IEC 61499 and the Promise of Holonic Systems

Robert W. Brennan(✉) and Guolin Lyu

Schulich School of Engineering, University of Calgary, Calgary, Canada
rbrennan@ucalgary.ca

Abstract. This paper provides a review of the IEC 61499 standard for industrial automation from the perspective of holonic manufacturing systems. This standard played a central role since the beginnings of the Holonic Manufacturing Systems (HMS) movement, and by many, was considered the enabling approach to realize holonic systems. In this paper, we ask if, after 20 years of research and development, this promise has been realized.

Keywords: IEC 61499 standard · Holonic manufacturing systems

1 Introduction

The concept of holonic systems has its roots in the goal of developing a distributed intelligent control paradigm for manufacturing. Unlike multi-agent systems (MAS), which is a broader software approach that can be also used for distributed intelligent control, a holonic manufacturing system (HMS) is, by definition, a manufacturing-specific approach to distributed intelligent control.

The concept was first introduced in 1989 by Suda [1] to address the shortcomings of extant industrial systems to address the pressures faced by manufacturers in the 1990s: i.e., increasingly stringent customer requirements for high-quality, customizable, low-cost products that can be delivered quickly, as well as the increasing levels of system complexity due to the distributed, concurrent, and stochastic nature of manufacturing systems.

During this time, research in distributed control led many to the realization that an "autonomous, distributed and co-operative" approach was required to address these issues. This new control software and hardware approach appeared to hold the most promise of realizing manufacturing systems that are both flexible (capable of reconfiguration) and responsive (capable of recovering from disturbances). However, past experiences with "green field" approaches like Flexible Manufacturing Systems (FMS) left a bad taste in the mouths of many manufacturers. Clearly, an incremental approach was required if new techniques of flexible automation were to be accepted by industry.

Central to this work on holonic manufacturing systems was concurrent work on the IEC 61499 standard for distributed intelligent control [2]. Although focused on the lower, real-time control level of the manufacturing control hierarchy, IEC 61499 appeared to show considerable promise for realizing holonic control (e.g., [3]).

Since early 2000, the IEC 61499 standard has gained more and more attention from academia and industry. However, it is still not clear if IEC 61499 has realized the goal

V. Mařík et al. (Eds.): HoloMAS 2019, LNAI 11710, pp. 3–12, 2019.
https://doi.org/10.1007/978-3-030-27878-6_1

of holonic manufacturing. In this paper we ask the question, after 20 years of research and development, has IEC 61499 realized its promised objectives and expected results? We begin by placing IEC 61499 in the context of its enabling technologies (Sect. 2), then focus on the efforts to implement the standard in industry (Sect. 3). We conclude this paper with a brief discussion of our perspectives on the IEC 61499 standard.

2 Realizing Intelligent Automation

In this section we focus on how IEC 61499 has been integrated with its enabling technologies to realize distributed intelligent industrial automation. Two perspectives are provided: design paradigms including object-oriented design, component-based design and service-oriented architecture, and computing paradigms including distributed intelligence, autonomic computing and cloud computing.

2.1 Design Paradigms

Object-oriented extensions (i.e., methods, inheritance, and interfaces) were first introduced to function blocks (FB) for automation programming in IEC 61131-3 [4], and made their way to IEC 61499. FBs encapsulate data structures and internal algorithms and can be instantiated working copies by type definitions. Polymorphism and inheritance are not often used in automation programming due to issues raised by computation cost and execution determinism, except that adapters provide kind of inheritance for similar FBs to share common interfaces.

Vyatkin et al. [5] proposed a conceptual object-oriented design (OOD) framework for modeling automation software based on IEC 61499 for potential benefits of intellectual property encapsulation and reuse. Dai and Vyatkin [6] proposed an object-oriented approach, including conversion of PLC code into an ECC and reuse of PLC code in an algorithm, to redesign distributed PLC control systems using IEC 61499 FBs. Two cases, modern building management systems [7] and airport baggage handling systems [8], were studied by using OOD to model IEC 61499 based system architectures.

The *component-based design paradigm* (CBD) uses coarse-grained and loose-coupled components with certain well-defined functionalities and pre-defined communication interfaces from a cohesive set of fine-grained objects. Compared with object-oriented modeling, component-based design models systems by functional components rather than physical objects, and multiple functions share a single algorithm with one generic event input instead of using dedicated events and algorithms for each method call [8]. Proposed frameworks in research for component-based design are mainly based on the Automation Component or Automaton Object (AC or AO) concept and then toward intelligent control [9–12].

Finally, the *service-oriented architecture (SOA) paradigm* approaches software system as a network of loose-coupled and discoverable services with formal defined interfaces communicating through messages [13]. Research projects of SOA in IEC 61499 based industrial automation systems were reviewed by Thramboulidis [14].

From a broad view, OOD, CBD and SOA can be considered as the same story. Designs are modeled through IEC 61499 FBs with mapping, creation, composition and execution of FBs as objects/components/services on different modeling levels. Some key points can be concluded as: (a) layered or multi-layer architecture is employed; (b) communication or interface design is focused; (c) reconfiguration, reuse, and flexibility is aimed.

2.2 Computing Paradigms

Distributed intelligence is a major step for distributed and intelligent automation, usually realized through *multi-agent systems* (MAS). With this computing paradigm, distributed and intelligent automation can be achieved through autonomous and cooperative agents that are capable of operating by themselves or collaborating with others to decide what actions to take, how and when to act in dynamic environments [15, 16]. More recently, MAS and SOA are increasingly considered as key enabling technologies to model IEC 61499 based industrial CPS with cloud and autonomous computing capabilities [17–19].

IEC 61499 based industrial automation systems are not only designed with fundamental features (e.g., distributed to be flexible, configurable, portable, and interoperable) but also envisioned for advanced ones (e.g., intelligent to incorporate self-managing capabilities) indicative of *autonomic computing* [20]. Intelligent behaviors require next generation industrial automation systems have flexible architectures (i.e., hardware and software) and adaptable strategies (i.e., rules and knowledge). The goal is to support real-time self-configuration (i.e., configure and reconfigure functions, structures, and processes), self-healing (i.e., detect and recover from disturbance and faults), self-optimization (i.e., improve and optimize what has been affected), and self-protection (i.e., identify and protect against safety and security attacks) for responsiveness to changes [15, 21]. More recently, the focus has been on formal mapping between IEC 61499 and service-oriented architectures (SOA) [22–24].

Cloud computing has also emerged as a new computing paradigm for industrial Cyber-physical Systems (CPS) [25]. As defined by the National Institute of Standards and Technology (NIST), cloud computing enables ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction [26]. For modeling industrial CPS, integration of the cyber (i.e., cloud) and the physical (i.e., devices) are enabled by encapsulating services in design entities (e.g., IEC 61499 FBs). Recent work in this area includes that of Karnouskos et al. [27], Dai et al. [6], and Demin et al. [28].

In summary, the key idea of above research is to introduce SOA and MAS, cloud and autonomic computing in the design modeling of industrial CPS to support flexibility and interoperability, and to realize part of self-management capabilities.

3 Implementing IEC 61499 Based Systems

The implementation of IEC 61499 based industrial automation systems requires first the development of IEC 61499 compliant engineering environments and then the application of them in research examples and industrial cases. This section is focused on how IEC 61499 engineering environments have been implemented.

3.1 Development of IEC 61499 Compliant Engineering Environments

Since the publication of the IEC 61499 standard, academic activities and industrial practices on developing engineering environments to implement IEC 61499 FB models have been conducted. Table 1 lists some typical projects for the development of IEC 61499 compliant engineering environments.

Table 1. IEC 61499 compliant engineering environments.

Product	Type[a]	Grade[b]	Capability[c]	Status	Technology
Holobloc [29]					
FBDK	ST	ACA	PIC	A	Oracle Java SE Platform; XML Document Type Definition
FBRT	RP				
4DIAC [30]					
4DIAC-IDE	ST	ACA	PIC	A	Eclipse Framework; C++ Programming Language
4DIAC-RTE	RP				
4DIAC-LIB	EL				
nxtControl [31]					

nxtSTUDIO	ST	IND	PIC	A	Microsoft .NET Framework;
					XML Document Type Definitions;
					ST Programming Language;
					Hardware Independent Engineering

nxtIECRT	RP				
nxtLIB	LB				
nxtHMI	RP				
ICS Triplex/ISaGRAF [32]					
Workbench	ST	IND	C	A	Microsoft Visual Studio Shell; Virtual Machine
Runtime	RP				
ISaVIEW	RP				
Fuber [33]					
FUBER	RP	ACA	n/a	A	Java; BeanShell
O3neida [34]					
Workbench	ST	ACA	n/a	I	Java; NetBeans; Eclipse
FBench	ST	ACA	n/a	I	Originates in O^{3}neida Workbench
Software Engineering Group, University of Patras [35]					
CORFU	EB	ACA	C	I	UML
Archimedes	RP	ACA	C	I	Model Integrated Mechatronics

[a] ST: Software Tool; RP: Runtime Platform; LB: Library of software components; EB: Runtime Embedded Tool

[b] ACA: Academic; IND: Industrial

[c] P: Portability; I: Interoperability; C: Configurability

Engineering environments usually include the following three components [36], in which some implementations integrate one or more components into one environment (e.g., Corfu) while others are with more function components (e.g., nxtHMI): (i) an integrated development environment to model designs (i.e., software tools), (ii) a runtime environment to execute programs (i.e., runtime platforms), (iii) a library to store elements (i.e., libraries of software components). Furthermore, these implementations can be classified into three categories. However, in this section we consider all as IEC 61499 compliant for simplicity although the utmost goal is to develop IEC 61499 based ones. More specifically, we consider IEC 61499 based implementations (e.g., Holobloc FBDK/FBRT), IEC 61131-3 based but IEC 61499 supported implementations (e.g., ISaGRAF Workbench/Runtime), and IEC 61499 and/or IEC 61131 based (hybrid) implementations (e.g., nxtControl nxtSTUDIO/nxtIECRT).

As proposed in IEC 61499-4, three key features are expected in developing IEC 61499 based systems, devices and software tools [37]. In this section, multi-source is defined as the described objects (i.e., devices, libraries, software tools, runtime platforms) are provided by different suppliers and/or developed through different techniques: i.e., configurability of multi-source devices by multi-source software tools, portability of multi-source libraries between multi-source software tools, and interoperability of multi-source devices in multi-source runtime platforms.

As the first IEC 61499 feasibility demonstration, the Function Block Development Kit (FBDK) is developed and maintained to support fundamental features and incorporate new updates of IEC 61499 [29]. It is widely used in academic research, experiment design, technical training, and commercial product test. The currently released version FBDK 3.3 in August 2018 enables one to build and test IEC 61499 library elements including data types, function block types, resource types, device types, network segment types, adapter types, functions and system configurations [29]. One of the key characteristics is to support design patterns, including Layered MVC, Local Multicast, Proxy, Tagged Data, Matrix Framework, Time-Stamped Messaging. The new version also supports many enhanced features for generation of HTML documentation and experimental features of transient EC states and compact encoding of management commands and responses.

The ISaGRAF is the first commercial software environment for industrial implementation of both IEC 61131-3 and IEC 61499 (IEC 61131-3 based but IEC 61499 supported) to develop distributed automation and control systems. It includes the software tool built on Microsoft Visual Studio Shell (i.e., ISaGRAF Workbench) that provides plug-in functions such as editor, documentation, library management, and a runtime platform based on a Virtual Machine (i.e., ISaGRAF Runtime) that executes target independent code (TIC) generated by control applications [32].

A more comprehensive industrial solution package for efficient and intelligent automation of distributed control systems comes from nxtControl. It includes the software tool nxtSTUDIO to integrate all automation tasks, the library nxtLIB to offers pre-fabricated software objects (CATs, Composite Automation Types), the runtime platform nxtIECRT to support hybrid control paradigms (i.e., IEC 61131-3 and/or IEC 61149 based), and the user interface nxtHMI together with SCADA to enable multi-client/multi-server visualization [31].

The Eclipse 4DIAC provides open source solutions to the development of IEC 61499 based industrial automation and control solutions in research activities and industrial adoption. It includes the software tool 4DIAC-IDE based on the Eclipse framework, the runtime platform 4DIAC-RTE supporting online reconfiguration of applications and real-time execution of function block types, the library 4DIAC-LIB containing function blocks, adapters and sub-applications [30].

In summary, the above four projects are widely recognized in academia and industry. Portability of library elements between software tools, configurability of devices by software tools, and interoperability of devices in runtime platforms have been formally and/or informally tested [37]. Generally, solutions from Holobloc, Eclipse 4DIAC, and nxtControl meet these requirements in which nxtControl provides more configurability classes [37]. ISaGRAF provides only portability through its Workbench, interoperability by its Runtime, and configurability between its Workbench and Runtime [32].

There are also some other projects providing IEC 61499 compliant engineering environments. Fuber (FUnction Block Execution Runtime, last updated October 2017) is an interpreter developed in Java and BeanShell and runs IEC 61499 based applications [33]. O3neida Workbench and FBench are pioneered by Christensen for experimental use in the development and test of Automation Objects (AOs) for industrial automation and control [5, 34]. The CORFU framework developed by Thrambouldis consists of an IEC 61499 compliant Engineering Support System (ESS) based on UML modeling approach, a 4-layer architecture to facilitate the application of FB model, and a development process based on model transformation [35]. Thrambouldis also proposed Archimedes system platform based on Model Integrated Mechatronics (MIM) architecture to support the development of complex mechatronic manufacturing systems that enable runtime reconfiguration by using IEC 61499 FB model. Most of these projects have not been active in recent years, but are still valuable reference implementations for developing IEC 61499 based automation and control systems.

3.2 Applications of IEC 61499 Compliant Engineering Environments

Wide adoption of IEC 61499 by industry has been a challenge. However, there are still some early adopters to build automation and control systems through applying previous discussed engineering environments. Table 2 provides an overview of these applications from different software vendors. As discussed in previous sections, engineering environments from Holobloc and 4DIAC are mainly used in academic research for case studies. For example, the case of airport baggage handling systems is thoroughly studied by Dai and Vyatkin for their research on IEC 61499. Industrial-grade applications from nxtControl and ISaGRAF can be found from their official websites. The main application domain includes smart grids, smart factories, smart buildings, etc.

Table 2. Applications of the IEC 61499 standard.

IEC 61499 application	Software
Meat processing plant and fertilizer production plant [38, 39] Airport baggage handling systems [40] Smart grids [41, 42] Transportation line for shoe manufacturing [43]	Holobloc [29]
Smart grids [30] Pick & Place station [30]	4DIAC [30]
HVAC and lighting control [31] Fertilizer production plant [31]	nxtControl [31]
Food processing embedded control [44] Research centre data acquisition and control on a drying test bench [44] Control of hydraulic parameters of district heating region in Sofia [45] High-speed train monitoring and control [46] Railway safety functions in mining transport system [47] I-8000 waste water treatment system [47] Smart grids [48]	ISaGRAF [32]

4 Summary and Perspectives

The IEC 61499 standard has advanced considerably since its introduction: i.e., from the early work on the reference architecture, to more recent work on function block execution and semantics. As noted previously, in recent years we have seen increasing interest in academic and industrial applications. However, compared with widely used IEC 61131-3 standard in industrial practices, IEC 61499 is mainly promoted in academia and seldom accepted by industry. If the adoption of this new technology is viewed in the context of the well-known three-phase S-shaped “logistic curve” [49], IEC 61499 was in the first phase of *Launch* promoted by innovators in 2012 for the second edition and is now in the transition to the second phase of *Takeoff* with some early adopters [50]. Arguably, there is still be a long way to go to reach the third phase of *Maturity* unless some key issues are not be reasonably solved.

Three main types of challenges for industrial adoption of IEC 61499 are prevalent: industrial concerns on business development, technical issues related to standard itself, and societal aspects of trained personnel. A summary of these challenges is provided in Table 3.

Considering these critical factors, much effort has been put into realizing successful industrial adoption of IEC 61499. Some reasonable solutions have been suggested such as providing feasible methods, techniques, and guidelines for designing new IEC 61499 based systems, applications and components; redesigning existing IEC 61131-3 based systems so they are compliant with IEC 61499; and providing qualified courses and hands-on training to learn and use IEC 61499. Efforts in the design of new IEC 61499 based systems must continue to focus on execution models and semantics (e.g., [9, 13]) as well as more recent work on autonomic and cloud computing discussed in Sect. 2. Although the work on redesigning existing IEC 61131-3 based systems has led to opportunities for industrial implementations of IEC 61499, this approach is still an intermediate step of full transition from existing IEC 61131-3 based systems to IEC 61499 based ones.

Table 3. Main challenges for industrial adoption of IEC 61499.

Main challenge	Detailed explanation
Industrial concerns	Large amount of existing IEC 61131-3 based systems Little demand for a completely new design approach Huge cost incurred by introducing new technologies Much effort required without significant performance improvement
Technical issues	Lack of system development and implementation guidelines and tools Few proved methods for migration from IEC 61131-3 to IEC 61499 Same execution semantics may result in different system behaviors Long way to realize its promised objectives and expected results
Societal aspects	New qualification requirements for control engineers New course design for teaching and learning IEC 61499 New industrial training for applying and using IEC 61499 Unwilling to be the first to use unmatured new technologies

Acknowledgments. The authors wish to thank the Natural Sciences and Engineering Research Council of Canada, Spartan Controls, and the Suncor Energy Foundation for their generous support of this research through grant CDE 486462-15.

References

1. Suda, H.: Future factory automation system formulated in Japan. Techno Japan **22**, 15–25 (1989)
2. Christensen, J.: HMS: initial architecture and standards directions. In: Proceedings of the 1st European Conference on Holonic Manufacturing Systems, Hannover, pp. 1–20 (1994)
3. Fletcher, M., Garcia-Herreros, E., Christensen, J., Deen, S., and Mittmann, R.: An open architecture for holonic cooperation and autonomy. In: Proceedings of the 11th International Workshop on Database and Expert Systems Applications, pp. 224–230 (2000)
4. Werner, B.: Object-oriented extensions for IEC 61131-3. IEEE Ind. Electron. Mag. **3**(4), 36–39 (2009)
5. Vyatkin, V., Christensen, J.H., Lastra, J.L.M.: OOONEIDA: an open, object-oriented knowledge economy for intelligent industrial automation. IEEE Trans. Industr. Inf. **1**(1), 4–17 (2005)
6. Dai, W., Vyatkin, V.: Redesign distributed PLC control systems using IEC 61499 function blocks. IEEE Trans. Autom. Sci. Eng. **9**(2), 390–401 (2012)
7. Vyatkin, V.: Software engineering in industrial automation: state-of-the-art review. IEEE Trans. Industr. Inf. **9**(3), 1234–1249 (2013)
8. Dai, W., Vyatkin, V., Christensen, J.H.: Applying IEC 61499 design paradigms: object-oriented programming component-based design and service-oriented architecture. In: Zoitl, A., Strasser, T. (eds.) Distributed Control Applications: Guidelines, Design Patterns, and Application Examples with the IEC 61499. CRC Press, Boca Raton (2017)

9. Vyatkin, V.: Intelligent mechatronic components: control system engineering using an open distributed architecture. In: Proceedings of 8th IEEE Conference on Emerging Technologies and Factory Automation, Lisbon, pp. 277–284 (2003)
10. Cengic, G., Ljungkrantz, O., Akesson, K.: A framework for component based distributed control software development using IEC 61499. In: Proceedings of 11th IEEE Conference on Emerging Technologies and Factory Automation, Prague, pp. 782–789 (2006)
11. Lepuschitz, W., Zoitl, A.: An engineering method for batch process automation using a component-oriented design based on IEC 61499. In: Proceedings of 13th IEEE International Conference Emerging Technologies and Factory Automation, Hamburg, pp. 207–214 (2008)
12. Zoitl, A., Prähofer, H.: Guidelines and patterns for building hierarchical automation solutions in the IEC 61499 modeling language. IEEE Trans. Industr. Inf. **9**(4), 2387–2396 (2013)
13. Jammes, F., Smit, H.: Service-oriented paradigms in industrial automation. IEEE Trans. Industr. Inf. **1**(1), 62–70 (2005)
14. Thramboulidis, K.: Service-oriented architecture in industrial automation systems - the case of IEC 61499: a review, arXiv preprint (2015)
15. Marik, V., McFarlane, D.: Industrial adoption of agent-based technologies. IEEE Intell. Syst. **20**(1), 27–35 (2005)
16. Brennan, R.W.: Toward real-time distributed intelligent control: a survey of research themes and applications. IEEE Trans. Syst. Man Cybern. - Part C: Appl. Rev. **37**(5), 744–765 (2007)
17. Leitão, P., Colombo, A.W., Karnouskos, S.: Industrial automation based on cyber-physical systems technologies: prototype implementations and challenges. Comput. Ind. **81**, 11–25 (2016)
18. Dai, W., Dubinin, V.N., Christensen, J.H., Vyatkin, V., Guan, X.: Toward self-manageable and adaptive industrial cyber-physical systems with knowledge-driven autonomic service management. IEEE Trans. Industr. Inf. **13**(2), 725–736 (2017)
19. Dai, W., Riliskis, L., Wang, P., Vyatkin, V., Guan, X.: A cloud-based decision support system for self-healing in distributed automation systems using fault tree analysis. IEEE Trans. Industr. Inf. **14**(3), 989–1000 (2018)
20. IBM: An architectural blueprint for autonomic computing. IBM White Paper (2006)
21. Mubarak, H., Göhner, P.: An agent-oriented approach for self-management of industrial automation systems. In: Proceedings of 8th IEEE International Conference on Industrial Informatics, pp. 721–726 (2010)
22. Lepuschitz, W., Zoitl, A., Vallée, M., Merdan, M.: Toward self-reconfiguration of manufacturing systems using automation agents. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **41**(1), 52–69 (2011)
23. Strasser, T., Froschauer, R.: Autonomous application recovery in distributed intelligent automation and control systems. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **42**(6), 1054–1070 (2012)
24. Kaindl, H., Vallée, M., Arnautovic, E.: Self-representation for self-configuration and monitoring in agent-based flexible automation systems. IEEE Trans. Syst. Man Cybern.: Syst. **43**(1), 164–175 (2013)
25. Givehchi, O., Trsek, H., Jasperneite, J.: Cloud computing for industrial automation systems - a comprehensive overview. In: Proceedings of 18th IEEE Conference on Emerging Technologies and Factory Automation, Cagliari, pp. 1–4 (2013)
26. Mell, P., Grance, T.: The NIST definition of cloud computing. National Institute of Standards and Technology, Gaithersburg, Maryland (2011)

27. Karnouskos, S., et al.: A SOA-based architecture for empowering future collaborative cloud-based industrial automation. In: Proceedings of 38th Annual Conference of the IEEE Industrial Electronics Society, Montreal, pp. 5766–5772 (2012)
28. Demin, E., Patil, S., Dubinin, V., Vyatkin, V.: IEC 61499 distributed control enhanced with cloud-based web-services. In: Proceedings of 10th IEEE Conference on Industrial Electronics and Applications, Auckland, pp. 972–977 (2015)
29. Holobloc, April 2019. https://www.holobloc.com/
30. Eclipse 4DIAC, April 2019. https://www.eclipse.org/4diac/
31. nxtControl, April 2019. https://www.nxtcontrol.com/en/references/
32. ICS Triplex ISaGRAF, April 2019. http://www.isagraf.com/index.htm
33. Fuber, April 2019. https://sourceforge.net/projects/fuber/
34. O3neida FBench, April 2019. https://www.controleng.com/articles/new-iec-61499-function-block-programming-safety-energy-systems-resources-from-isa-o3neida/
35. SEG Corfu, April 2019. http://seg.ece.upatras.gr/Corfu/dev/index.htm
36. Vyatkin, V.: IEC 61499 as enabler of distributed and intelligent automation: state-of-the-art review. IEEE Trans. Industr. Inf. **7**(4), 768–781 (2011)
37. Christensen, J.H., Strasser, T., Valentini, A., Vyatkin, V., Zoitl, A.: The IEC 61499 function block standard: software tools and runtime platforms. ISA Automation Week, Orlando (2012)
38. Tait, P.: A path to industrial adoption of distributed control technology. In: Proceedings of 3rd IEEE International Conference on Industrial Informatics, Perth, pp. 86–91 (2005)
39. Zoitl, A., Strasser, T., Hall, K., Staron, R., Sünder, C., Favre-Bulle, B.: The past, present, and future of IEC 61499. In: Mařík, V., Vyatkin, V., Colombo, A.W. (eds.) HoloMAS 2007. LNCS (LNAI), vol. 4659, pp. 1–14. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-74481-8_1
40. Black, G., Vyatkin, V.: Intelligent component-based automation of baggage handling systems with IEC 61499. IEEE Trans. Autom. Sci. Eng. **7**(2), 337–351 (2010)
41. Vyatkin, V., Zhabelova, G., Higgins, N., Schwarz, K., Nair, N.: Towards intelligent smart grid devices with IEC 61850 interoperability and IEC 61499 open control architecture. In: Proceedings of IEEE PES Transmission and Distribution Conference and Exposition, New Orleans, pp. 1–8 (2010)
42. Zhabelova, G., Vyatkin, V.: Multiagent smart grid automation architecture based on IEC 61850/61499 intelligent logical nodes. IEEE Trans. Industr. Electron. **59**(5), 2351–2362 (2012)
43. Colla, M., Brusaferri, A., Carpanzano, E.: Applying the IEC-61499 model to the shoe manufacturing sector. In: Proceedings of 11th IEEE Conference on Emerging Technologies and Factory Automation, Prague, pp. 1301–1308 (2006)
44. Centris Technologies, April 2019. https://www.centristech.com/en/
45. KIBERNETIKA, April 2019. http://www.kibernetika-bg.com/
46. EKE Electronics, April 2019. https://www.eke-electronics.com/
47. ICP DAS USA, April 2019. https://www.icpdas-usa.com/, (April 2019)
48. Brusaferri, A., Ballarino, A., Carpanzano, E.: Reconfigurable knowledge-based control solutions for responsive manufacturing systems. Stud. Inform. Control **20**(1), 31–42 (2011)
49. Shapiro, C., Varian, H.R.: Information Rules: a Strategic Guide to the Network Economy. Harvard Business Press, Boston (1999)
50. Strasser, T., et al.: The IEC 61499 function block standard: launch and takeoff. ISA Automation Week, Orlando (2012)