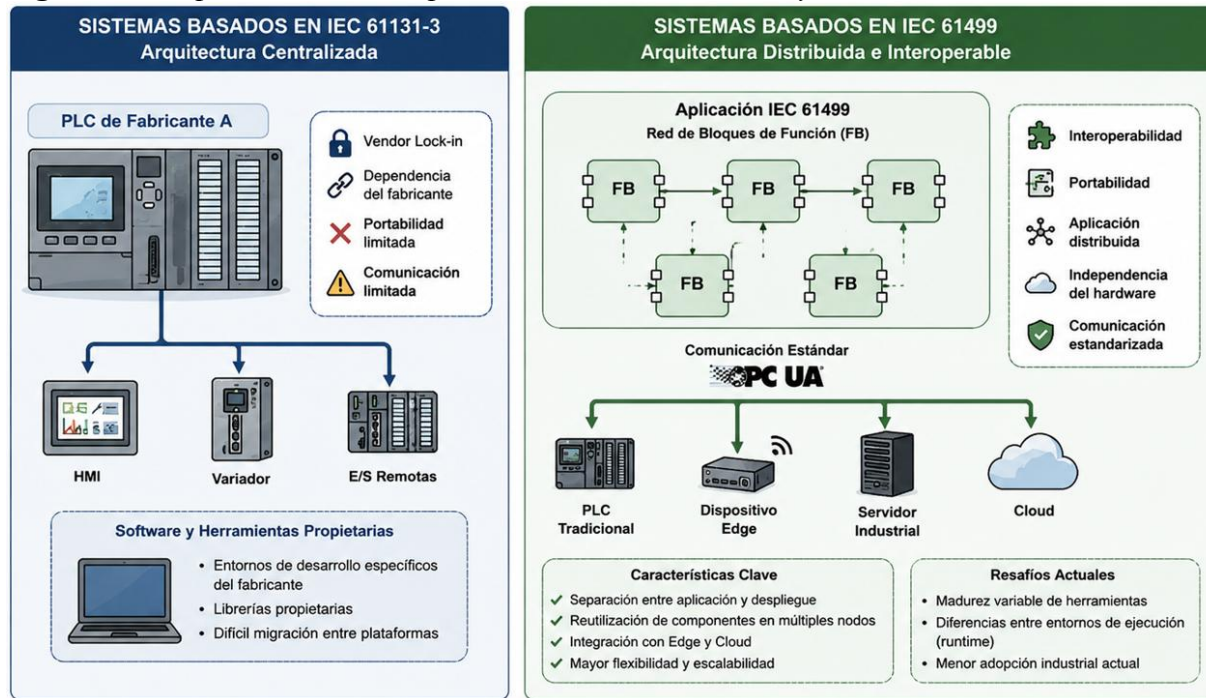
Por otro lado, el estándar también promueve el uso de modelos de comunicación estandarizados como Publicador/Suscriptor, donde el entorno de ejecución es el encargado de coordinar el intercambio de información entre los diferentes dispositivos del sistema [6, 24]. Con esto se busca reducir la dependencia de plataformas propietarias y facilitar el desarrollo de aplicaciones que puedan ejecutarse en distintos equipos utilizando una misma estructura de diseño.

Sin embargo, al comparar ambos estándares en situaciones reales de la industria, se puede observar que el problema de la interoperabilidad no desaparece por completo con la llegada de IEC 61499. Aunque este estándar propone una arquitectura más flexible y abierta, todavía existen algunas limitaciones relacionadas con las herramientas disponibles y el nivel de adopción que tienen actualmente en la industria [6, 30].

Diferentes entornos de desarrollo compatibles con IEC 61499, como FBDK, 4diac o nxtSTUDIO, implementan el estándar con variaciones propias en la forma en que gestionan los modelos XML y las representaciones gráficas de ejecución. Estas diferencias pueden ocasionar ciertas incompatibilidades que dificultan la migración directa de aplicaciones entre distintas

plataformas, por lo que en algunos casos es necesario realizar ajustes manuales [23, 30]. Además, aunque IEC 61499 fue concebido para facilitar la interoperabilidad, en muchos entornos industriales todavía se requiere el uso de tecnologías complementarias para garantizar una comunicación efectiva entre dispositivos de diferentes fabricantes. Entre las soluciones más utilizadas se encuentra OPC UA, que proporciona un mecanismo estandarizado para el intercambio de información dentro de las redes industriales [6, 20, 22].

**Figura 5.** Comparativa de Interoperabilidad de IEC 61131-3 y 61499



En conclusión, los resultados obtenidos indican que IEC 61499 ofrece una arquitectura más adecuada para favorecer la interoperabilidad en sistemas de automatización distribuidos. No obstante, las ventajas propuestas por el estándar aún se encuentran condicionadas por factores como la madurez de las herramientas de desarrollo disponibles y el grado de adopción alcanzado

dentro del sector industrial. Por su parte, IEC 61131-3 continúa siendo ampliamente utilizado, aunque generalmente dentro de entornos más dependientes de soluciones específicas de cada fabricante.

Aunque IEC 61499 propone una arquitectura más abierta para integrar dispositivos de diferentes fabricantes, alcanzar una interoperabilidad completa sigue siendo un desafío. Esto se debe a que no depende únicamente del estándar, sino también de la compatibilidad entre las herramientas utilizadas y del nivel de adopción que tenga por parte de la industria [6, 23, 30].

#### **4.5 Implementación industrial**

Si se habla sobre el nivel de adopción de un estándar en los entornos de manufactura, donde la viabilidad práctica y su alineación con las necesidades reales de la industria son factores determinantes, IEC 61131-3 continúa siendo el predominante a nivel global, ofreciendo un entorno de desarrollo caracterizado por su madurez técnica, estabilidad operativa y comportamiento altamente predecible [3, 18]. Dicho predominio se explica, en gran medida, por el amplio respaldo que ha recibido por parte de los principales fabricantes de hardware y software industrial, quienes durante décadas han desarrollado herramientas, bibliotecas y entornos de programación basados en este estándar [18, 30].

Como resultado, IEC 61131-3 forma actualmente la base de una enorme infraestructura instalada en plantas de manufactura y procesos industriales alrededor del mundo.

Por otro lado, el estándar IEC 61499 fue concebido con el objetivo de superar algunas de las limitaciones estructurales de los sistemas de control centralizados y responder a las nuevas necesidades de flexibilidad, distribución y escalabilidad propias de la manufactura moderna. Su arquitectura orientada a eventos y basada en redes de bloques de función propone un enfoque más

adecuado para la coordinación de sistemas de automatización distribuidos y para la integración de dispositivos dentro de entornos industriales interconectados [2, 16]. Sin embargo, a pesar de sus ventajas conceptuales y de las mejoras que introduce a nivel arquitectónico, su adopción dentro de aplicaciones industriales reales sigue siendo relativamente limitada. Diversos estudios reportan que la mayoría de las implementaciones de IEC 61499 se encuentran todavía en proyectos de investigación, entornos académicos o aplicaciones piloto, sin haber logrado aún una penetración significativa en el mercado industrial masivo [3, 18, 30].

Este panorama da lugar a una duda bastante grande sobre IEC 61499: si es un modelo que ofrece diversas mejoras arquitectónicas frente a su antecesor, IEC 61131-3, ¿por qué su adopción en la industria sigue siendo tan limitada? Para entender las razones detrás de este fenómeno, es necesario comprender los factores que influyen directamente en la toma de decisiones dentro de las empresas industriales.

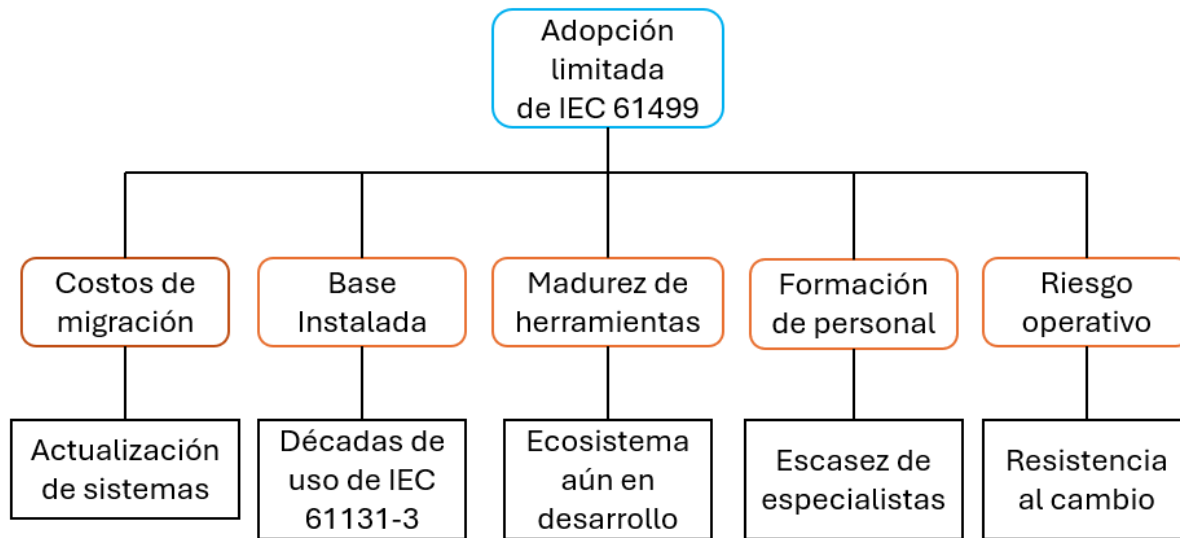
En primer lugar, se encuentra la gran inercia y base instalada de IEC 61131-3. Este sistema ha sido utilizado durante muchos años por la gran mayoría de compañías y gran parte de su arquitectura se fundamenta en dicho estándar. Aunque presenta algunas limitaciones, la confianza y familiaridad que genera tienen un peso bastante grande para los ingenieros y técnicos del área [3,18,30]. Además, reemplazar toda la arquitectura por un sistema completamente diferente al acostumbrado solo generaría más gastos, tanto por la adquisición de equipos como por la capacitación del personal, ya que son pocos los profesionales que actualmente están capacitados para el manejo correcto del estándar IEC 61499 debido a su limitada implantación en la industria.

También se han observado en investigaciones recientes algunas diferencias en la forma en que distintas plataformas implementan el estándar, particularmente en lo relacionado con la ejecución de eventos y la interpretación de ciertos elementos del modelo [6, 16, 23]. Estas



variaciones pueden generar inconsistencias entre herramientas de distintos proveedores, lo que reduce la portabilidad real de las aplicaciones y puede obligar a realizar ajustes manuales al migrar un sistema entre plataformas [19].

**Figura 6.** *Limitantes de adopción de IEC 61499*



El panorama actual de la automatización industrial muestra una clara coexistencia entre ambos estándares. Mientras que IEC 61131-3 continúa siendo la solución predominante dentro de la industria gracias a su madurez tecnológica, fiabilidad comprobada y amplio ecosistema de herramientas, IEC 61499 representa una propuesta arquitectónica orientada a los sistemas de automatización distribuidos que probablemente ganará mayor relevancia a medida que evolucionen las necesidades de la manufactura digital. No obstante, para que pueda ocurrir una transición, es necesario desarrollar estrategias de implementación, mejorar la estabilidad y la estandarización de las herramientas disponibles y, a su vez, intentar integrar estas nuevas herramientas sin comprometer la estructura base de los procesos productivos ya establecidos.

## 5. Discusión

El análisis comparativo entre ambos estándares mostró que la implementación de la Industria 4.0 no depende exclusivamente de los avances de hardware o de la capacidad de procesamiento, sino también de la estructura de software desarrollada en los sistemas de control. Durante años, la arquitectura centralizada basada en ciclos de escaneo de IEC 61131-3 ha sido bastante robusta y muy confiable, pero, a su vez, a medida que aparecen nuevos retos y aplicaciones que requieren un enfoque distribuido, esta no siempre puede satisfacer dichas necesidades.

El estándar IEC 61499 propone una arquitectura de bloques modulares bastante interesante, con un ecosistema de desarrollo altamente flexible basado en la portabilidad del software, la interoperabilidad entre dispositivos de distintos fabricantes y una mayor independencia del hardware. Sin embargo, al comparar las ventajas teóricas con la realidad de la industria, se observa que, en la práctica, su uso es bastante más limitado de lo que se propuso en un inicio. Diferentes herramientas de desarrollo interpretan el estándar de manera ligeramente distinta, lo que genera inconsistencias entre plataformas. Esto puede observarse en entornos como FBDK, 4diac o nxtSTUDIO, donde ciertas diferencias en la lógica de ejecución o en el manejo de los modelos XML dificultan la transferencia directa de aplicaciones entre sistemas.

Como resultado, la posibilidad de trasladar una aplicación de control entre distintas plataformas sin realizar modificaciones importantes todavía no se cumple de manera consistente en entornos industriales reales.

A estas dificultades se suman algunas limitaciones propias del estándar IEC 61499. Su arquitectura distribuida ofrece una mayor flexibilidad, pero también introduce un nivel de complejidad superior al del modelo tradicional basado en ciclos de escaneo. En este tipo de

sistemas pueden ocurrir varios eventos al mismo tiempo en diferentes dispositivos, lo que dificulta mantener un comportamiento completamente predecible y requiere mecanismos adicionales de sincronización.

Por otra parte, la eliminación de las variables globales como medio de comunicación entre bloques ayuda a reducir la dependencia entre módulos. Sin embargo, también obliga a establecer una mayor cantidad de conexiones entre ellos, haciendo que los diagramas de control sean más complejos y que tareas como la depuración, el mantenimiento o la detección de fallos resulten más difíciles.

Con lo anterior mencionado, migrar desde IEC 61131-3 resulta poco práctico, pues este, a pesar de todo, ha demostrado ser un estándar de gran confiabilidad y madurez en el mercado. Además, aplicar cambios mediante otro modelo solo generaría riesgos organizacionales que podrían traer más problemas que beneficios. La enorme inversión acumulada por parte de las entidades, sumada a que la gran mayoría de los profesionales ya están adaptados a este modelo, hace que la idea de migrar hacia IEC 61499 sea bastante compleja tanto a nivel de lógica como de tiempo e inversión.

El IEC 61499 puede verse como una evolución del modelo de ciclos si se analiza desde el ámbito de las propuestas que plantea. Su arquitectura orientada a eventos, su capacidad modular y su enfoque hacia la integración de nuevas tecnologías, como la computación en la nube y el procesamiento distribuido, son algunas de sus características más prometedoras, ya que se ajustan a muchas de las necesidades que busca la industria en la actualidad.

Sin embargo, al analizar su adopción real hoy en día, se observa que aún es un estándar que requiere mayor desarrollo para ser óptimo y aplicarse a gran escala. La falta de herramientas

comerciales estandarizadas y la dificultad para traducir algunos enfoques tradicionales de programación limitan su adopción como sustituto de IEC 61131-3.

En resumen, aunque IEC 61499 ofrece varias ventajas frente a IEC 61131-3, su adopción en la industria sigue siendo limitada. Muchas empresas no consideran que los beneficios actuales justifiquen el costo y el esfuerzo que implica una migración completa. Además, el desarrollo de sistemas distribuidos todavía presenta desafíos técnicos y las herramientas disponibles continúan evolucionando. A esto se suma que gran parte del mercado sigue dependiendo de soluciones y plataformas asociadas a fabricantes específicos, lo que dificulta una transición más amplia hacia este estándar. Mientras estas condiciones no cambien, es probable que IEC 61499 continúe utilizándose principalmente como un marco conceptual para arquitecturas distribuidas avanzadas, coexistiendo con la solidez y la amplia adopción del IEC 61131-3 en los niveles más tradicionales del control industrial. En consecuencia, los resultados obtenidos sugieren que IEC 61499 difícilmente puede considerarse en la actualidad un reemplazo directo de IEC 61131-3. Su papel parece orientarse principalmente hacia la ampliación de las capacidades de los sistemas de automatización mediante arquitecturas distribuidas, lo que permite interpretarlo simultáneamente como un complemento práctico y como una evolución conceptual del paradigma tradicional de control industrial.

**Tabla 1.** Comparación entre ambos estándares

| Aspecto                    | IEC 61131-3                                            | IEC 61499                                                        |
|----------------------------|--------------------------------------------------------|------------------------------------------------------------------|
| <i>Arquitectura</i>        | Control centralizado basado en ciclos de escaneo.      | Control distribuido basado en eventos.                           |
| <i>Madurez tecnológica</i> | Muy alta; ampliamente implementado en la industria.    | Menor madurez y adopción industrial limitada.                    |
| <i>Interoperabilidad</i>   | Dependiente de herramientas y fabricantes específicos. | Diseñado para favorecer la interoperabilidad entre dispositivos. |

|                                   |                                                                          |                                                                                       |
|-----------------------------------|--------------------------------------------------------------------------|---------------------------------------------------------------------------------------|
| <i>Portabilidad</i>               | Generalmente limitada entre plataformas.                                 | Busca facilitar la reutilización y migración de aplicaciones.                         |
| <i>Determinismo</i>               | Alto nivel de predictibilidad temporal.                                  | Puede requerir mecanismos adicionales para garantizar sincronización.                 |
| <i>Modularidad</i>                | Adecuada, aunque con menor flexibilidad para sistemas distribuidos.      | Elevada modularidad mediante bloques de función y ECC.                                |
| <i>Escalabilidad</i>              | Eficiente para sistemas centralizados.                                   | Mejor adaptación a sistemas distribuidos complejos.                                   |
| <i>Curva de aprendizaje</i>       | Amplia disponibilidad de personal capacitado.                            | Requiere aprendizaje de nuevos paradigmas de diseño.                                  |
| <i>Herramientas de desarrollo</i> | Ecosistema consolidado y ampliamente soportado.                          | Menor cantidad de herramientas y diferencias entre implementaciones.                  |
| <i>Industria 4.0</i>              | Puede integrarse, pero no fue diseñado específicamente para ello.        | Alineado con conceptos de sistemas ciberfísicos e Industria 4.0.                      |
| <i>Rol actual en la industria</i> | Estándar predominante para aplicaciones de automatización tradicionales. | Complemento y evolución conceptual orientada a sistemas distribuidos e Industria 4.0. |

## 6. Conclusiones

La investigación y el análisis comparativo desarrollado en esta monografía permiten concluir que IEC 61499 no constituye actualmente un reemplazo directo del estándar IEC 61131-3 dentro de los sistemas modernos de automatización industrial. Por el contrario, la evidencia analizada indica que su papel se aproxima más al de un complemento tecnológico y una evolución conceptual de los paradigmas tradicionales de control. El estándar IEC 61499 fue concebido para extender las bases tecnológicas consolidadas de IEC 61131-3, incorporando lenguajes de programación ampliamente probados, como el Texto Estructurado, dentro de un modelo de ejecución orientado a eventos.

Este estándar introdujo una forma distinta de desarrollar sistemas de automatización mediante la incorporación de una arquitectura orientada a eventos y sistemas distribuidos, permitiendo una mayor independencia de la lógica de control con respecto al hardware y

facilitando además la integración de tecnologías asociadas a la Industria 4.0 y los Sistemas Ciberfísicos. Aun así, su limitada implementación y nivel de desarrollo dentro del mercado hacen que la industria continúe optando por IEC 61131-3 debido a su amplio historial de uso y a su gran adopción como arquitectura de referencia.

En conclusión, IEC 61499 ofrece características que responden a varias de las necesidades actuales de la automatización industrial. Sin embargo, su adopción a gran escala todavía depende de una mayor madurez de las herramientas disponibles y de una integración más sólida con la infraestructura existente. Mientras esto ocurre, IEC 61131-3 continuará siendo un estándar ampliamente utilizado dentro de la industria debido a su confiabilidad, estabilidad y alto nivel de adopción.

### Referencias

- [1] M. Grave, M. Meingast, V. Klos, A. Zoitl, L. Sonnleithner, and J. Walter, “A Formal Model Transformation from Data Flow Models to IEC 61499 Models,” in 2025 IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Porto, Portugal, Oct. 2025. doi: <https://ieeexplore.ieee.org/document/11205747>.
- [2] P. Gsellmann, M. Melik-Merkumians, A. Zoitl, and G. Schitter, “A Novel Approach for Integrating IEC 61131-3 Engineering and Execution into IEC 61499,” *IEEE Trans. Industr. Inform.*, vol. 17, no. 8, pp. 5411–5418, Aug. 2021, doi: <https://ieeexplore.ieee.org/document/9238463>.
- [3] J. Aalto, “Benefits of transferring from IEC 61131-3 to IEC 61499,” Bachelor's thesis, Aalto University, Espoo, Finland, 2025. doi: <https://aaltodoc.aalto.fi/items/7a110286-bbc6-418c-94b0-e2a8f56ed4e2>.
- [4] E. Monroy Cruz, L. R. García Carrillo, L. A. Cruz Salazar and B. D. Hernández Rojo, “Comparación de estándares IEC 61131-3 e IEC 61499 para implementar sistemas de control distribuido,” in *XXVIII Congreso Internacional Anual de la SOMIM – Memorias de Divulgación Científica, Tecnológica e Innovación*, Bogotá, Colombia, Sep. 2022. doi: [https://www.researchgate.net/publication/367251037\\_Comparacion\\_de\\_estandares\\_IEC\\_61131-3\\_e\\_IEC\\_61499\\_para\\_implementar\\_sistemas\\_de\\_control\\_distribuido](https://www.researchgate.net/publication/367251037_Comparacion_de_estandares_IEC_61131-3_e_IEC_61499_para_implementar_sistemas_de_control_distribuido).
- [5] P. Gsellmann, M. Melik-Merkumians, and G. Schitter, “Comparison of Code Measures of IEC 61131-3 and 61499 Standards for Typical Automation Applications,” in *2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA)*, Turin, Italy, Sep. 2018, pp. 1047–1050. doi: <https://ieeexplore.ieee.org/document/8502464>.

- [6] P. Padmathilaka, “Comparison, co-existence and interoperability of heterogeneous IEC 61499 control systems,” Master's thesis, Aalto University, Espoo, Finland, 2025. doi: <https://aaltodoc.aalto.fi/items/0c1c2594-40a2-4b94-900b-d1f31fc959e2>.
- [7] T. Lyu, U. D. Atmojo, and V. Vyatkin, “Complexity-Aware Distribution Optimization for Computing-intensive Distributed Automation Systems with IEC 61499,” in *2025 IEEE 23rd International Conference on Industrial Informatics (INDIN)*, Kunming, China, Jul. 2025, pp. 1–6. doi: <https://ieeexplore.ieee.org/document/11279613>.
- [8] L. Sonnleithner, B. Wiesmayr, A. M. Gutiérrez, R. Rabiser, and A. Zoitl, “Complexity of Structured Text in IEC 61499 Function Blocks: A Survey,” in *2023 IEEE 28th International Conference on Emerging Technologies and Factory Automation (ETFA)*, Sinaia, Romania, Sep. 2023. doi: <https://ieeexplore.ieee.org/document/10275434>.
- [9] A. Homaý, M. De Sousa, A. Zoitl, and M. Wollschlaeger, “Coupling and Decoupling in IEC 61499 and IEC 61131-3 Applications,” in *2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA)*, Stuttgart, Germany, Sep. 2022. doi: <https://ieeexplore.ieee.org/document/9921472>.
- [10] M. Xavier, S. Patil, and V. Vyatkin, “Cyber-Physical Automation Systems Modelling with IEC 61499 for their Formal Verification,” in *2021 IEEE 19th International Conference on Industrial Informatics (INDIN)*, Palma de Mallorca, Spain, Jul. 2021. doi: <https://ieeexplore.ieee.org/document/9557416>.
- [11] W. Dai, Y. Zhang, L. Kong, J. H. Christensen, and D. Huang, “Design of Industrial Edge Applications Based on IEC 61499 Microservices and Containers,” *IEEE Trans. Industr. Inform.*, vol. 19, no. 7, pp. 7925–7935, Jul. 2023, doi: <https://ieeexplore.ieee.org/document/9918040>.



- [12] J. Cabral, M. Wenger, and A. Zoitl, “Enable Co-Simulation for Industrial Automation by an FMU Exporter for IEC 61499 Models,” in *2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA)*, Turin, Italy, Sep. 2018. doi: <https://ieeexplore.ieee.org/document/8502654>.
- [13] M. Oberlehner, A. Eigner, M. Wimmer, and A. Zoitl, “Exploring Refactoring Operations for IEC 61499,” in *2023 IEEE 28th International Conference on Emerging Technologies and Factory Automation (ETFA)*, Sinaia, Romania, Sep. 2023. doi: <https://ieeexplore.ieee.org/document/10275635>.
- [14] P. Ovsianikova, T. Liakh, P. Jhunhunwala, and V. Vyatkin, “Generative AI Co-Pilot for Rapid Prototyping of IEC 61499 Control Applications,” in *2024 IEEE 29th International Conference on Emerging Technologies and Factory Automation (ETFA)*, Padova, Italy, Sep. 2024. doi: <https://ieeexplore.ieee.org/document/10711156>.
- [15] M. M. Merkmians, P. Gsellmann, and G. Schitter, “Hierarchization and Integration of IEC 61131-3 and IEC 61499 for Enhanced Reusability,” in *2021 IEEE 26th International Conference on Emerging Technologies and Factory Automation (ETFA)*, Vasteras, Sweden, Sep. 2021. doi: <https://ieeexplore.ieee.org/document/9613226>.
- [16] I. A. Batchkova, Y. A. Belev, and D. L. Tzakova, “IEC 61499 Based Control of Cyber-Physical Systems,” *International Scientific Journal Industry 4.0*, vol. 5, no. 1, pp. 10–13, 2020. doi: <https://stumejournals.com/journals/i4/2020/1/10.full.pdf>
- [17] J. Conway, “IEC 61499: La norma de automatización industrial para la portabilidad que permite aprovechar las ventajas de la Industria 4.0,” White Paper, Schneider Electric, Aug. 2021. doi: [https://www.se.com/es/es/download/document/998-21041914\\_GMA\\_WhitePaper\\_ES/](https://www.se.com/es/es/download/document/998-21041914_GMA_WhitePaper_ES/).

- [18] V. Ashiwal, O. Miguel-Escrig, B. Wiesmayr, A. Zoitl, and J. A. Romero-Perez, “Identification and Evaluation of Pitfalls in the Migration From IEC 61131-3 to IEC 61499: A Review,” *IEEE Open Journal of the Industrial Electronics Society*, vol. 6, pp. 575–590, 2025. doi: <https://ieeexplore.ieee.org/document/10955217>.
- [19] M. Foletti, L. Agbomemewa, P. Pedrazzoli, and M. Confalonieri, “Implementation and System Architecture Challenges for Event-Based Programmable Logic Controllers - A Literature Review,” *Procedia CIRP*, vol. 130, pp. 1529–1536, 2024. doi: <https://www.sciencedirect.com/science/article/pii/S2212827124014355>.
- [20] V. Tavakkoli, K. Mohsenzadegan, W. V. Kambale, and K. Kyamakya, “Integrating IEEE 1451 and IEC 61499 Standards for Enhanced Interoperability in Smart Manufacturing and Transportation Systems,” in *2024 19th International Workshop on Semantic and Social Media Adaptation and Personalization (SMAP)*, Athens, Greece, Nov. 2024, pp. 14–18. doi: <https://ieeexplore.ieee.org/document/10858972>.
- [21] G. Lyu, A. Fazlirad, and R. W. Brennan, “Multi-agent modeling of cyber-physical systems for IEC 61499 based distributed automation,” *Procedia Manufacturing*, vol. 51, pp. 1200–1206, 2020. doi: <https://www.sciencedirect.com/science/article/pii/S2351978920320254>.
- [22] I. Seilonen, V. Vyatkin, and U. D. Atmojo, “OPC UA Information Model and a Wrapper for IEC 61499 Runtimes,” in *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, Helsinki and Espoo, Finland, Jul. 2019, pp. 1008–1013. doi: <https://ieeexplore.ieee.org/document/8971964>.
- [23] A. Hopsu, U. D. Atmojo, and V. Vyatkin, “On Portability of IEC 61499 Compliant Structures and Systems,” in *2019 IEEE 28th International Symposium on Industrial*

- Electronics (ISIE)*, Vancouver, Canada, Jun. 2019, pp. 1306–1311. doi: <https://ieeexplore.ieee.org/document/8781290>.
- [24] S. Gao, K. Qin, X. Zhang, and W. Dai, “Prioritized Deterministic Real-Time Execution Semantics for Industrial Edge Applications Based on IEC 61499 Event Types,” *IEEE Trans. Industr. Inform.*, vol. 21, no. 11, pp. 8550–8561, Nov. 2025, doi: <https://ieeexplore.ieee.org/document/11125811>.
- [25] P. Padmathilaka, P. Jhunjhunwala, and V. Vyatkin, “Process Simulation Development with IEC 61499 for Runtime Execution,” in *2025 IEEE 30th International Conference on Emerging Technologies and Factory Automation (ETFA)*, Porto, Portugal, Sep. 2025. doi: <https://ieeexplore.ieee.org/document/11205643>.
- [26] A. Tanveer, R. Sinha, and M. M. Y. Kuo, “Secure Links: Secure-by-Design Communications in IEC 61499 Industrial Control Applications,” *IEEE Trans. Ind. Informatics*, vol. 17, no. 6, pp. 3992–4002, Jun. 2021. doi: <https://ieeexplore.ieee.org/document/9140423>.
- [27] A. A. Santos and A. F. da Silva, “Simulation and Control of a Cyber-Physical System under IEC 61499 Standard,” in *30th International Conference on Flexible Automation and Intelligent Manufacturing (FAIM 2021)*, Athens, Greece, Sep. 2021, pp. 72–79. doi: <https://www.sciencedirect.com/science/article/pii/S2351978921002080>.
- [28] V. Bashev, I. S. Anureev, and V. E. Zyubin, “The Post Language: Process-Oriented Extension for IEC 61131-3 Structured Text,” in *2020 International Russian Automation Conference (RusAutoCon)*, Sochi, Russia, Sep. 2020. doi: <https://ieeexplore.ieee.org/document/9208049>.

- [29] G. Lyu and R. W. Brennan, “Towards IEC 61499 Based Distributed Intelligent Automation: Design and Computing Perspectives,” in *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, Helsinki, Finland, Jul. 2019, pp. 160–163. doi: <https://ieeexplore.ieee.org/document/8972153>.
- [30] G. Lyu and R. W. Brennan, “Towards IEC 61499-Based Distributed Intelligent Automation: A Literature Review,” *IEEE Trans. Ind. Informatics*, vol. 17, no. 4, pp. 2295–2306, Apr. 2021, doi: <https://ieeexplore.ieee.org/document/9169795>.
- [31] T. Lyu, T. Nguyen, R. Rumiantsev, M. Kolesnikov, H. Perera, U. D. Atmojo, and V. Vyatkin, “Towards Web Platform for Cloud-Based Virtual Commissioning of IEC 61499 Distributed Automation Systems,” in *2024 IEEE 29th International Conference on Emerging Technologies and Factory Automation (ETFA)*, Padova, Italy, Sep. 2024. doi: <https://ieeexplore.ieee.org/document/10710351>.
- [32] H. Kothalawala, P. Jhunjhunwala, and V. Vyatkin, “Towards Migration from IEC 61131-3 to IEC 61499 in Process Industry: Redesign of Visualisation,” in *2023 IEEE 32nd International Symposium on Industrial Electronics (ISIE)*, Helsinki, Finland, Jun. 2023, doi: <https://ieeexplore.ieee.org/document/10227953>.
- [33] V. E. Zyubin, A. S. Rozov, I. S. Anureev, N. O. Garanina, and V. Vyatkin, “PoST: A Process-Oriented Extension of the IEC 61131-3 Structured Text Language,” *IEEE Access*, vol. 10, pp. 35238–35250, 2022, doi: <https://ieeexplore.ieee.org/document/9729833>.

