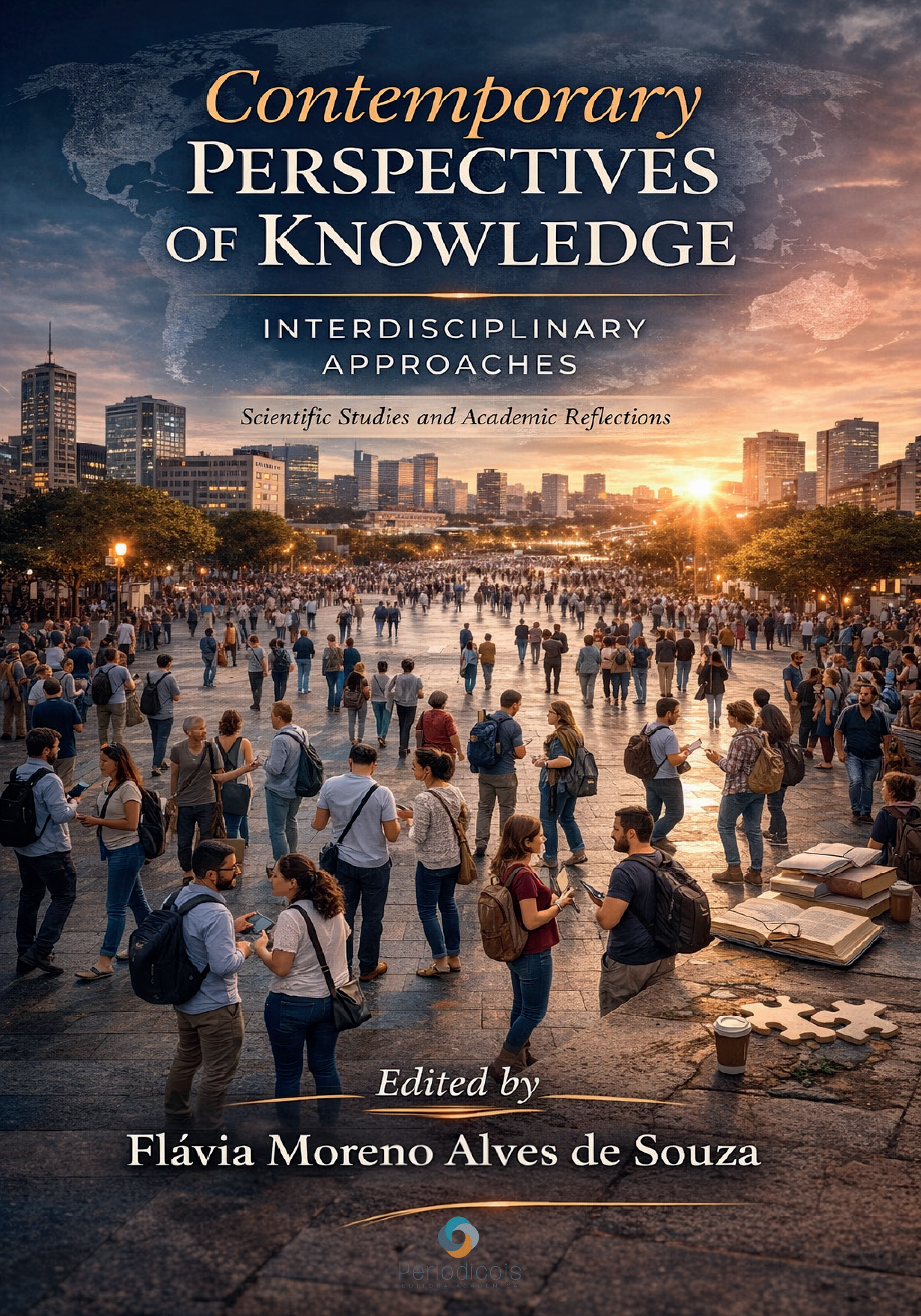


Contemporary PERSPECTIVES OF KNOWLEDGE

INTERDISCIPLINARY APPROACHES

Scientific Studies and Academic Reflections

Edited by

Flávia Moreno Alves de Souza

Periodicojs
EDITORA ACADÊMICA

Graphic Design, Layout and Cover

Academic Publisher Periodicojs

Language

Portuguese and English

International Cataloging-in-Publication Data (CIP)

(Brazilian Book Chamber, SP, Brazil)

C761 Contemporary Perspectives of Knowledge: Interdisciplinary Approaches/ Flávia Moreno Alves de Souza (org) – João Pessoa: Periodicojs publisher, 2026.

E-book: il. color.

Includes bibliography
ISBN: 978-65-6010-196-8

1. Free themes. I. Souza, Flávia Moreno Alves. II. Title

CDD 370

Prepared by Dayse de França Barbosa CRB 15-553

Index for systematic catalog:

Indexes for systematic catalog:

1. Education: 370

Work without funding from public or private bodies.

The published works have been submitted to peer review and evaluation (double-blind), with respective acceptance letters in the publisher's system.

The work is the result of studies and research from the Interdisciplinary Studies in Human Sciences section of the Humanities in Perspective book collection.

Filipe Lins dos Santos
President and Senior Editor of Periodicojs

CNPJ: 39.865.437/0001-23

Rua Josias Lopes Braga, n. 437, Bancários, João Pessoa - PB - Brazil
website: www.periodicojs.com.br
instagram: @periodicojs

Chapter 21

MULTI-BRAND INTEGRATION IN ROBOTIC SYSTEMS: STRATEGIES FOR HARMONIZING DIFFERENT PLATFORMS IN INDUSTRIAL ENVIRONMENTS

MULTI-BRAND INTEGRATION IN ROBOTIC SYSTEMS: STRATEGIES FOR HARMONIZING DIFFERENT PLATFORMS IN INDUSTRIAL ENVIRONMENTS

André Paccelli de Oliveira[1]

Abstract: Industrial plants increasingly operate robotic fleets composed of equipment from multiple manufacturers, driven by mergers and acquisitions, capacity expansions, application-specific niches, and the growing adoption of collaborative robots alongside conventional industrial arms. This chapter examines multi-brand integration as a structural engineering condition rather than a transitory problem to be eliminated. It compares the programming languages, simulation environments, controller architectures, and market positioning of ABB, FANUC, KUKA, Yaskawa/Motoman, and Universal Robots, and characterizes the challenges that manufacturer diversity introduces across five interdependent layers: programming, industrial communication, maintenance, regulatory and safety compliance, and organizational knowledge management. Building on this diagnosis, the chapter presents complementary strategies for standardizing programming conventions, adopting multi-brand offline-programming and simulation tools, exploring software-abstraction frameworks such as ROS-Industrial, and establishing change-governance and cross-training practices. Particular attention is given to the OPC UA for Robotics companion specification as a vendor-neutral interoperability layer, supported by field-measured latency and performance evidence from recent engineering literature, alongside the regulatory implications of the revised ISO 10218-1:2025 and ISO 10218-2:2025 standards

1 Founder and technical director of Pacelli Robotics, a company specializing in programming, simulation, and optimization of industrial robotic cells. He holds a technical degree in Mechatronics from SENAI-CLC in Curitiba, with an emphasis on industrial electricity, electronics, sensors, PLCs, systems, and industrial robotics. He also has a bachelor's degree in Electrical Engineering from the Centro Universitário Curitiba (UniCuritiba). In 2020, he was invited to teach the mini-course "Programming of Manipulator Robots in the Automotive Industry" during the ROSIE 2020 academic event, consolidating his work in the areas of robotics and industrial automation.

and IEC 62443 cybersecurity guidance. A structured six-phase methodology for implementing and expanding multi-brand cells is proposed, followed by illustrative scenarios demonstrating gains in flexibility, maintenance efficiency, and international scalability. The chapter concludes that multi-brand integration competence should be regarded as a strategic asset that broadens engineering's freedom of choice, rather than a necessary evil to be tolerated only when single-brand standardization becomes impractical.

Keywords: Multi-brand integration. Industrial robotics. Robot interoperability. OPC UA for Robotics. Industrial automation standardization.

INTRODUCTION: THE EVOLUTION OF INDUSTRIAL PLANTS TOWARD MULTI--BRAND ENVIRONMENTS

For decades, industrial automation followed a relatively predictable logic: a plant was designed, specified, and deployed around a single robot manufacturer. This choice responded to legitimate strategic reasons — a consolidated commercial relationship, standardized spare parts, a single learning curve for the maintenance and programming team, and long-term technical support contracts. The result was a homogeneous robotic fleet, in which every controller, every programming language, and every communication protocol belonged to a single ecosystem.

That scenario has changed irreversibly. Corporate mergers and acquisitions unified, under the same plant, production lines that were historically independent and served by different automation suppliers. Capacity expansions, driven by aggressive delivery deadlines, led companies to purchase whichever robot was available on the market at the moment of need, not necessarily the brand already installed. New applications — machine vision, palletizing, laser welding, collaborative handling — each brought in whichever manufacturer best served that specific technical niche. Add to this the maturing of collaborative robots, a segment in which Universal Robots has consolidated a leading position, often coexisting, in the same cell, with a large industrial robot from another manufacturer responsible for the high-payload task.

Industry figures confirm the magnitude of this transformation. According to the World Robotics 2025 report from the International Federation of Robotics (IFR), 542,000 industrial robots were installed worldwide in 2024, more than double the volume recorded ten years earlier, raising the global operational stock to 4.66 million units in simultaneous use (IFR, 2025). This growth was not concentrated in a single supplier: the report itself evidences the geographic and sectoral dispersion of demand, with the electronics industry surpassing, for the first time, the automotive sector as the largest consumer of robots (IFR, 2025). In a market of this size and diversity, it is statistically inevitable that medium and large plants will accumulate, over their operating life, equipment from multiple manufacturers.

It is also worth noting that this diversification is neither a passing phenomenon nor a regional one. The very geographic distribution of installations — with Asia accounting for 74% of new deployments in 2024, versus 16% in Europe and 9% in the Americas (IFR, 2025) — reveals markets at distinct stages of technological maturity, each with historical preferences for particular manufacturers, which translates, in the case of multinational companies, into sister plants equipped with completely different robot brands, even when manufacturing the same product under the same corporate process standard. An automation engineer working across global operations frequently discovers that the robot standard adopted at headquarters simply is not the most competitive, available, or locally supported option at the subsidiary where a new cell needs to be installed.

The contemporary automation engineer, therefore, rarely has the privilege of operating in a single-brand environment. Their real challenge is different: making ABB, FANUC, KUKA, Yaskawa, Universal Robots, and other manufacturers coexist productively, safely, and sustainably within the same plant — without technological heterogeneity turning into organizational heterogeneity, with fragmented teams, siloed knowledge, and systematic rework at every intervention. This chapter starts from exactly that premise: manufacturer diversity is not a problem to be eliminated, but a structural condition to be managed with method, engineering discipline, and long-term vision.

It is also important to situate this discussion within a broader economic perspective. The decision to standardize a robotic fleet around a single manufacturer is usually justified by a promise of operational simplicity that, in practice, rarely holds up over the entire life cycle of the plant. Suppliers change their commercial strategy, discontinue product lines, alter delivery times due to geopolitical supply-chain disputes, and the technically most suitable robot for a new application does not always belong to the historically adopted brand. Treating manufacturer homogeneity as a goal in itself — rather than as an eventual consequence of specific technical and commercial decisions — tends to produce suboptimal purchasing decisions, driven more by the convenience of keeping a single supplier than by a genuine cost-benefit analysis of each application. This chapter therefore advocates a change in posture: multi-brand integration competence should be seen as a strategic asset that expands engineering's freedom of choice, not as a necessary evil to be tolerated only when homogeneity becomes impractical.

KEY DIFFERENCES BETWEEN ABB, FANUC, KUKA, YASKAWA, AND UNIVERSAL ROBOTS

Before discussing harmonization strategies, it is necessary to understand, with engineering precision — and not merely in commercial terms — where the real differences between the main platforms found on the shop floor actually lie. These differences are not merely cosmetic; they directly affect teams' learning curves, commissioning time, and the cell's integration architecture. Table 1 summarizes the main points of contrast, which will be detailed manufacturer by manufacturer in the following subsections.

Table 1 – Key differences between industrial robotic platforms

Manufacturer	Native language	Simulation/OLP environment	Controller architecture	Predominant market focus
ABB	RAPID	RobotStudio	IRC5 / OmniCore	General industry, painting, assembly
FANUC	TP / KAREL	ROBOGUIDE	R-30iB	Welding, high-speed handling
KUKA	KRL	KUKA.Sim / WorkVisual	KR C4 / KR C5	Automotive, heavy payload
Yaskawa/Motoman	INFORM III	MotoSim EX	YRC1000	Arc welding, material handling
Universal Robots	URScript / Polyscope	Polyscope / URSim	CB-Series / eSeries	Collaborative, low/medium payload

Source: prepared by the author (2026), based on manufacturers' technical documentation.

ABB

Over its trajectory, ABB has built a programming ecosystem centered on the RAPID language, structured modularly around modules, routines, and strongly typed data types, executed on the IRC5 controller architecture — currently transitioning to the OmniCore platform, which has a more open design oriented toward third-party software integration. The RobotStudio environment, widely adopted for offline programming and simulation, is recognized for the quality of its model library and the depth of its integration with native CAD data, which makes it especially competitive in painting, precision assembly, and general handling applications. From an organizational standpoint, ABB tends to attract engineering teams with strong familiarity in structured programming, since RAPID's syntax bears conceptual similarities to general-purpose programming languages.

FANUC

FANUC operates with two programming levels: the TP (Teach Pendant) language, aimed at direct programming via the teaching terminal and widely used on the shop floor for quick adjustments, and the KAREL language, a higher-level language used for complex application logic and peripheral integration. This duality is a distinctive characteristic of the platform: teams that master only TP programming frequently struggle when they need to customize more sophisticated behaviors, which require KAREL knowledge. ROBOGUIDE is the market standard for simulation and offline programming of FANUC cells, being particularly robust in spot-welding and high-speed handling applications — niches in which the brand has historically consolidated market leadership, especially in the Asian and North American automotive sector.

KUKA

KUKA uses KRL (KUKA Robot Language), a language with syntax close to structured Pascal, executed on KR C-family controllers — currently in their fourth and fifth generations (KR C4 and KR C5). The company is historically associated with heavy-payload applications in the automotive sector, especially in body-handling and heavy assembly operations, although it has consistently expanded its portfolio into smaller-scale segments and collaborative robotics. KUKA.Sim, integrated with the WorkVisual environment, offers kinematic and cycle simulation features that stand out in applications requiring fine synchronization between multiple external axes, such as rotary tables and linear tracks — a common scenario in large automotive cells.

Yaskawa/Motoman

Yaskawa, whose industrial robot line is marketed under the Motoman brand, uses the INFORM III language on the YRC1000 controller architecture, and is historically recognized for its robustness in arc-welding applications — a niche in which it has one of the largest installed bases in the world. The MotoSim EX simulation environment is the standard for offline programming on this platform. A relevant differentiator for Yaskawa, discussed in more depth in Section 2.5 of this chapter, is its relative pioneering role in offering a native OPC UA server on its controllers, directly implementing the information model of the OPC UA for Robotics companion specification (YASKAWA MOTOMAN, 2025) — a concrete step toward the interoperability that will be central to the discussion on multi-brand integration.

Universal Robots

Universal Robots follows a programming philosophy structurally distinct from the four manufacturers above, built around the Polyscope graphical interface and the URScript scripting language, with simplified syntax oriented toward composing logic blocks via drag-and-drop. This approach significantly lowers the entry barrier for operators without advanced robotics training, but in turn imposes payload, reach, and speed constraints that structurally differentiate cobots from conventional large industrial robots. This difference in purpose — not merely in language — is what explains why, in practice, modern plants frequently combine a Universal Robots cobot for close human-interaction tasks with an industrial robot from another manufacturer for high-payload, high-speed tasks within the same cell, creating exactly the kind of multi-brand environment that motivates this chapter.

Reading these five platforms together reveals a central point: each manufacturer developed its own low-level programming language, its own philosophy for handling frames and coordinate systems, and its own controller architecture. A trajectory written in RAPID for an ABB IRC5 is not portable, without translation, to a FANUC R-30iB controller — not even the concepts of reference point (TCP, work object, user frame) are named or implemented identically across manufacturers. Beyond the programming layer, there are relevant differences in functional-safety philosophy, in the implementation of protective zones, in how each controller exposes digital and analog input/output variables, and in the depth of access allowed to low-level drive-system parameters. An integrator who has mastered only one platform tends to underestimate these differences until the moment they actually need to bring into production a cell with two or more manufacturers coexisting on the same line.

INTEGRATION CHALLENGES BETWEEN DIFFERENT AUTOMATION ARCHITECTURES

The coexistence of multiple brands within the same plant generates challenges that manifest across at least five distinct layers: the programming layer, the industrial communication layer, the maintenance layer, the regulatory and safety layer, and the organizational layer. Each of these layers demands a specific engineering response, and failing to address any one of them in isolation tends to compromise the outcome of the others.

It is worth noting that these five layers do not operate independently of one another; they feed back into each other in ways that are not always obvious in an initial diagnosis. A decision seemingly confined to the programming layer — such as the choice not to standardize variable naming across manufacturers — tends to surface, months later, as a maintenance-layer problem, when a generalist technician cannot quickly locate the fault-handling logic in a cell they are unfamiliar with. Likewise, a gap in the organizational layer — the absence of a technical owner with visibility across the entire multi-brand fleet, rather than over a single specific brand — tends to manifest as recurring delays in the regulatory layer, when compliance updates for a specific manufacturer stop being tracked due to a lack of clear ownership over that set of equipment. For this reason, assessing a plant's multi-brand integration maturity must necessarily consider all five layers together, not in isolation.

Programming layer

Each native language has its own syntax, its own data-handling model, and its own way of structuring routines and subroutines. A programmer experienced in KUKA's KRL does not automatically transfer their productivity to a Yaskawa INFORM III environment; the learning curve, even if shorter on the second or third platform learned, is never zero. In plants with staff turnover, this fragmentation of knowledge becomes a real operational risk, since the availability of qualified labor for each specific brand does not always match the actual distribution of installed equipment. This risk is compounded in regions where local technical training has historically favored a particular brand, creating an imbalance between the supply of qualified professionals and the actual composition of the installed fleet.

Industrial communication layer

Historically, each manufacturer favored its own protocols and forms of integration for communicating with programmable logic controllers, MES systems, and SCADA, which required proprietary gateways, manual variable mappings, and point-to-point integrations specific to each combination of robot brand and logic-controller brand. This scenario has been changing with the growing adoption of OPC UA as a vendor-neutral communication layer — a topic explored further in Section 2.5 — but historical heterogeneity is still present in the vast majority of plants in operation, especially those whose robotic fleet was deployed over different decades and under different controller generations.

Maintenance layer

Spare parts, diagnostic tools, program-backup software, and calibration procedures are specific to each manufacturer. This means maintaining separate technical-support contracts, separate periodic training for each platform, and, frequently, segmented spare-parts inventories — which raises the maintenance cost per installed robot compared with a homogeneous fleet. In large plants, this additional cost tends to be absorbed without much scrutiny; in medium-sized plants, however, it frequently becomes the decisive — though mistaken — argument for trying to force fleet homogenization at any cost, even when that means discarding fully functional equipment.

Regulatory and safety layer

An additional challenge, historically less discussed in the literature on multi-brand integration but of growing practical relevance, is the regulatory layer. The 2025 publication of the revised ISO 10218-1:2025 and ISO 10218-2:2025 standards — which replace the 2011 editions after nearly a decade of technical work — brought structural changes to how the safety of industrial robots and their applications is specified, including a new robot classification system, explicit functional-safety requirements, and, for the first time, the normative incorporation of cybersecurity requirements for networked robotic systems (ASSOCIATION FOR ADVANCING AUTOMATION, 2025; ISO, 2025). In multi-brand environments, this regulatory revision is particularly sensitive: each manufacturer advances at its own pace in adapting its controllers to the new requirements, which can result, during the transition period, in a plant where robots of different brands meet different stages of regulatory compliance, requiring the safety and engineering teams to track compliance separately by manufacturer and by controller version.

The cybersecurity dimension deserves separate attention. As robotic controllers from different manufacturers increasingly expose network interfaces — whether via OPC UA or via proprietary remote-diagnostics protocols — the attack surface of a multi-brand cell ceases to be the simple sum of each robot's individual surfaces and comes to include the integration points between them as well. Frameworks such as IEC 62443, whose 2024 edition of Part 2-1 reorganized security-program requirements around structured security elements and a four-level maturity model (IEC 62443-2-1:2024), offer the conceptual framework of zones and conduits that allows a multi-brand cell to be segmented so that a vulnerability in one manufacturer's controller does not automatically propagate to the line's other equipment. This zone-and-conduit segmentation approach, combined with explicit authentication between OPC UA clients and servers, is now considered best practice for any networked industrial environment, and becomes even more relevant when multiple manufacturers — each with its own firmware-update and vulnerability-patching cycle — share the same communication infrastructure.

Organizational layer

Perhaps the most underestimated challenge is the organizational one: engineering teams tend to informally specialize in a single brand, creating knowledge silos. When an expansion project requires simultaneous mastery of two or more platforms, the company frequently discovers that it does not internally possess the necessary cross-brand competence, becoming dependent on external integrators for any intervention involving more than one manufacturer. This type of dependency, when not deliberately managed, tends to be self-perpetuating: the lack of internal competence reinforces the hiring of third parties, and the recurring hiring of third parties reduces the incentive to develop the competence internally.

STRATEGIES FOR STANDARDIZING PROGRAMMING AND COMMUNICATION

The response to these challenges does not lie in eliminating heterogeneity — which would be economically unviable in most cases — but in building abstraction and standardization layers that reduce the cost of living with it. Four complementary strategies stand out in multi-brand integration practice.

Standardizing naming conventions and logical structure

Regardless of each robot's native language, it is possible — and advisable — to impose on the engineering team a corporate standard for variable naming, routine structuring (initialization, execution, fault handling, finalization), and code-commenting conventions. This standardization does not eliminate the syntactic difference between RAPID, KRL, KAREL, or INFORM III, but it ensures that any programmer on the team, upon opening the code of any cell, immediately recognizes the program's overall logic, regardless of the brand of robot executing it. In practice, this is usually formalized in an engineering-standard document — a true corporate robotic-programming style guide — reviewed periodically and applied as an acceptance criterion for any new project, whether in-house or outsourced.

Multi-brand offline-programming tools

The adoption of simulation and offline-programming platforms capable of operating with multiple manufacturers simultaneously has become one of the most effective practical standardization strategies. Tools such as RoboDK, for example, currently support more than a thousand robot models from dozens of different manufacturers — including ABB, FANUC, KUKA, Yaskawa, and Universal Robots — within a single simulation environment, allowing mixed cells to be composed in the same virtual scene and, from a common trajectory logic, generating each controller's specific native code through dedicated post-processors (ROBODK, 2025; SIEMENS, 2025).

It is worth qualifying the role of each tool, however, according to the project's scale and complexity. RoboDK tends to be more associated with one-off applications, feasibility studies, and smaller-scale projects, and is rarely adopted as the central environment for simulating and virtually commissioning a complete industrial station involving multiple robots, peripherals, and safety devices operating in a synchronized manner. For that type of project — typical of automotive lines and large-scale discrete manufacturing — the market predominantly relies on Siemens Process Simulate (part of the Tecnomatix portfolio), widely established as one of the industry's reference tools for simulation, reachability and ergonomics studies, cycle-time validation, and virtual commissioning of complete cells and lines with robots from multiple manufacturers (SIEMENS, 2025). RoboDK remains a valid and accessible alternative, mainly due to its lower entry cost and gentler learning curve, but a realistic picture of multi-brand offline-programming tools would not be complete without acknowledging the central role of Process Simulate in larger and more complex projects, where its adoption — and the corresponding know-how within the engineering team — tends, in practice, to be a more recurring requirement than that of RoboDK itself.

Unlike the commercial discourse of these tools' own vendors — which tends to disclose only aggregate gain percentages — there is also engineering literature with explicit measurement methodology on the impact of simulation-assisted configuration in robotic cells. A recent study combining artificial-intelligence techniques with automated configuration of manufacturing cells measured, in a digital-twin-controlled environment, an average reduction of 21.19% in the area occupied by the cell and of up to 20.15% in operating cycle time, with metrics compared directly against the configuration obtained through traditional design methods (RESEARCHGATE, 2024). This type of measurement — with percentages tied to a reproducible experimental methodology, rather than merely a figure from promotional material — is the standard of evidence that multi-brand integration engineering should seek before justifying investment in a new tool or standardization process, and it is the standard this chapter adopts as its preferred reference whenever available.

The same logic applies to the adoption of offline programming: cases documented by the vendor ecosystem itself — such as that of the Chinese robot manufacturer Estun, which reported reductions of around 30% in virtual commissioning time and 20% in total deployment cost after adopting simulation software supporting more than 30 robot brands (SIEMENS, n.d.) — are useful as evidence of real industrial-scale impact, but should be read as a case study reported by the tool's vendor, not as an independent measurement. Reading engineering studies with their own methodology alongside documented industrial cases together is what allows the reader to properly calibrate their return expectations, rather than anchoring the investment decision solely on the most favorable figure available.

Software abstraction frameworks

In academic settings and in more complex applications, middleware frameworks such as ROS-Industrial have been advancing precisely in the direction of offering a hardware-abstraction layer that allows application logic to be written once and reused across different robotic platforms, with each manufacturer's specific driver absorbing the low-level particularities (ROS-INDUSTRIAL, 2023; CONTROL ENGINEERING, 2025). Although its adoption in purely industrial settings is still more common in research applications, material handling, and experimental-cell integration than in high-volume automotive lines, the growth trend of ROS 2 as a multi-manufacturer robotic middleware is a relevant indicator of where software standardization in the sector is heading (INTERNATIONAL JOURNAL OF INTELLIGENT ROBOTICS AND APPLICATIONS, 2024).

Change governance and cross-training

None of the three previous strategies is sustainable without a fourth dimension, of a less technical and more organizational nature: change governance and cross-training of the team. Naming standards only remain alive if there is a formal code-review process before any program is released to production, regardless of the robot's manufacturer. Likewise, investment in a multi-brand offline-programming tool only pays off if the team is effectively trained and encouraged to use it as its single work environment, rather than merely as a technological curiosity running alongside the actual programming process, which continues to happen in isolation within each proprietary software package. Companies that treat multi-brand integration as a one-off engineering project, without instituting these ongoing governance mechanisms, tend to see standardization silently deteriorate over two or three years, as new projects and new external integrators reintroduce divergent practices.

OPC Robotics stands for a complete motion device system (INTERNATIONAL

FEDERATION OF ROBOTICS, [n.d.]).

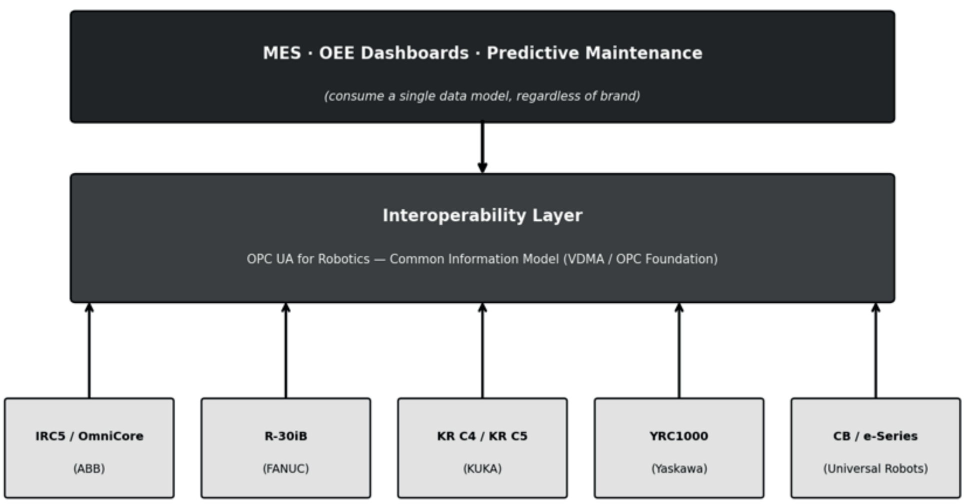
INTEROPERABILITY BETWEEN CONTROLLERS AND INDUSTRIAL SYSTEMS

If programming standardization solves part of the problem at the engineering layer, interoperability at the industrial-communication level solves the other half — arguably the most critical one for MES operations, supervision, and predictive maintenance at scale. On this front, the most relevant development of the past decade is the OPC UA for Robotics companion specification, jointly developed by VDMA (the German mechanical-engineering association) and the OPC Foundation.

This specification defines a standardized, vendor-neutral information model capable of representing the state, configuration parameters, and diagnostic data of any robotic system — whether a traditional industrial robot, a mobile robot, or a collaborative robot — under a single common semantic structure, allowing a single OPC UA client to collect comparable data from robots of different manufacturers without the need for individual mappings (VDMA/OPC FOUNDATION, 2025; VISION SYSTEMS DESIGN, 2019; IFR, n.d.). Part 1 of the specification, focused on asset management, condition monitoring, and vertical integration, is already implemented by multiple manufacturers; Yaskawa, for example, provides a native OPC UA server on its Motoman controllers based exactly on this information model (YASKAWA MOTOMAN, 2025). In October 2025, the initiative advanced with the publication of version 1.02 of the specification, which now also covers high-level control — program loading, system start/stop commands, and detailed task-level status monitoring — expanding OPC UA's scope from a purely informational layer to one that also handles supervisory commands (VDMA OPC ROBOTICS INITIATIVE, 2025).

This advance is strategically relevant to this chapter's topic because it shifts the point of multi-brand integration from each manufacturer's proprietary layer to a common semantic layer, situated above the individual controllers. In practice, this means that MES systems, Overall Equipment Effectiveness (OEE) dashboards, and predictive-maintenance platforms can be developed once, consuming the same OPC UA information model, regardless of whether the physical cell is equipped with an ABB IRC5, a FANUC R-30iB, or a Yaskawa YRC1000 controller. This does not eliminate the need to know each robot's native programming for the application tasks themselves — welding, painting, handling — but it removes an entire layer of custom integration that previously had to be rebuilt every time a new manufacturer was added to the plant. Figure 1 diagrams this relationship: the native controllers remain intact at the base of the architecture, and it is the interoperability layer that becomes the new single point of integration for higher-level systems.

Figure 1 – OPC UA interoperability layer above native controllers



Source: prepared by the author (2026), based on the information model of the OPC UA for Robotics specification (VDMA/OPC Foundation).

What the literature actually measured, rather than merely described

The functional description of the OPC UA for Robotics specification, given so far, needs to be complemented by field-measured performance evidence, lest the reader conclude that interoperability carries negligible latency cost simply because the standard-setter's institutional material describes it that way. Engineering literature with its own experimental methodology shows a more nuanced picture. A 2026 study on secure OPC UA data aggregation in legacy systems measured, across 14,287 samples collected on a test bench, end-to-end latencies between 2.6 and 24.7 milliseconds, with a standard deviation of 3.0 milliseconds — values fully compatible with condition monitoring and MES, but already approaching the acceptable limit for fine real-time control loops (ARXIV, 2026). An earlier study, dedicated specifically to characterizing OPC UA's real-time performance over networks with deterministic synchronization (TSN), systematically evaluated the behavior of cyclic publisher/subscriber communication under different network conditions, offering a reproducible benchmarking methodology that any engineering team can adapt to validate its own multi-brand network architecture before putting it into production (KIRDAN, 2023).

These figures matter because they dispel a common oversimplification: that adopting OPC UA as a neutral layer solves interoperability without additional engineering cost. In practice, the latency introduced by the abstraction layer must be sized and validated for each application — an OEE dashboard can tolerate tens of milliseconds with no perceptible impact, but a synchronization loop between a robot and a high-speed vision system may not tolerate the same latency budget. The practical recommendation stemming from this literature is to measure the network architecture's actual latency before assuming the interoperability layer is ready to support a specific supervisory function, rather than taking the real-time promise found in the specification's promotional material for granted.

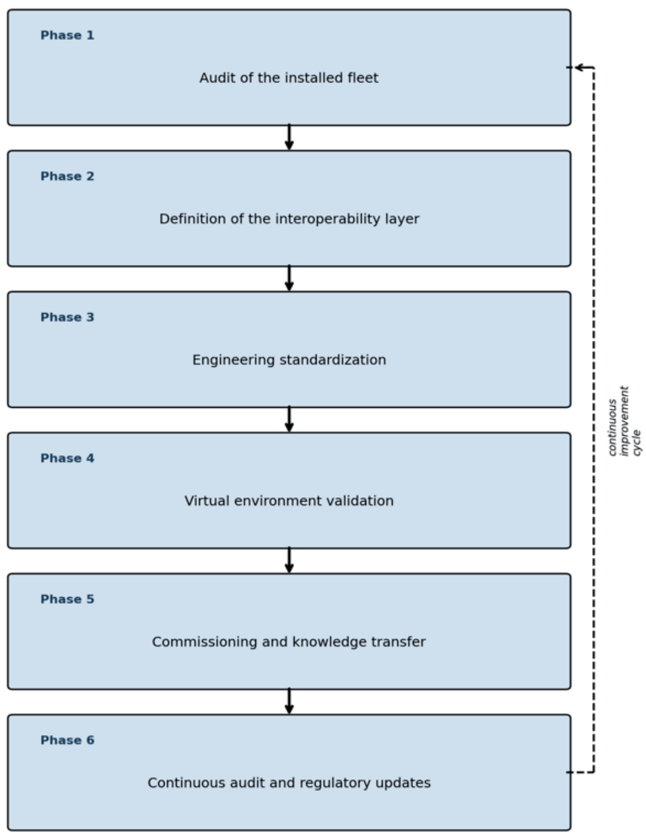
From the standpoint of real industrial-scale field evidence — not merely laboratory-bench evidence — a 2025 case study documented the development of a mobile-robot fleet-management system (AGVs and AMRs) operating with equipment from multiple manufacturers within the automotive sector, under the applied-research project GreenAuto. The system unified, on a single web platform, position and status data from robots of different brands and models, each originally using its own data format, through a data-abstraction and protocol-translation layer developed specifically for that purpose (MDPI, 2025). Although the subject of this study is mobile robots, not fixed robotic arms, the engineering lesson is directly transferable to this chapter's topic: real-scale multi-brand interoperability requires deliberate investment in a translation layer, and does not emerge spontaneously from the mere physical coexistence of equipment in the same plant.

In addition, recent academic discussions on reference architectures for Industry 4.0 — such as RAMI 4.0 and the Asset Administration Shell (AAS) — have reinforced the importance of treating interoperability as a design requirement from the cell's conception, rather than as a later patch. The literature points to the lack of standardization between systems as one of the main remaining obstacles to the full implementation of cyber-physical production systems in real industrial environments (ESTRADA-JIMENEZ et al., 2025). In this context, communication-level interoperability and regulatory compliance in safety and cybersecurity, discussed in Section 2.3.4, should be treated as complementary facets of the same problem: a multi-brand cell that is technically interoperable but does not adequately segment its network trust domains simply trades a fragmentation risk for an exposure risk.

METHODOLOGIES FOR IMPLEMENTING AND EXPANDING MULTI-BRAND CELLS

Practical experience in multi-brand integration projects allows for the consolidation of a structured six-phase methodology, applicable both to the deployment of new cells and to the expansion of existing lines with equipment from additional manufacturers. It is important to stress that these phases should not be interpreted as a strictly sequential, irreversible flow: in real projects, it is common for the audit of the installed fleet to reveal, already at an advanced stage, the need to revisit the definition of the interoperability layer, or for virtual-environment validation to expose inconsistencies that require adjustments to the very engineering standard defined in the previous phase. The methodology should therefore be treated as an iterative refinement cycle, not as a list of steps to simply be checked off — Figure 2 represents this cycle.

Figure 2 – Six-phase cycle for implementing and expanding multi-brand cells



Source: prepared by the author (2026).

Phase 1 — Audit of the installed fleet

A complete survey of models, firmware versions, supported communication protocols, and available documentation for every robot already installed in the plant. This phase frequently reveals inconsistencies between what appears in the asset register and what is actually in operation — outdated controller versions, optional communication modules not installed, or technical documentation lost over the years. The audit should also include an assessment of each robot's compliance level with current safety standards, so as to anticipate update needs before they become blockers at a later stage of the project.

Phase 2 — Definition of the interoperability layer

A conscious choice of which communication layer will be adopted as the corporate standard — typically OPC UA, given its open-specification nature, its growing adoption among leading manufacturers, and its brand neutrality — and mapping which robots already natively support this layer and which will require translation gateways. In this same phase, it is advisable to already define the network-segmentation architecture following the zones-and-conduits principles of IEC 62443, so that the communication topology is designed with security from the outset, rather than retrofitted onto a flat network already in operation.

Phase 3 — Engineering standardization

Definition of the corporate standard for naming conventions, program structure, and safety conventions, applicable to every brand present in the plant, plus the choice of the simulation and offline-programming tool that will be adopted as the single validation environment for trajectories, regardless of the physical robot's manufacturer. This phase should produce a living document, reviewed periodically as new brands or new controller generations are added to the installed fleet.

Phase 4 — Virtual environment validation

Before any physical intervention in the cell, the new programming logic and planned trajectories must be validated in a simulation environment, with checks for collisions, singularities, and joint limits specific to each robot model involved. This step is particularly critical in multi-brand cells, in which the workspaces of robots from different manufacturers frequently overlap physically, and small differences in how each controller calculates joint limits and protective zones can produce unexpected behaviors that only manifest under specific operating conditions.

Phase 5 — Commissioning and knowledge transfer

Physical deployment followed by structured documentation and cross-training of the maintenance team, ensuring that knowledge about the new cell is not confined to a single specialist or a single external vendor — mitigating the organizational risk discussed in Section 2.3.5. Multi-brand commissioning must mandatorily include integrated safety tests involving all robots present in the cell simultaneously, not merely the isolated validation of each piece of equipment separately.

Phase 6 — Continuous audit and regulatory updates

Unlike a single-brand deployment, in which regulatory compliance can be treated as a one-off event tied to commissioning, a multi-brand cell requires a continuous audit cycle, since each manufacturer evolves its firmware, its safety documentation, and its adherence to regulatory revisions

— such as the transition to ISO 10218-1:2025 and ISO 10218-2:2025 — at its own pace. Establishing a periodic compliance-review routine by manufacturer, not merely by plant, is what ensures that technological heterogeneity does not, over time, turn into regulatory-risk heterogeneity.

ILLUSTRATIVE SCENARIOS OF GAINS IN FLEXIBILITY, MAINTENANCE, AND SCALABILITY

The situations below are presented as illustrative scenarios, representative of recurring patterns observed in multi-brand integration projects, and are intended to make the concepts discussed in the previous sections tangible — they do not refer to any specific plant or client, preserving the didactic nature of this work.

Scenario 1 — Expansion of an automotive line with welding robots from two manufacturers

A plant with a history of spot-welding robots from a single manufacturer receives, as part of an expansion project, a new line equipped with robots from another manufacturer, chosen for shorter delivery availability. Adopting a common offline-programming environment for both brands allows the process engineering team to validate the welding trajectories of both lines on the same simulation platform before physical commissioning, reducing on-site fine-tuning time and preventing the new brand's learning curve from impacting the start-up schedule. The prior standardization of process-variable naming, applied from the design phase onward, also allows the same line supervisory system to handle cycle data from both brands in a unified way, without the need for two separate monitoring panels.

Scenario 2 — Hybrid cell with an industrial robot and a cobot

An inspection-and-palletizing cell combines a large industrial robot, responsible for moving heavy loads, with a collaborative robot from another manufacturer, responsible for the inspection step involving close interaction with the operator. Standardizing the communication layer via OPC UA allows the cell's supervisory system to monitor both robots' status under the same data model, simplifying the development of the supervision panel and reducing the effort required to maintain shop-floor software. The physical coexistence of the two pieces of equipment in this scenario also demands extra attention to overlapping safety zones, addressed already at the design phase following the robot-classification logic introduced by the 2025 regulatory revision.

Scenario 3 — Consolidation of multi-brand maintenance

A plant that historically kept maintenance teams segmented by manufacturer standardizes diagnostic procedures and variable naming across the brands present, allowing generalist technicians to handle the first response in any cell, escalating specialized manufacturer support only for more complex failures — reducing the average response time to unplanned downtime. In this scenario, the cross-training program is structured around modules common to all brands — safety fundamentals, diagnostic reading, backup procedures — complemented by shorter, brand-specific modules for each controller's particularities.

Scenario 4 — International expansion with a single process standard and different manufacturers by region

A multinational company decides to replicate, at a newly acquired plant in another country, the same production process already validated at headquarters — but the robot manufacturer predominant at headquarters does not have the same local technical-support presence in the new region. Instead of forcing the import of the same brand at any cost, corporate engineering chooses to specify the process independently of manufacturer, documenting trajectories, cycle times, and quality parameters in functional terms, and allowing the subsidiary to select the most competitive local manufacturer. Engineering standardization and the OPC UA interoperability layer, already adopted as a global corporate standard, ensure that the new plant's performance indicators remain comparable to those of headquarters, even while operating with a completely different robot manufacturer.

FUTURE TRENDS FOR THE INTEGRATION OF HETEROGENEOUS ROBOTIC PLATFORMS

The OPC UA for Robotics specification is expected to continue expanding its scope beyond asset management, moving toward high-level supervisory control, as already signaled by version 1.02, published in October 2025 (VDMA OPC ROBOTICS INITIATIVE, 2025). This trajectory suggests that, in the coming years, functions currently exclusive to each manufacturer's proprietary software — such as remote program loading and system start/stop commands — will become accessible through a common interface, further reducing the effort of vertical integration in multi-brand plants.

In parallel, the ROS 2 middleware is expected to continue maturing in larger-scale industrial applications as well, as it incorporates real-time and safety features more aligned with that context's requirements (INTERNATIONAL JOURNAL OF INTELLIGENT ROBOTICS AND APPLICATIONS, 2024). Although its adoption in high-volume automotive lines remains, for now, more limited than in research applications and material handling, the sustained growth of the ROS-Industrial community and the growing interest of mid-sized integrators in this abstraction layer indicate a gradual but consistent adoption trajectory.

Reference architectures such as the Asset Administration Shell and RAMI 4.0 tend to consolidate as a common semantic layer for integrating heterogeneous assets in Industry 4.0 plants, offering a shared vocabulary that transcends both the robot manufacturer and the supervisory-system vendor (ESTRADA-JIMENEZ et al., 2025). This normative consolidation should proceed hand in hand with the maturing of digital twins, capable of representing, within a single virtual environment, cells composed of robots from multiple manufacturers, with prior validation of reconfiguration and AI-assisted optimization (MO et al., 2023; JOURNAL OF INTELLIGENT MANUFACTURING, 2026). The convergence between multi-brand digital twins and interoperability layers such as OPC UA promises to further reduce the time between the decision to reconfigure a cell and its actual validation in production.

The regulatory dimension is also expected to keep evolving rapidly. The transition of plants in operation to the new requirements of ISO 10218-1:2025 and ISO 10218-2:2025 — which substantially expanded the treatment of contact risks between robot and operator and, for the first time, formalized cybersecurity requirements within the industrial-robot safety standard itself — is expected to extend over several years, as manufacturers update their lines and integrators adapt their cell-design procedures (ASSOCIATION FOR ADVANCING AUTOMATION, 2025). In the meantime, it is reasonable to expect that regulatory compliance will itself become yet another relevant evaluation criterion when planning the coexistence of multiple brands within the same plant, complementing the traditional criteria of cost, availability, and technical performance.

CONCLUDING REMARKS

There is no more free lunch in choosing a robot manufacturer: whoever decides to lock into a single brand for short-term convenience will pay that bill sooner or later, whether through a delivery deadline the preferred brand cannot meet, or through a new application for which it simply is not the best technically available option. The right question is no longer "which robot brand should we standardize on?" but rather "is our engineering ready to make two or three brands coexist without generating chaos?"

The previous sections of this chapter attempted to answer that second question along three concrete fronts, summarized in Figures 1 and 2: a neutral communication layer — today led by OPC UA for Robotics, with performance already measured rather than merely promised — a programming-standardization discipline that survives staff turnover, and an implementation methodology that treats regulatory compliance and network segmentation as part of the project from day one, not as a last-minute patch.

If you are planning your plant's next expansion, the practical test is simple: ask your team whether they can, today, answer for any installed robot — of any brand — in under five minutes, without needing to call the integrator. If the answer is no, the problem is not the next brand you are going to buy. It is the foundation missing before it arrives.

REFERENCES

ARXIV. Secure Data Bridging in Industry 4.0: An OPC UA Aggregation Approach for Including Insecure Legacy Systems. arXiv, 2026. Available at: https://arxiv.org/pdf/2601.10929. Accessed on: Aug. 23, 2026.

ASSOCIATION FOR ADVANCING AUTOMATION (A3). Updated ISO 10218: Major Advancements in Industrial Robot Safety Standards Now Available. A3, 2025. Available at: https://www.automate.org/robotics/news/updated-iso-10218-major-advancements-in-industrial-robot-safety-standards-now-available. Accessed on: Aug. 23, 2026.

ASSOCIATION FOR ADVANCING AUTOMATION (A3). Updated ISO 10218 | Answers to Frequently Asked Questions (FAQs). A3, 2025. Available at: https://www.automate.org/robotics/blogs/updated-iso-10218-faq. Accessed on: Aug. 23, 2026.

CONTROL ENGINEERING. Advance plant-floor robotics three ways with ROS-Industrial tools. Control Engineering, 2025. Available at: https://www.controleng.com/advance-plant-floor-robotics-three-ways-with-ros-industrial-tools/. Accessed on: Aug. 23, 2026.

ESTRADA-JIMENEZ, L. A. et al. Digital twin-based self-learning decision-making framework for industrial robots in manufacturing. The International Journal of Advanced Manufacturing Technology, Springer Nature, 2025. Available at: https://link.springer.com/article/10.1007/s00170-025-15844-w. Accessed on: Aug. 23, 2026.

INTERNATIONAL ELECTROTECHNICAL COMMISSION (IEC). IEC 62443-2-1:2024 — Security program requirements for IACS asset owners. Geneva: IEC, 2024.

INTERNATIONAL FEDERATION OF ROBOTICS (IFR). World Robotics 2025 — Industrial Robots. Frankfurt: IFR, Sept. 2025. Available at: https://ifr.org/ifr-press-releases/news/global-robot-demand-in-factories-doubles-over-10-years. Accessed on: Aug. 23, 2026.

INTERNATIONAL FEDERATION OF ROBOTICS (IFR). Faster robot communication through the OPC Robotics Companion Specification. IFR Blog. Available at: https://ifr.org/post/faster-robot-communication-through-the-opc-robotics-companion-specification. Accessed on: Aug. 23, 2026.

INTERNATIONAL JOURNAL OF INTELLIGENT ROBOTICS AND APPLICATIONS. Inquiring the robot operating system community on the state of adoption of the ROS 2 robotics middleware. Springer Nature, 2024. Available at: https://link.springer.com/article/10.1007/s41315-024-00393-4. Accessed on: Aug. 23, 2026.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (ISO). ISO 10218-1:2025 — Robotics — Safety requirements — Part 1: Industrial robots. Geneva: ISO, 2025.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (ISO). ISO 10218-2:2025 — Robotics — Safety requirements — Part 2: Industrial robot applications and robot cells. Geneva: ISO, 2025.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (ISO). ISO 22166-1:2021 — Robotics — Modularity for service robots — Part 1: General requirements. Geneva: ISO, 2021.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (ISO). ISO 22166-201:2024 — Robotics — Modularity for service robots — Part 201: Common information model for modules. Geneva: ISO, 2024.

JOURNAL OF INTELLIGENT MANUFACTURING. A review on integration of digital twins within industry 4.0. Springer Nature, 2026. Available at: https://link.springer.com/article/10.1007/s10845-026-02851-1. Accessed on: Aug. 23, 2026.

KIRDAN, E. Real-Time Performance of OPC UA. arXiv, 2023. Available at: https://arxiv.org/pdf/2310.17052. Accessed on: Aug. 23, 2026.

MDPI. Integrated Fleet Management of Mobile Robots for Enhancing Industrial Efficiency: A Case Study on Interoperability in Multi-Brand Environments Within the Automotive Sector. Applied Sciences, 2025. Available at: https://www.mdpi.com/2076-3417/15/13/7235. Accessed on: Aug. 23, 2026.

MO, F.; CHAPLIN, J. C.; SANDERSON, D.; REHMAN, H. U.; MONETTI, F. M.; MAFFEI, A.; RATCHEV, S. A framework for manufacturing system reconfiguration and optimisation utilising digital twins and modular artificial intelligence. Robotics and Computer-Integrated Manufacturing, v. 82, 102524, 2023.

OPC FOUNDATION. OPC 40010 — OPC UA for Robotics, Part 1: Vertical Integration. Available at: https://reference.opcfoundation.org/specs/OPC-40010-1/4. Accessed on: Aug. 23, 2026.

OPC FOUNDATION. Robotics — Markets Collaboration. Available at: https://opcfoundation.org/markets-collaboration/robotics/. Accessed on: Aug. 23, 2026.

RESEARCHGATE. Rapid and automated configuration of robot manufacturing cells. 2024. Available at: https://www.researchgate.net/publication/384678675_Rapid_and_automated_configuration_of_robot_manufacturing_cells. Accessed on: Aug. 23, 2026.

ROBODK INC. Robot Offline Programming. RoboDK, 2025. Available at: https://robodk.com/offline-programming. Accessed on: Aug. 23, 2026.

SCIENCEDIRECT. Evolution of safety requirements in industrial robotics: Comparative analysis of ISO 10218-1/2 (2011 vs. 2025) and integration of ISO/TS 15066. Elsevier, 2026. Available at: https://www.sciencedirect.com/science/article/pii/S2590123026015203. Accessed on: Aug. 23, 2026.

SIEMENS. Reducing onsite virtual commissioning time by 30 percent and overall costs by 20 percent — Estun Automation case study. Siemens Digital Industries Software, n.d. Available at: https://

resources.sw.siemens.com/en-US/case-study-estun-automation/. Accessed on: Aug. 23, 2026.

SIEMENS. RoboDK — Simulation and offline programming software for industrial robots. Siemens Digital Industries Software, 2025. Available at: https://www.siemens.com/en-us/products/robodk/. Accessed on: Aug. 23, 2026.

SIEMENS. Tecnomatix Process Simulate — Robotics and automation simulation software. Siemens Digital Industries Software, 2025. Available at: https://plm.sw.siemens.com/en-US/tecnomatix/products/process-simulate-software/. Accessed on: Aug. 23, 2026.

TANDFONLINE. A Digital Twin-based paradigm for programming and control of cooperating robots in reconfigurable production systems. International Journal of Computer Integrated Manufacturing, published online Nov. 15, 2024. Available at: https://www.tandfonline.com/doi/full/10.1080/0951192X.2024.2428683. Accessed on: Aug. 23, 2026.

UMATI. OPC 40010 UA for Robotics. Available at: https://umati.org/industries_robotics/. Accessed on: Aug. 23, 2026.

VDMA OPC ROBOTICS INITIATIVE. VDMA Releases OPC UA Robotics Companion Specification 1.02 for Seamless Industrial Robot Integration. Frankfurt, Oct. 23, 2025. Available at: https://www.industrialautomationindia.in/news/vdma-opc-robotics-companion-specification-1-02-digital-production. Accessed on: Aug. 23, 2026.

VISION SYSTEMS DESIGN. OPC UA machine vision and robotics specifications introduced at automatica. Vision Systems Design, 2019. Available at: https://www.vision-systems.com/factory/article/16751729/opc-ua-machine-vision-and-robotics-specifications-introduced-at-automatica. Accessed on: Aug. 23, 2026.

YASKAWA MOTOMAN. OPCUA Server on Yaskawa Motoman Controllers. Yaskawa Motoman Knowledge Base, 2025. Available at: https://knowledge.motoman.com/hc/en-us/articles/24393065124114-OPCUA-Server-on-Yaskawa-Motoman-Controllers. Accessed on: Aug. 23, 2026.