Machine Translated by Google
RCMOS – Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

Microservices architecture and artificial intelligence applied to SAP S/4HANA: the role of ABAP RAP and SAP BTP in industrial process automation.

Microservices architecture and artificial intelligence applied to sap S/4hana: the role of ABAP RAP and SAP BTP in industrial process automation

Microservices architecture and artificial intelligence applied to SAP S/4HANA: the role of ABAP RAP and SAP BTP in industrial process automation

François Pierre Fagundes Francelino

SUMMARY

This article aims to analyze, from a historical and evolutionary perspective, the transition from monolithic SAP system architectures to a microservices-oriented paradigm, highlighting the role of the ABAP RESTful Application Programming Model (RAP) and the SAP Business Technology Platform (BTP) in the automation of industrial processes mediated by Artificial Intelligence (AI). It begins with the technological trajectory that starts with SAP R/2 and SAP R/3, passes through the consolidation of SAP ECC, and culminates in the disruption promoted by SAP S/4HANA and in-memory computing based on SAP HANA. The theoretical foundations of microservices architecture, Industry 4.0, and AI applied to enterprise resource planning (ERP) systems are discussed, establishing a conceptual link between these domains and the technical capabilities of RAP and BTP. Through qualitative and exploratory bibliographic research, supported by official SAP technical sources and related academic literature, this paper proposes a reference architecture that integrates ABAP Cloud microservices, embedded AI services (SAP AI Core and AI Foundation), and event buses (SAP Event Mesh) to enable industrial automation scenarios, with emphasis on predictive maintenance and real-time quality control. The results indicate that RAP, by operationalizing the clean core principle, acts as an enabling element for transactional microservices within the digital core, while BTP extends this core through data-decoupled services, integration, and artificial intelligence.

It can be concluded that the convergence between these components represents not only an incremental technological evolution, but a paradigmatic shift in the way ERP systems support the automation of production processes, requiring, on the other hand, increased attention to governance, information security, and the management of technical debt resulting from decades of customization.

Keywords: SAP S/4HANA. ABAP RAP. SAP BTP. Microservices. Artificial Intelligence. Industrial Automation.

ABSTRACT

This article aims to analyze, from a historical and evolutionary perspective, the transition of SAP's monolithic architectures toward a microservices-oriented paradigm, highlighting the role of the ABAP RESTful Application Programming Model (RAP) and SAP Business Technology Platform (BTP) in Artificial Intelligence (AI)-mediated industrial process automation. The analysis departs from the technological trajectory that begins with SAP R/2 and SAP R/3, passes through the consolidation of SAP ECC, and culminates in the disruption promoted by SAP S/4HANA and SAP HANA in-memory computing. The theoretical foundations of microservices architecture, Industry 4.0, and AI applied to Enterprise Resource Planning (ERP) systems are discussed, establishing a conceptual bridge between these domains and the technical capabilities of RAP and BTP. Through qualitative, exploratory bibliographic research supported by official SAP technical documentation and related academic literature, a reference architecture is proposed that integrates ABAP Cloud microservices, embedded AI services (SAP AI Core and AI Foundation), and event backbones (SAP Event Mesh) to enable industrial

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS – Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

automation scenarios, with emphasis on predictive maintenance and real-time quality control. Results indicate that RAP, by operationalizing the clean core principle, functions as an enabler of transactional microservices within the digital core, while BTP extends this core through decoupled data, integration, and artificial intelligence services. It is concluded that the convergence of these components represents not merely an incremental technological evolution but a paradigmatic shift in how ERP systems support production process automation, while demanding heightened attention to governance, information security, and the management of technical debt accumulated over decades of customization.

Keywords: SAP S/4HANA. ABAP RAP. SAP BTP. Microservices. Artificial Intelligence. Industrial Automation.

INTRODUCTION

The automation of industrial processes has ceased to be, in the last two decades, a domain... restricted to supervisory control and data acquisition (SCADA) systems and has become integrated- if, in an increasingly narrow way, to corporate business management systems (Enterprise Resource Planning (ERP). In this context, SAP SE, a leading global provider of solutions ERP has consolidated, over more than five decades, a technological trajectory that has evolved from Centralized monolithic architectures for distributed, service-oriented platforms, and More recently, integrated microservices via the cloud.

The launch of SAP S/4HANA in 2015, and its technological foundation based on the database of SAP HANA in-memory data represented a turning point that was not limited to gains. computational performance. This transition imposed the need to rethink the model itself. of enterprise application development, culminating in the proposal of ABAP RESTful. Application Programming Model (RAP) and the consolidation of SAP Business Technology Platform (BTP) as an environment for extension, integration, and artificial intelligence. These two The elements — RAP and BTP — together constitute the technical foundation upon which it rests. the proposal for a clean digital core, capable of supporting architectures of microservices without compromising stability and the ability to continuously update the central system.

The research problem guiding this article can be formulated as follows: in what way... The historical evolution of SAP architectures, from the mainframe era to S/4HANA, explains and It underlies the role currently played by ABAP RAP and SAP BTP in enabling AI-driven microservices architectures for automation of Industrial processes? The hypothesis is that the adoption of microservices in the ecosystem SAP does not represent an abrupt break, but rather the result of an evolutionary process. cumulative, in which each technological generation — R/2, R/3, mySAP, NetWeaver, ECC, HANA — is considered.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS - Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

S/4HANA prepared the technical and organizational conditions for the subsequent generation.
The overall objective of this work is to analyze, in a technical and scientific manner, the historical trajectory of SAP architectures and their convergence with microservices principles, examining
Specifically, how do ABAP RAP and SAP BTP operationalize this convergence?
To enable industrial automation scenarios supported by Artificial Intelligence. The objectives include:
Specifically, the aim is to: (i) reconstruct the historical evolution of SAP platforms, relating-
a) to the broader transformations in software engineering; (ii) to systematize the fundamentals
theorists of microservices architecture, Industry 4.0 and AI applied to ERP systems; (iii)
Describe the technical workings of ABAP RAP and its relationship to the clean core concept;
(iv) characterize the data, integration and AI services provided by SAP BTP; and (v)
propose a reference architecture that integrates these elements in an applied scenario of
automation of industrial processes.
From a methodological point of view, a qualitative approach of an exploratory nature is adopted.
descriptive, supported by bibliographic and documentary research, whose procedures are
detailed in section 4. The relevance of the topic is justified by the growing demand from
industrial organizations, through systems capable of integrating factory floor data, algorithms
machine learning and real-time business transaction processes — demand
This cannot be fully met by purely monolithic architectures, given their
low capacity for independent scaling, incremental deployment, and adoption of
Heterogeneous programming languages and paradigms.
The article is structured in nine sections, including this introduction. Section 2 presents the...
The historical evolution of SAP architectures. Section 3 discusses the theoretical framework on
Microservices, Industry 4.0, and AI applied to ERP. Section 4 describes the methodology. The sections
Sections 5 and 6 detail ABAP RAP and SAP BTP, respectively. Section 7 presents the architecture.
a proposed reference and an illustrative case. Section 8 discusses the results in light of the literature.
Section 9 presents the final considerations.

HISTORICAL EVOLUTION OF SAP ARCHITECTURES: FROM R/2 TO S/4HANA

Understanding the current role of ABAP RAP and SAP BTP necessarily requires a
a reconstruction of the historical trajectory of SAP software architectures, since each
The technological generation incorporated specific responses to the limitations of the previous generation, to
while preserving structural principles that remain in effect to this day,

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS - Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

such as business process-centric data modeling and proprietary language
ABAP (Advanced Business Application Programming).

SAP R/2 and SAP R/3: the era of monolithic systems

Founded in 1972 in Germany, SAP (Systemanalyse und Programmentwicklung) launched, at
In the late 1970s, the R/2 system, an integrated business management solution, was designed...
to operate in mainframe environments. OR/2 already incorporated a principle that would remain.
as a distinctive feature of the company: the integration of functional modules — finance, materials,
production — based on a single, shared database, eliminating duplication of
typical information from the isolated departmental systems that were then predominant.
In 1992, SAP launched R/3, migrating from a mainframe-centric architecture to...
three-tier client-server model (presentation, application, and database), then
emerging in the software industry. This transition represented the first major disruption.
The company's architectural structure allows for the distribution of processing load among different
Hardware layers and paving the way for interoperability with operating systems.
and third-party databases. OR/3 also consolidated the use of the ABAP language as
The main instrument for customizing and extending the standard system, a practice that would become,
Decades later, one of the main challenges to architectural modernization—a phenomenon
known as technical customization debt.

mySAP.com, SAP NetWeaver, and the openness to integration.

In the early 2000s, with the popularization of the commercial internet, SAP launched the initiative.
mySAP.com, orienting its portfolio towards e-commerce scenarios and integration between
companies (business-to-business). This phase culminated in the launch of SAP in 2003.
NetWeaver, an integration platform based on open standards — such as XML and
Web services — which began to act as an intermediary layer between the core
transactional R/3 (later ECC) and external systems.
NetWeaver introduced components that would become conceptual precursors to today's SAP.
BTP, like Process Integration (PI), is geared towards orchestrating messages between systems.
heterogeneous, and the Enterprise Portal, responsible for consolidating user interfaces of
Different applications in a single access point. In 2006, SAP launched SAP ERP Central.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS – Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

Component (ECC) 6.0, the version that would remain the predominant transactional core in organizations for more than a decade, and whose mainstream support, originally intended for Initially set to end in 2025, it was later extended by SAP until 2027, with support options. extended until 2030 — a deadline that currently puts pressure on organizations to accelerate the migration to S/4HANA.

SAP HANA and in-memory computing as a technological disruption.

The second major architectural disruption at SAP occurred with the commercial launch of SAP. HANA (High-Performance Analytic Appliance), in 2010–2011. Conceived from research conducted by Hasso Plattner and Alexander Zeier at the Hasso Plattner Institute, the HANA introduced an in-memory, column-oriented relational database capable of eliminate the traditional distinction between transactional systems (OLTP) and analytical systems (OLAP), processing both types of payload on the same database residing in High-performance volatile memory.
This technological breakthrough wasn't limited to performance: by drastically reducing latency... By reading and aggregating data, HANA enabled real-time analytical scenarios directly. about transactional data — an indispensable technical prerequisite for subsequent applications of Embedded Artificial Intelligence for the automation of time-sensitive industrial processes. such as continuous monitoring of production lines.

SAP S/4HANA: simplification, Fiori and the new digital core

Launched in February 2015, SAP S/4HANA (Business Suite 4 SAP HANA) represented The reconstruction of the SAP transactional core specifically for the in-memory architecture of HANA, abandoning redundant data structures inherited from R/3 and ECC — such as Old financial and inventory aggregation tables — in favor of a data model. simplified, in which individual documents (line items) became the sole source of truth, with the totals being calculated dynamically.

In parallel, SAP introduced SAP Fiori as a user experience layer, based on... in responsive web applications (SAPUI5) and consumption of OData services, breaking with the traditional interface based on the SAP GUI. This interface paradigm shift required, therefore In turn, a new programming model capable of exposing data and business logic in a way...

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS – Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

standardized, REST-oriented services — a demand that would be met a few years later, through the ABAP RESTful Application Programming Model, the central theme of section 5 of this article. It is at this point of convergence — simplification of the data model, in-memory computing and service-oriented interfaces — which can be identified as the immediate origin of the transition from SAP is moving towards a microservices architecture, even while operating within the boundaries of A historically monolithic ERP system. The following section examines the theoretical foundations. which allow us to understand this transition in broader software engineering terms contemporary.

THEORETICAL FOUNDATION

This section systematizes three theoretical bodies indispensable to the proposed analysis: the architecture of Microservices, as a software engineering paradigm; Industry 4.0, as conceptual framework of contemporary industrial automation; and Artificial Intelligence applied to ERP systems, as an element of embedded intelligence in business processes.

Microservices architecture: principles and evolution

Microservices architecture can be defined as a development style of software in which an application is structured as a set of small services, independently deployable, organized around specific business capabilities, communicating through lightweight mechanisms — typically HTTP/REST APIs or asynchronous messaging. Fowler and Lewis (2014), in a seminal text on the subject, contrast This model contrasts with the traditional monolithic architecture, in which all application functionality is integrated. It resides in a single deployable process, sharing the same codebase, the same compilation cycle and the same process at runtime.

Newman (2021) systematizes the main advantages of this architectural style: Independent scalability of components according to the specific demands of each service; possibility of technological heterogeneity, allowing different services to be implemented in different languages and platforms according to their nature; isolation of failures, so that the unavailability of a service does not necessarily compromise the system as a whole; and continuous deployment, reducing risk. associated with large release cycles.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

On the other hand, the literature also acknowledges costs and risks associated with fragmentation.
excessive systems, including the complexity of distributed orchestration, the need for
robust observability mechanisms (logging, tracing, and distributed monitoring) and the
challenges of data consistency between services that, by definition, do not share a
single transactional basis. In the context of ERP systems — historically conceived under the
The principle of centralized transactional integrity—the adoption of microservices—cannot,
Therefore, it cannot be total or unrestricted, requiring a hybrid model that preserves a core.
A cohesive transactional system that simultaneously allows for modular extension through
decoupled services. It is precisely this hybrid model that SAP would later formalize under the concept...
of clean core, discussed in section 5.

Industry 4.0 and industrial process automation

The term Industry 4.0 (Industrie 4.0) was coined in Germany and formalized in the report of
recommendations prepared by Kagermann, Wahlster and Helbig (2013) for the federal government
German, within the framework of the strategic initiative of the same name. The concept describes the fourth
The industrial revolution as the integration of cyber-physical systems.
Industrial Internet of Things (IIoT) and cloud computing to manufacturing processes, from
in order to enable smart factories capable of self-configuration,
Self-monitoring and decentralized decision-making.
In this paradigm, ERP systems cease to operate as passive data repositories.
Consolidated batch transactions now require real-time integration with
sensors, programmable logic controllers (PLCs) and manufacturing execution systems
(Manufacturing Execution Systems – MES). This integration is typically made possible by
Event-Driven Architectures, in which state changes
in a production environment — such as a vibration sensor reading outside of operational limits.
— They send asynchronous messages that are consumed by multiple services, including modules.
maintenance, quality and production planning of the ERP itself.

Artificial Intelligence applied to ERP systems

According to Russell and Norvig (2020), Artificial Intelligence comprises the study of agents.
computational systems capable of perceiving their environment and acting in a way that maximizes probability.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS – Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

of success in relation to a goal, encompassing techniques such as machine learning.
supervised and unsupervised learning, reinforcement learning, language processing
Natural vision and computer vision. In the context of ERP systems, the application of these techniques
Over the past decade, it has migrated from isolated analytical scenarios — typically executed
in data warehouse environments separate from the transactional system — for scenarios of
embedded intelligence, in which predictive and prescriptive models
They operate directly on the flow of business processes.

In the specific case of manufacturing, three classes of AI applications stand out in the literature and
In industrial practice: (i) predictive maintenance, based on regression and classification models.
trained on time series of sensor data, capable of anticipating equipment failures beforehand.
that occur; (ii) automated quality control, supported by computer vision for
(iii) detection of defects in production lines; and (iii) optimization of planning and
production sequencing, employing combinatorial optimization algorithms and
Reinforcement learning. The technical feasibility of these three application classes, in a way
Integrated into the transactional core of the ERP, this is precisely the problem that convergence between
ABAP RAP and SAP BTP aim to solve this, as will be demonstrated in sections 5, 6, and 7.

METHODOLOGY

This article adopts a qualitative approach, of an exploratory and descriptive nature, supported by...
bibliographic and documentary research (GIL, 2019). The methodological choice is justified by
Nature of the object of study — the architectural evolution of a technological platform
proprietary technology and its convergence with software engineering paradigms—which requires the
Triangulation between consolidated academic literature on microservices, Industry 4.0 and
Artificial Intelligence, on the one hand, and official technical documentation from the manufacturer (SAP SE), on the other.
On the other hand, given the relative scarcity of peer-reviewed academic production specifically on
ABAP RAP and SAP BTP are technologies that have been recently adopted commercially.

The data collection procedure comprised three stages. In the first stage, it was...
reconstructed the historical timeline of SAP platforms (R/2, R/3, mySAP, NetWeaver,
ECC, HANA and S/4HANA), based on official technical documentation, corporate blogs of
SAP Community and specialized literature published by SAP Publishing.
PRESS. In the second stage, the theoretical foundations of microservices were systematized.
Industry 4.0 and AI applied to ERP, based on widely cited academic references in

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS – Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

area (FOWLER; LEWIS, 2014; NEWMAN, 2021; KAGERMANN; WAHLSTER; HELBIG, 2013; RUSSELL; NORVIG, 2020). In the third stage, the technical characterization of the ABAP RAP and SAP BTP, based on official documentation from the SAP Help Portal and SAP. Community, complemented by a reference work on the subject (HAAS; MATHEW, 2020). The analysis of the collected data was conducted using qualitative content analysis. seeking to identify, in each historical phase examined, the architectural elements that They anticipated or technically enabled the subsequent phase. Based on this analysis, it was A conceptual reference architecture was developed—but not implemented. empirically in this work—whose purpose is to illustrate, in a systematic way, the integration between ABAP Cloud microservices, SAP BTP AI services, and event buses, applied to a simulated industrial automation scenario. As a methodological limitation, It is recognized that the absence of empirical validation in a real-world production environment restricts the The conclusions can be reached at a conceptual and prescriptive level, not allowing for inferences. Quantitative indicators regarding performance, cost, or return on investment.

The Role of ABAP RAP in Modern SAP Architecture

While section 2 demonstrated that the evolution of SAP architectures has created, over decades, the technical prerequisites — in-memory computing, service-oriented interfaces and Simplifying the data model — for a transition towards microservices, this This section examines the specific technical instrument through which this transition occurs. It operates within the transactional core: ABAP RESTful Application Programming Model (RAP).

From monolithic programs to Clean Core

Historically, SAP system customization was based on direct modification or... extending repository objects within the transactional core itself, a practice that generated strong coupling between the specific business logic of each client and the standard source code provided by SAP. This model, sustainable during the R/3 and ECC era — when the Version updates occurred in multi-year cycles. became progressively incompatible with the continuous delivery model adopted by SAP starting from the S/4HANA Cloud, whose quarterly updates require custom extensions

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS - Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

They should be decoupled from the core, otherwise recurring conflicts may occur in each upgrade cycle. In response to this dilemma, SAP formalized the principle of a clean core, according to whereby the default system should remain as close as possible to its original configuration, being all client-specific extensions implemented through released APIs APIs), in-app extensibility based on predefined extension points, or development of side-by-side applications hosted on SAP BTP. ABAP RAP constitutes the central technical instrument for implementing this principle within the environment itself. ABAP, by formalizing a programming model structured in well-defined layers, Business Objects-oriented and compatible with both in-extensibility app como com desenvolvimento lado lado na BTP.

RAP: concepts, layers and development cycle

ABAP RAP was introduced by SAP in 2018 and became available for production use in on-premise environments starting from SAP S/4HANA version 1909, and since then, made available simultaneously in the SAP BTP ABAP environment (SAP BTP ABAP). Environment) and in SAP S/4HANA Cloud, in both its public and private editions. Conceptually, RAP is described by official SAP documentation as an evolutionary successor. from the ABAP Programming Model for SAP Fiori, substantially simplifying the process of Creating OData services by eliminating the need for SAP GUI transactions and generation. Service manual via the SEGW transaction, programming model features previous.

From a technical point of view, RAP is structured in four main layers. The layer of Data modeling is built upon Core Data Services (CDS Views), which define entities, associations, and semantic annotations directly at the database level. leveraging SAP HANA's in-memory processing capabilities to delegate perform aggregation and join operations directly on the database, instead of processing them in the database itself. application layer — a principle known as code-to-data (bringing the code to the data, and (not the data up to the code).

The behavior definition layer, written in the Behavior Definition Language. (BDL) specifies the operations allowed on each business object — creation, reading, Update, delete, and custom actions. —, as well as validations, determinations Automatic value and authorization rules. The behavior implementation layer. (behavior implementation), written in conventional object-oriented ABAP, contains the logic

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS – Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

of the business itself. Finally, the service exposure layer.
Publishes the business object as an OData V2 or V4 service, consumable by applications.
SAP Fiori, as well as external systems, including services hosted on SAP BTP.
RAP offers two types of implementation: the managed implementation type, in
Much of the persistence logic is generated automatically by the framework from
the behavior definition, recommended for greenfield developments (starting from scratch); and the
unmanaged implementation, in which the developer implements
Manually implement the persistence logic; recommended for brownfield integration scenarios.
with legacy business logic already existing in the system. This duality allows organizations
With decades of accumulated customization in ECC or on-premise S/4HANA, adopt RAP.
incrementally, without the need to completely rewrite your processes.
business.

RAP as an enabler of microservices within S/4HANA

It is possible to characterize each RAP business object as a cohesive functional unit, endowed
a well-defined interface contract (the exposed OData service), logic encapsulation
internal (the implementation of behavior) and relative independence from other objects of
business, communicating with them preferably through compositions and associations.
formally declared, rather than direct coupling at the code level. This characterization
It brings RAP closer, on a reduced scale and within the limits of a single ABAP system, to
structural principles that Newman (2021) attributes to microservices — even though it is a matter of,
Technically, it's about internal modularization within the monolith, and not about physically providing services.
distributed across independent processes.
The distinction is relevant: within S/4HANA itself, RAP enables what the literature of
Software architecture refers to a modular monolith — a system that
It preserves the unity of deployment and centralized transactional integrity, but organizes its
Internal code in strongly cohesive and loosely coupled modules, analogous to microservices in their
logical structure, though not in their execution topology. It is
It is precisely this logical modularity that allows RAP, when combined with SAP, to work effectively.
BTP, extending to a physically distributed microservices architecture: each object
RAP business data can be consumed, through its standardized OData API, by services.
Hosted on BTP — implemented in Java, Node.js, Python, or CAP (Cloud Application).

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS - Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

Programming Model) — which assume specific functions, such as orchestrating flows of industrial automation or implementation of Artificial Intelligence models, without the need for Direct access to the S/4HANA database.

Therefore, RAP does not replace microservices architecture, but complements it from a basic perspective. from within the transactional core: it ensures that the core remains clean, cohesive and Updatable, while simultaneously exposing, in a standardized and governed manner, the points of extension required for external services — hosted on BTP or other cloud platforms — can, together with the ERP, form a more distributed architecture broad, focused on the automation of industrial processes mediated by Artificial Intelligence, topic developed in the following section.

SAP BTP as a Platform for Integration, Data, and Intelligence ARTIFICIAL

If ABAP RAP enables modularity and governed extensibility within the core In the digital realm, it is the SAP Business Technology Platform (BTP) that provides the execution environment. distributed, in which this modularity expands to a microservices architecture. properly speaking, incorporating integration capabilities, data management and Intelligence. Artificial elements that go beyond the technical scope of the traditional ABAP environment.

From SAP Cloud Platform to Business Technology Platform

BTP's trajectory dates back to the launch of the SAP HANA Cloud Platform in 2012. later renamed SAP Cloud Platform (SCP) in 2017, and finally repositioned as SAP Business Technology Platform in 2021, consolidating under a single brand. commercial four pillars previously offered in a relatively segmented way: bank of Data and data management, analytics, application integration and data (Integration Suite) and application extensibility (Extension Suite), to which is More recently, they have added Artificial Intelligence services gathered under the SAP brand. Business AI.

This consolidation reflects a broader strategic move by SAP, guided by The concept of RISE with SAP (2021) and, subsequently, GROW with SAP, according to which the S/4HANA is now being marketed and deployed as a standardized digital core, around around which revolve extension, integration, and intelligence services hosted at BTP —

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS - Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

replicating, on an organizational and commercial scale, the same core architectural principle cohesive plus distributed periphery already identified, on a technical scale, in the relationship between RAP and microservices.

Artificial Intelligence Services: SAP AI Core, AI Foundation, and Business AI

Within the BTP framework, SAP AI Core is the service responsible for execution and implementation. Lifecycle management of machine learning models in an environment Kubernetes, offering training, versioning, and deployment capabilities. Model serving is provided through REST APIs. This service is complemented by SAP AI. Launchpad, a centralized interface for monitoring and governing AI scenarios, and by a broader set called SAP AI Foundation, which brings together access to models of third-party large language models — such as those provided through partnerships with specialized providers — and orchestration tools of Generative artificial intelligence flows.
Under the SAP Business AI brand, the company commercially positions a suite of AI capabilities already embedded in specific business processes of S/4HANA and others product line solutions, such as predictive recommendations in demand planning, Anomaly detection in financial transactions and conversational interaction assistants. in natural language (SAP Joule). From an architectural point of view, the most relevant aspect For industrial automation, the possibility lies in using models hosted on SAP AI Core. to consume operational data exposed by OData services built with ABAP RAP, processing this data through machine learning algorithms, and to provide feedback to the transactional core with results — such as probabilities of failure of equipment or quality classifications — again through standardized operations from the RAP business object itself, closing a feedback loop between data Operational, AI inference, and transactional action.

Integration Suite, Event Mesh, and event-driven architecture.

Communication between the S/4HANA digital core, the AI services hosted on BTP and External industrial automation systems (MES, SCADA, process historians) depend robust integration mechanisms provided by the SAP Integration Suite. This set of

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS - Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

Services include Cloud Integration, responsible for orchestrating point-to-point integration flows.
Point-to-point and mediated by standards such as REST, SOAP, and OData; API Management, responsible
for the governance, versioning, and security of externally exposed APIs; and the SAP Event
Mesh, a messaging bus based on the publish-subscribe paradigm.
(publish/subscribe), which enables event-driven architectures at scale.
It is the Event Mesh that allows an industrial sensor — or, more precisely, the system to
automation that monitors it — publish an anomaly event without needing to
prior knowledge about which services will consume this information, whether they are services
RAP within S/4HANA, CAP applications hosted on BTP, or running AI models.
in SAP AI Core. This temporal and topological decoupling between producers and consumers.
Event automation is a necessary architectural condition for process automation.
Industrial systems operate in real time, without creating rigid synchronous dependencies between the floor.
factory and the transactional core of the ERP system — a risk that would compromise both performance
Regarding the resilience of the system as a whole.

CONVERGENCE: MICROSERVICES, RAP, BTP AND AI IN AUTOMATION OF INDUSTRIAL PROCESSES

The previous sections have demonstrated, in isolation, how the historical evolution of
SAP architectures (section 2), the theoretical foundations of microservices, Industry 4.0 and AI.
(section 3), ABAP RAP (section 5) and SAP BTP (section 6) are components of the same system.
architectural puzzle. This section proposes a reference architecture that articulates these
pieces revolving around an applied industrial automation scenario, followed by a discussion about
its implementation challenges.

Proposed reference architecture

The proposed reference architecture is organized into four logical layers. The edge layer
Industrial (edge layer) includes sensors, programmable logic controllers, and systems.
MES/SCADA systems are responsible for the primary collection of process data — temperature, vibration,
Pressure, line velocity — typically aggregated by an edge gateway.
before its transmission to the cloud, in order to reduce traffic volume and latency.
The integration and events layer, supported by SAP Integration Suite and SAP Event.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS – Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

Mesh receives data from the edge layer and distributes it asynchronously to the...
Interested consumers. The intelligence layer, hosted on SAP AI Core, consumes the
relevant events, applies previously trained machine learning models — by
For example, an industrial bearing failure risk classification model trained on
historical vibration series — and publishes its results back to the event bus or
They are made available through a REST API.
Finally, the transactional layer, consisting of the S/4HANA digital core, receives the results.
AI inference through RAP business objects specifically designed for this purpose.
end — as an extension of the Maintenance Order business object,
capable of automatically creating a preventative order when the probability of failure is high.
The value calculated by the AI model exceeds a certain threshold. This entire flow operates without the...
AI inference logic resides within the ABAP core, and without the transactional logic of
The business resides outside of it — simultaneously preserving the principle of a clean core and the
Distributed modularity is a characteristic of microservices.
Figure 1 — Conceptual flow of the reference architecture: Industrial Edge ÿ SAP Event Mesh
ÿ SAP AI Core (inference) ÿ RAP Business Object (S/4HANA) ÿ Maintenance Order
Preventive. Source: prepared by the author (2026).

Illustrative case: predictive maintenance in a production line

To illustrate the proposed architecture, consider a hypothetical predictive maintenance scenario.
Vibration sensors installed on critical electric motors in an industrial bottling line.
They transmit readings every second to an edge gateway, which aggregates these readings into windows.
Sixty-second shots are recorded and published as events on SAP Event Mesh. A hosted service.
In SAP AI Core, previously trained with historical data of labeled failures, it consumes these
The system calculates, for each window, the probability of imminent equipment failure in the next few events.
seventy-two hours.
When this probability exceeds a predefined threshold — for example, seventy-five
The AI service publishes an alert event, which is consumed by a CAP percent application.
Hosted on SAP BTP, responsible for enriching the alert with additional contextual data.
(history of previous equipment maintenance, availability of spare parts)
(in stock) obtained through calls to RAP business objects exposed by S/4HANA.
Once the relevance of the alert is confirmed, the CAP application invokes an action of the object.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

RAP maintenance order business, automatically creating a preventive order,
assigned to the responsible maintenance team, with priority calculated based on...
criticality of the equipment and the estimated probability of failure.
This complete cycle — from sensor reading to creating the maintenance order in the ERP system — illustrates
In concrete terms, the central thesis of this article is: the automation of industrial processes oriented towards...
The role of Artificial Intelligence in the contemporary SAP ecosystem does not stem from a single...
not just isolated technology, but the orchestration between microservices for integration and intelligence.
hosted at BTP, and a modular, governed transactional core, enabled by ABAP.
RAP.

Challenges: governance, legacy integration, and security.

Implementing the proposed architecture is not without significant challenges. Firstly,
In this context, API governance becomes a critical element: as the number of objects of
Business RAP exposed as released APIs grow, organizations need processes
Formal API lifecycle management tools — versioning, controlled depreciation and
documentation — otherwise, the same problem will be reproduced on a distributed scale.
The tight coupling that microservices architecture seeks to avoid.
Secondly, integration with legacy industrial automation systems — many of which
which operate using proprietary protocols or industry standards that predate the popularization of
REST APIs, such as OPC-UA, Modbus, or Profibus, often require additional layers.
protocol conversion at the edge, before the data can be incorporated into the stream of
events based on web standards described in section 7.1. Thirdly, the surface of
exhibition created through the integration of the factory floor, public cloud, and transactional core.
This significantly increases information security risks, requiring rigorous controls.
authentication, authorization, and network segmentation between edge layers, integration,
Intelligence and transaction processing — especially relevant in industrial sectors subject to regulation.
critical infrastructure.

DISCUSSION

The findings systematized in this article allow us to revisit the initial hypothesis that the adoption of
Microservices in the SAP ecosystem represent a cumulative evolutionary process, not a disruption.
abrupt. The historical analysis in section 2 shows that each architectural transition — from R/2 to

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS - Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

R/3, from R/3 to NetWeaver, from ECC to HANA, from HANA to S/4HANA — replied to
specific technical limitations of the previous generation, while preserving
fundamental structural elements, such as the centrality of the ABAP language and the principle of
data integration on a single database. This historical continuity is technically confirmed in
RAP's own concept, which official SAP documentation describes as its successor
evolutionary — and not a disruptive replacement — of the ABAP Programming Model for SAP Fiori.
From a theoretical point of view, this article contributes to the literature by proposing a distinction.
conceptual framework between internal logical modularity and the transactional core — enabled by RAP.
in the form of a modular monolith — and physical distribution of services in an architecture of
microservices proper — made possible by the combination of RAP and SAP BTP.
This distinction proves relevant because the current literature on microservices (NEWMAN,
2021; FOWLER; LEWIS, 2014), conceived mostly from native systems of
Cloud computing, built from its inception under this paradigm, does not adequately address the case of
large-scale legacy ERP systems, whose transition to microservices is essential.
from a cohesive transactional core, at the risk of compromising data integrity guarantees.
critical for financial and manufacturing operations.
From the perspective of industrial automation, the results suggest that the main contribution
The convergence between RAP, BTP, and AI does not lie in raw processing speed — it already
largely ensured by SAP HANA in-memory computing since 2011. —,
but in the governance of the integration between artificial intelligence and transactional processes of
business. That is, the architecture proposed in section 7 does not make predictive maintenance technically feasible.
possible for the first time — machine learning models for this purpose are already available.
widely used in ERP-independent industrial IoT platforms—but it makes
It is possible that the results of this prediction will translate, in an auditable, governed and
integrated into transactional actions within the business management system, without requiring
Point-by-point and non-standardized integration between heterogeneous systems.
As a limitation of this study, its eminently theoretical-conceptual nature is reiterated: a
The proposed reference architecture has not been subjected to empirical validation in a real-world environment.
productive, and peer-reviewed academic literature on ABAP RAP and SAP BTP
remains incipient, which has required substantial reliance on official technical documentation of
Manufacturer — legitimate source, but subject to commercial bias inherent to the very nature of SAP.
as a stakeholder in promoting its proprietary technologies. Future studies of
empirical in nature, including case studies in real-world industrial settings and assessments.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS – Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

Quantitative performance and cost data are needed to corroborate or qualify the...
propositions presented here.

FINAL CONSIDERATIONS

This article analyzed, from a historical and evolutionary perspective, the trajectory of SAP architectures.
From R/2 to S/4HANA, demonstrating how each technological generation created the conditions.
cumulative techniques for the transition, observable today, towards architectures of
Microservices driven by Artificial Intelligence. It was found that ABAP RAP operates
as an instrument for governed modularization of the transactional core, materializing the
The principle of clean core and enabling extensibility without compromising integrity and
Continuous updatable standard system. It was also found that SAP
BTP provides the distributed execution environment — integration, data, and Artificial Intelligence.
— in which this modularity expands into a physically microservices architecture.
distributed, capable of supporting real-time industrial process automation scenarios.
The reference architecture proposed in section 7, illustrated by means of a hypothetical case of
Predictive maintenance demonstrated how these elements—industrial edge layer,
event-driven integration, Artificial Intelligence inference, and transactional core
Modular—they can be articulated in a coherent and governed way. It is concluded that the main...
The contribution of this technological convergence does not reside solely in any of its...
components, but in the ability to integrate them in a standardized, auditable, and resilient way.
Continuous update cycles — a capability that distinguishes the architectural proposal here.
discussed the historically common, ad hoc, and ungoverned integrations in environments of
industrial automation.
As a future research agenda, conducting empirical case studies is recommended in
industrial organizations that have effectively implemented architectures similar to
proposal, with a view to quantitatively measuring performance indicators, resilience
Operational performance and return on investment. Further investigation is also recommended.
of the data governance and information security challenges inherent in expanding
Integration surface between the factory floor, public cloud, and critical transactional systems.
This topic, given its growing relevance, goes beyond the scope of this article and requires further consideration.
specific in subsequent research.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google

Year VII, v.1 2026 | Submission: 08/25/2026 | Accepted: 08/28/2026 | Publication: 08/31/2026

REFERENCES

Brazilian Association of Technical Standards. NBR 6023: Information and documentation: References: Preparation. Rio de Janeiro: ABNT, 2018.

Brazilian Association of Technical Standards. NBR 14724: Information and documentation: academic works: presentation. Rio de Janeiro: ABNT, 2011.

FOWLER, Martin; LEWIS, James. Microservices: a definition of this new architectural term. martinfowler.com, 2014. 25 sea. Available at in: https://martinfowler.com/articles/microservices.html. Accessed on: July 2, 2026.

GIL, Antonio Carlos. How to prepare research projects. 6th ed. São Paulo: Atlas, 2019.

HAAS, Stefan; MATHEW, Bince. ABAP RESTful Programming Model. Boston: Rheinwerk/SAP PRESS, 2020.

KAGERMANN, Henning; WAHLSTER, Wolfgang; HELBIG, Johannes (org.). Recommendations for implementing the strategic initiative INDUSTRIE 4.0: final report of the Industrie 4.0 working group. Frankfurt am Main: Forschungsunion; Acatech, 2013.

NEWMAN, Sam. Building Microservices: designing fine-grained systems. 2nd ed. Sebastopol: O'Reilly Media, 2021.

PLATTNER, Hasso; ZEIER, Alexander. In-Memory Data Management: an inflection point for enterprise applications. 2nd ed. Berlin: Springer, 2011.

RUSSELL, Stuart J.; NORVIG, Peter. Artificial Intelligence: a modern approach. 4th ed. Hoboken: Pearson, 2020.

SAP SE. ABAP RESTful Application Programming Model (RAP). SAP Community, 2024. Available at: https://pages.community.sap.com/topics/abap/rap. Accessed on: August 5, 2026.

SAP SE. Getting started with the ABAP RESTful Application Programming Model (RAP). SAP Community Technology Blog, Jan. 2024. Available at: https://community.sap.com/t5/technology-blog-posts-by-sap/getting-started-with-the-abaprestful-application-programming-model-rap/ba-p/13420829. Accessed on: Aug. 12, 2026.

SAP SE. SAP Business Technology Platform. Available at in: https://www.sap.com/products/technology-platform.html. Accessed on: August 12, 2026.

SAP SE. SAP S/4HANA. Available at: https://www.sap.com/products/erp/s4hana.html. Accessed on: August 17, 2026.

DEUTSCHE SAP-ANWENDERGRUPPE (DSAG). Das ABAP RESTful Application Programming Model (RAP). ABAP-Leitfaden, 2024. Available at: https://1dsag.github.io/ABAP-Leitfaden/abap/restful_abap/. Accessed on: August 19, 2026.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.