RCMOS – Revista Científica Multidisciplinar O Saber.
ISSN: 2675-9128. São Paulo-SP.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

Arquitetura de microsserviços e inteligência artificial aplicada ao sap s/4hana: o papel do abap rap e da sap btp na automação de processos industriais

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

Arquitectura de microservicios e inteligencia artificial aplicadas a SAP S/4HANA: el papel de ABAP RAP y SAP BTP en la automatización de procesos industriales

François Pierre Fagundes Francelino

RESUMO

Este artigo tem como objetivo analisar, sob uma perspectiva histórica e evolutiva, a transição das arquiteturas monolíticas do sistema SAP para um paradigma orientado a microsserviços, evidenciando o papel do ABAP RESTful Application Programming Model (RAP) e da SAP Business Technology Platform (BTP) na automação de processos industriais mediada por Inteligência Artificial (IA). Parte-se da trajetória tecnológica que se inicia com o SAP R/2 e o SAP R/3, passa pela consolidação do SAP ECC e culmina na ruptura promovida pelo SAP S/4HANA e pela computação in-memory baseada no SAP HANA. Discutem-se os fundamentos teóricos da arquitetura de microsserviços, da Indústria 4.0 e da IA aplicada a sistemas de gestão empresarial (ERP), estabelecendo-se uma ponte conceitual entre esses domínios e as capacidades técnicas do RAP e da BTP. Por meio de pesquisa bibliográfica de natureza qualitativa e exploratória, com apoio documental em fontes técnicas oficiais da SAP e na literatura acadêmica correlata, propõe-se uma arquitetura de referência que integra microsserviços ABAP Cloud, serviços de IA embarcados (SAP AI Core e AI Foundation) e barramentos de eventos (SAP Event Mesh) para viabilizar cenários de automação industrial, com destaque para manutenção preditiva e controle de qualidade em tempo real. Os resultados indicam que o RAP, ao operacionalizar o princípio do clean core, funciona como elemento habilitador de microsserviços transacionais dentro do núcleo digital, enquanto a BTP amplia esse núcleo por meio de serviços desacoplados de dados, integração e inteligência artificial. Conclui-se que a convergência entre esses componentes representa não apenas uma evolução tecnológica incremental, mas uma mudança paradigmática na forma como sistemas ERP suportam a automação de processos produtivos, exigindo, por outro lado, atenção redobrada à governança, à segurança da informação e à gestão da dívida técnica decorrente de décadas de customização.

Palavras-chave: SAP S/4HANA. ABAP RAP. SAP BTP. Microsserviços. Inteligência Artificial. Automação Industrial.

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

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

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.

INTRODUÇÃO

A automação de processos industriais deixou de ser, nas últimas duas décadas, um domínio restrito a sistemas de controle supervisório e aquisição de dados (SCADA) e passou a integrar-se, de forma cada vez mais estreita, aos sistemas corporativos de gestão empresarial (Enterprise Resource Planning – ERP). Nesse contexto, a SAP SE, fornecedora líder mundial de soluções ERP, consolidou ao longo de mais de cinco décadas uma trajetória tecnológica que evoluiu de arquiteturas monolíticas centralizadas para plataformas distribuídas, orientadas a serviços e, mais recentemente, a microsserviços integrados por meio da nuvem.

O lançamento do SAP S/4HANA, em 2015, e sua base tecnológica assentada sobre o banco de dados in-memory SAP HANA, representou um ponto de inflexão que não se limitou a ganhos de desempenho computacional. Tal transição impôs a necessidade de repensar o próprio modelo de desenvolvimento de aplicações corporativas, culminando na proposição do ABAP RESTful Application Programming Model (RAP) e na consolidação da SAP Business Technology Platform (BTP) como ambiente de extensão, integração e inteligência artificial. Esses dois elementos — RAP e BTP — constituem, em conjunto, o alicerce técnico sobre o qual se assenta a proposta de um núcleo digital limpo (clean core), capaz de sustentar arquiteturas de microsserviços sem comprometer a estabilidade e a capacidade de atualização contínua do sistema central.

O problema de pesquisa que orienta este artigo pode ser assim formulado: de que maneira a evolução histórica das arquiteturas SAP, da era mainframe até o S/4HANA, explica e fundamenta o papel atualmente exercido pelo ABAP RAP e pela SAP BTP na viabilização de arquiteturas de microsserviços orientadas por Inteligência Artificial para a automação de processos industriais? Parte-se da hipótese de que a adoção de microsserviços no ecossistema SAP não constitui uma ruptura abrupta, mas sim o resultado de um processo evolutivo cumulativo, no qual cada geração tecnológica — R/2, R/3, mySAP, NetWeaver, ECC, HANA

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

e S/4HANA — preparou as condições técnicas e organizacionais para a geração subsequente.

O objetivo geral deste trabalho é analisar, de forma técnica e científica, a trajetória histórica das arquiteturas SAP e sua convergência com os princípios de microsserviços, examinando especificamente como o ABAP RAP e a SAP BTP operacionalizam essa convergência para viabilizar cenários de automação industrial apoiados em Inteligência Artificial. Como objetivos específicos, busca-se: (i) reconstituir a evolução histórica das plataformas SAP, relacionando-a às transformações mais amplas da engenharia de software; (ii) sistematizar os fundamentos teóricos da arquitetura de microsserviços, da Indústria 4.0 e da IA aplicada a sistemas ERP; (iii) descrever o funcionamento técnico do ABAP RAP e sua relação com o conceito de clean core; (iv) caracterizar os serviços de dados, integração e IA disponibilizados pela SAP BTP; e (v) propor uma arquitetura de referência que integre esses elementos em um cenário aplicado de automação de processos industriais.

Do ponto de vista metodológico, adota-se uma abordagem qualitativa, de caráter exploratório e descritivo, apoiada em pesquisa bibliográfica e documental, cujos procedimentos são detalhados na seção 4. A relevância do tema justifica-se pela crescente demanda, por parte de organizações industriais, por sistemas capazes de integrar dados de chão de fábrica, algoritmos de aprendizado de máquina e processos transacionais de negócio em tempo real — demanda esta que não pode ser plenamente atendida por arquiteturas puramente monolíticas, dada sua baixa capacidade de escalonamento independente, de implantação incremental e de adoção de linguagens e paradigmas de programação heterogêneos.

O artigo está estruturado em nove seções, incluindo esta introdução. A seção 2 apresenta a evolução histórica das arquiteturas SAP. A seção 3 discute o referencial teórico sobre microsserviços, Indústria 4.0 e IA aplicada a ERP. A seção 4 descreve a metodologia. As seções 5 e 6 detalham, respectivamente, o ABAP RAP e a SAP BTP. A seção 7 apresenta a arquitetura de referência proposta e um caso ilustrativo. A seção 8 discute os resultados à luz da literatura, e a seção 9 apresenta as considerações finais.

EVOLUÇÃO HISTÓRICA DAS ARQUITETURAS SAP: DE R/2 AO S/4HANA

Compreender o papel atual do ABAP RAP e da SAP BTP exige, necessariamente, uma reconstituição da trajetória histórica das arquiteturas de software da SAP, uma vez que cada geração tecnológica incorporou respostas específicas às limitações da geração anterior, ao mesmo tempo em que preservou princípios estruturais que permanecem vigentes até o presente,

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

como a modelagem de dados centrada em processos de negócio e a linguagem proprietária ABAP (Advanced Business Application Programming).

SAP R/2 e SAP R/3: a era dos sistemas monolíticos

Fundada em 1972 na Alemanha, a SAP (Systemanalyse und Programmentwicklung) lançou, ao final da década de 1970, o sistema R/2, uma solução integrada de gestão empresarial concebida para operar em ambientes de mainframe. O R/2 já incorporava um princípio que se manteria como marca distintiva da empresa: a integração de módulos funcionais — finanças, materiais, produção — sobre uma base de dados única e compartilhada, eliminando a duplicidade de informações típica dos sistemas departamentais isolados então predominantes.

Em 1992, a SAP lançou o R/3, migrando de uma arquitetura centrada em mainframe para o modelo cliente-servidor de três camadas (apresentação, aplicação e banco de dados), então emergente na indústria de software. Essa transição representou a primeira grande ruptura arquitetural da empresa, permitindo a distribuição da carga de processamento entre diferentes camadas de hardware e abrindo caminho para a interoperabilidade com sistemas operacionais e bancos de dados de terceiros. O R/3 consolidou, ainda, o uso da linguagem ABAP como principal instrumento de customização e extensão do sistema padrão, prática que se tornaria, décadas mais tarde, um dos principais desafios à modernização arquitetural — fenômeno conhecido como dívida técnica de customização.

mySAP.com, SAP NetWeaver e a abertura para a integração

No início dos anos 2000, com a popularização da internet comercial, a SAP lançou a iniciativa mySAP.com, orientando seu portfólio para cenários de comércio eletrônico e integração entre empresas (business-to-business). Essa fase culminou, em 2003, no lançamento do SAP NetWeaver, uma plataforma de integração baseada em padrões abertos — como XML e serviços web (Web Services) — que passou a atuar como camada intermediária entre o núcleo transacional R/3 (posteriormente ECC) e sistemas externos.

O NetWeaver introduziu componentes que se tornariam precursores conceituais da atual SAP BTP, como o Process Integration (PI), voltado à orquestração de mensagens entre sistemas heterogêneos, e o Enterprise Portal, responsável por consolidar interfaces de usuário de diferentes aplicações em um único ponto de acesso. Em 2006, a SAP lançou o SAP ERP Central

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

Component (ECC) 6.0, versão que permaneceria como núcleo transacional predominante nas organizações por mais de uma década, e cujo suporte mainstream, originalmente previsto para se encerrar em 2025, foi posteriormente estendido pela SAP até 2027, com opções de suporte estendido até 2030 — prazo que hoje pressiona as organizações a acelerarem a migração para o S/4HANA.

SAP HANA e a computação in-memory como ruptura tecnológica

A segunda grande ruptura arquitetural da SAP ocorreu com o lançamento comercial do SAP HANA (High-Performance Analytic Appliance), em 2010–2011. Idealizado a partir de pesquisas conduzidas por Hasso Plattner e Alexander Zeier no Hasso Plattner Institute, o HANA introduziu um banco de dados relacional in-memory, orientado a colunas, capaz de eliminar a distinção tradicional entre sistemas transacionais (OLTP) e sistemas analíticos (OLAP), processando ambos os tipos de carga sobre a mesma base de dados residente em memória volátil de alto desempenho.

Essa ruptura tecnológica não se limitou ao desempenho: ao reduzir drasticamente a latência de leitura e agregação de dados, o HANA viabilizou cenários analíticos em tempo real diretamente sobre dados transacionais — pré-requisito técnico indispensável para aplicações posteriores de Inteligência Artificial embarcada e para a automação de processos industriais sensíveis a tempo, como o monitoramento contínuo de linhas de produção.

SAP S/4HANA: simplificação, Fiori e o novo núcleo digital

Lançado em fevereiro de 2015, o SAP S/4HANA (Business Suite 4 SAP HANA) representou a reconstrução do núcleo transacional da SAP especificamente para a arquitetura in-memory do HANA, abandonando estruturas de dados redundantes herdadas do R/3 e do ECC — como as antigas tabelas de agregação financeira e de estoque — em favor de um modelo de dados simplificado, no qual documentos individuais (line items) tornaram-se a única fonte de verdade, sendo os totais calculados dinamicamente.

Paralelamente, a SAP introduziu o SAP Fiori como camada de experiência do usuário, baseada em aplicações web responsivas (SAPUI5) e consumo de serviços OData, rompendo com a interface tradicional baseada no SAP GUI. Essa mudança de paradigma de interface exigiu, por sua vez, um novo modelo de programação capaz de expor dados e lógica de negócio de forma

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

padronizada, orientada a serviços REST — demanda que seria respondida, poucos anos depois, pelo ABAP RESTful Application Programming Model, tema central da seção 5 deste artigo.
É nesse ponto de convergência — simplificação do modelo de dados, computação inmemory e interfaces orientadas a serviços — que se pode identificar a origem imediata da transição da SAP em direção a uma arquitetura de microsserviços, ainda que operando dentro dos limites de um sistema ERP historicamente monolítico. A seção seguinte examina os fundamentos teóricos que permitem compreender essa transição em termos mais amplos da engenharia de software contemporânea.

FUNDAMENTAÇÃO TEÓRICA

Esta seção sistematiza três corpos teóricos indispensáveis à análise proposta: a arquitetura de microsserviços, enquanto paradigma de engenharia de software; a Indústria 4.0, enquanto marco conceitual da automação industrial contemporânea; e a Inteligência Artificial aplicada a sistemas ERP, enquanto elemento de inteligência embarcada nos processos de negócio.

Arquitetura de microsserviços: princípios e evolução

A arquitetura de microsserviços pode ser definida como um estilo de desenvolvimento de software no qual uma aplicação é estruturada como um conjunto de serviços pequenos, independentemente implantáveis, organizados em torno de capacidades de negócio específicas, comunicando-se por meio de mecanismos leves — tipicamente APIs HTTP/REST ou mensageria assíncrona. Fowler e Lewis (2014), em texto seminal sobre o tema, contrapõem esse modelo à arquitetura monolítica tradicional, na qual toda a funcionalidade da aplicação reside em um único processo implantável, compartilhando a mesma base de código, o mesmo ciclo de compilação e o mesmo processo em tempo de execução.
Newman (2021) sistematiza as principais vantagens desse estilo arquitetural:
escalabilidade independente de componentes conforme a demanda específica de cada serviço; possibilidade de heterogeneidade tecnológica, permitindo que diferentes serviços sejam implementados em linguagens e plataformas distintas conforme sua natureza; isolamento de falhas, de modo que a indisponibilidade de um serviço não necessariamente comprometa o sistema como um todo; e implantação contínua (continuous deployment), reduzindo o risco associado a grandes ciclos de liberação de versões (releases).

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

Por outro lado, a literatura também reconhece custos e riscos associados à fragmentação excessiva de sistemas, entre eles a complexidade de orquestração distribuída, a necessidade de mecanismos robustos de observabilidade (logging, tracing e monitoramento distribuído) e os desafios de consistência de dados entre serviços que, por definição, não compartilham uma única base transacional. No contexto de sistemas ERP — historicamente concebidos sob o princípio da integridade transacional centralizada —, a adoção de microsserviços não pode, portanto, ser total ou irrestrita, sendo necessário um modelo híbrido que preserve um núcleo transacional coeso e, simultaneamente, permita a extensão modular por meio de serviços desacoplados. É exatamente esse modelo híbrido que a SAP viria a formalizar sob o conceito de clean core, discutido na seção 5.

Indústria 4.0 e automação de processos industriais

O termo Indústria 4.0 (Industrie 4.0) foi cunhado na Alemanha e formalizado no relatório de recomendações elaborado por Kagermann, Wahlster e Helbig (2013) para o governo federal alemão, no âmbito da iniciativa estratégica de mesmo nome. O conceito descreve a quarta revolução industrial como a integração de sistemas ciber-físicos (cyberphysical systems), Internet das Coisas Industrial (IIoT) e computação em nuvem aos processos de manufatura, de modo a viabilizar fábricas inteligentes (smart factories) capazes de autoconfiguração, automonitoramento e tomada de decisão descentralizada.

Nesse paradigma, sistemas ERP deixam de operar como repositórios passivos de dados transacionais consolidados em lote (batch) e passam a demandar integração em tempo real com sensores, controladores lógicos programáveis (CLPs) e sistemas de execução da manufatura (Manufacturing Execution Systems – MES). Essa integração é tipicamente viabilizada por arquiteturas orientadas a eventos (Event-Driven Architecture), nas quais alterações de estado no ambiente produtivo — como a leitura de um sensor de vibração fora dos limites operacionais — disparam mensagens assíncronas consumidas por múltiplos serviços, entre os quais módulos de manutenção, qualidade e planejamento de produção do próprio ERP.

Inteligência Artificial aplicada a sistemas ERP

Segundo Russell e Norvig (2020), a Inteligência Artificial compreende o estudo de agentes computacionais capazes de perceber seu ambiente e agir de modo a maximizar a probabilidade

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

RCMOS – Revista Científica Multidisciplinar O Saber.
ISSN: 2675-9128. São Paulo-SP.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

de sucesso em relação a um objetivo, abrangendo técnicas como aprendizado de máquina supervisionado e não supervisionado, aprendizado por reforço, processamento de linguagem natural e visão computacional. No contexto de sistemas ERP, a aplicação dessas técnicas migrou, ao longo da última década, de cenários analíticos isolados — tipicamente executados em ambientes de data warehouse apartados do sistema transacional — para cenários de inteligência embarcada (embedded intelligence), na qual modelos preditivos e prescritivos operam diretamente sobre o fluxo de processos de negócio.

No caso específico da manufatura, três classes de aplicação de IA destacam-se na literatura e na prática industrial: (i) manutenção preditiva, baseada em modelos de regressão e classificação treinados sobre séries temporais de sensores, capazes de antecipar falhas de equipamentos antes que ocorram; (ii) controle de qualidade automatizado, apoiado em visão computacional para detecção de defeitos em linhas de produção; e (iii) otimização de planejamento e sequenciamento de produção, empregando algoritmos de otimização combinatória e aprendizado por reforço. A viabilização técnica dessas três classes de aplicação, de forma integrada ao núcleo transacional do ERP, é precisamente o problema que a convergência entre ABAP RAP e SAP BTP busca resolver, como se demonstrará nas seções 5, 6 e 7.

METODOLOGIA

Este artigo adota abordagem qualitativa, de natureza exploratória e descritiva, apoiada em pesquisa bibliográfica e documental (GIL, 2019). A opção metodológica justifica-se pela natureza do objeto de estudo — a evolução arquitetural de uma plataforma tecnológica proprietária e sua convergência com paradigmas de engenharia de software —, que exige a triangulação entre literatura acadêmica consolidada sobre microsserviços, Indústria 4.0 e Inteligência Artificial, de um lado, e documentação técnica oficial do fabricante (SAP SE), de outro, dada a relativa escassez de produção acadêmica revisada por pares especificamente sobre o ABAP RAP e a SAP BTP, tecnologias de adoção comercial recente.

O procedimento de coleta de dados compreendeu três etapas. Na primeira etapa, foi reconstituída a linha do tempo histórica das plataformas SAP (R/2, R/3, mySAP, NetWeaver, ECC, HANA e S/4HANA), a partir de documentação técnica oficial, blogs corporativos da comunidade SAP (SAP Community) e literatura especializada publicada pela editora SAP PRESS. Na segunda etapa, foram sistematizados os fundamentos teóricos de microsserviços, Indústria 4.0 e IA aplicada a ERP, a partir de referências acadêmicas amplamente citadas na

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

área (FOWLER; LEWIS, 2014; NEWMAN, 2021; KAGERMANN; WAHLSTER; HELBIG, 2013; RUSSELL; NORVIG, 2020). Na terceira etapa, foi realizada a caracterização técnica do ABAP RAP e da SAP BTP, com base em documentação oficial do SAP Help Portal e da SAP Community, complementada por obra de referência sobre o tema (HAAS; MATHEW, 2020). A análise dos dados coletados foi conduzida por meio de análise de conteúdo qualitativa, buscando-se identificar, em cada fase histórica examinada, os elementos arquiteturais que anteciparam ou viabilizaram tecnicamente a fase subsequente. A partir dessa análise, foi elaborada uma arquitetura de referência de natureza conceitual — não implementada empiricamente neste trabalho —, cuja finalidade é ilustrar, de forma sistemática, a integração entre microsserviços ABAP Cloud, serviços de IA da SAP BTP e barramentos de eventos, aplicada a um cenário simulado de automação industrial. Como limitação metodológica, reconhece-se que a ausência de validação empírica em ambiente produtivo real restringe o alcance das conclusões a um nível conceitual e prescritivo, não permitindo inferências quantitativas sobre desempenho, custo ou retorno sobre investimento.

O PAPEL DO ABAP RAP NA ARQUITETURA MODERNA SAP

Enquanto a seção 2 demonstrou que a evolução das arquiteturas SAP criou, ao longo de décadas, as pré-condições técnicas — computação in-memory, interfaces orientadas a serviços e simplificação do modelo de dados — para uma transição em direção a microsserviços, esta seção examina o instrumento técnico específico por meio do qual essa transição se operacionaliza dentro do núcleo transacional: o ABAP RESTful Application Programming Model (RAP).

Dos programas monolíticos ao Clean Core

Historicamente, a customização de sistemas SAP baseava-se na modificação direta ou na extensão de objetos de repositório dentro do próprio núcleo transacional, prática que gerava forte acoplamento entre a lógica de negócio específica de cada cliente e o código-fonte padrão fornecido pela SAP. Esse modelo, sustentável durante a era do R/3 e do ECC — quando as atualizações de versão ocorriam em ciclos plurianuais —, tornou-se progressivamente incompatível com o modelo de entrega contínua (continuous delivery) adotado pela SAP a partir do S/4HANA Cloud, cujas atualizações trimestrais exigem que extensões customizadas

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

sejam desacopladas do núcleo, sob pena de conflitos recorrentes em cada ciclo de upgrade.

Em resposta a esse dilema, a SAP formalizou o princípio do clean core (núcleo limpo), segundo o qual o sistema padrão deve permanecer o mais próximo possível de sua configuração original, sendo toda extensão específica de cliente implementada por meio de APIs liberadas (released APIs), extensibilidade in-app baseada em pontos de extensão predefinidos, ou desenvolvimento de aplicações lado a lado (side-by-side) hospedadas na SAP BTP. O ABAP RAP constitui o instrumento técnico central para a implementação desse princípio dentro do próprio ambiente ABAP, ao formalizar um modelo de programação estruturado em camadas bem definidas, orientado a objetos de negócio (Business Objects) e compatível tanto com extensibilidade in-app quanto com desenvolvimento lado a lado na BTP.

RAP: conceitos, camadas e ciclo de desenvolvimento

O ABAP RAP foi apresentado pela SAP em 2018 e tornou-se disponível para uso produtivo em ambientes on-premise a partir da versão SAP S/4HANA 1909, sendo, desde então, disponibilizado simultaneamente no ambiente ABAP da SAP BTP (SAP BTP ABAP Environment) e no SAP S/4HANA Cloud, tanto em sua edição pública quanto privada. Conceitualmente, o RAP é descrito pela documentação oficial da SAP como sucessor evolutivo do ABAP Programming Model for SAP Fiori, simplificando substancialmente o processo de criação de serviços OData ao eliminar a necessidade de transações do SAP GUI e de geração manual de serviços por meio da transação SEGW, características de modelos de programação anteriores.

Do ponto de vista técnico, o RAP estrutura-se em quatro camadas principais. A camada de modelagem de dados é construída sobre Core Data Services (CDS Views), que definem entidades, associações e anotações semânticas diretamente no nível do banco de dados, aproveitando a capacidade de processamento in-memory do SAP HANA para delegar operações de agregação e junção (join) ao próprio banco de dados, em vez de processá-las na camada de aplicação — princípio conhecido como code-to-data (levar o código até os dados, e não os dados até o código).

A camada de comportamento (behavior definition), escrita na Behavior Definition Language (BDL), especifica as operações permitidas sobre cada objeto de negócio — criação, leitura, atualização, exclusão e ações customizadas —, bem como validações, determinações automáticas de valores e regras de autorização. A camada de implementação de comportamento (behavior implementation), escrita em ABAP orientado a objetos convencional, contém a lógica

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

RCMOS – Revista Científica Multidisciplinar O Saber.
ISSN: 2675-9128. São Paulo-SP.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

de negócio propriamente dita. Por fim, a camada de exposição de serviço (service exposure) publica o objeto de negócio como serviço OData V2 ou V4, consumível tanto por aplicações SAP Fiori quanto por sistemas externos, incluindo serviços hospedados na SAP BTP.

O RAP oferece dois tipos de implementação: o tipo gerenciado (managed implementation), no qual grande parte da lógica de persistência é gerada automaticamente pelo framework a partir da behavior definition, recomendado para desenvolvimentos greenfield (partindo do zero); e o tipo não gerenciado (unmanaged implementation), no qual o desenvolvedor implementa manualmente a lógica de persistência, recomendado para cenários brownfield de integração com lógica de negócio legada já existente no sistema. Essa dualidade permite que organizações com décadas de customização acumulada no ECC ou no S/4HANA on-premise adotem o RAP de forma incremental, sem a necessidade de reescrever integralmente seus processos de negócio.

RAP como habilitador de microsserviços dentro do S/4HANA

É possível caracterizar cada objeto de negócio RAP como uma unidade funcional coesa, dotada de contrato de interface bem definido (o serviço OData exposto), encapsulamento de lógica interna (a implementação de comportamento) e independência relativa de outros objetos de negócio, comunicando-se com estes preferencialmente por meio de composições e associações formalmente declaradas, em vez de acoplamento direto no nível de código. Essa caracterização aproxima o RAP, em escala reduzida e dentro dos limites de um único sistema ABAP, dos princípios estruturais que Newman (2021) atribui a microsserviços — ainda que se trate, tecnicamente, de uma modularização interna ao monólito, e não de serviços fisicamente distribuídos em processos independentes.

A distinção é relevante: dentro do próprio S/4HANA, o RAP viabiliza o que a literatura de arquitetura de software denomina monólito modular (modular monolith) — um sistema que preserva a unidade de implantação e a integridade transacional centralizada, mas organiza seu código interno em módulos fortemente coesos e fracamente acoplados, análogos a microsserviços em sua estrutura lógica, embora não em sua topologia de execução. É precisamente essa modularidade lógica que permite ao RAP, quando combinado com a SAP BTP, estender-se para uma arquitetura de microsserviços fisicamente distribuída: cada objeto de negócio RAP pode ser consumido, por meio de sua API OData padronizada, por serviços hospedados na BTP — implementados em Java, Node.js, Python ou CAP (Cloud Application

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

Programming Model) —, que assumem funções específicas, como orquestração de fluxos de automação industrial ou execução de modelos de Inteligência Artificial, sem necessidade de acesso direto ao banco de dados do S/4HANA.

Dessa forma, o RAP não substitui a arquitetura de microsserviços, mas a complementa a partir de dentro do núcleo transacional: ele garante que o núcleo permaneça limpo, coeso e atualizável, ao mesmo tempo em que expõe, de forma padronizada e governada, os pontos de extensão necessários para que serviços externos — hospedados na BTP ou em outras plataformas de nuvem — possam compor, junto ao ERP, uma arquitetura distribuída mais ampla, orientada à automação de processos industriais mediada por Inteligência Artificial, tema desenvolvido na seção seguinte.

SAP BTP COMO PLATAFORMA DE INTEGRAÇÃO, DADOS E INTELIGÊNCIA ARTIFICIAL

Se o ABAP RAP viabiliza a modularidade e a extensibilidade governada dentro do núcleo digital, é a SAP Business Technology Platform (BTP) que fornece o ambiente de execução distribuído no qual essa modularidade se expande para uma arquitetura de microsserviços propriamente dita, incorporando capacidades de integração, gestão de dados e Inteligência Artificial que extrapolam o escopo técnico do ambiente ABAP tradicional.

Da SAP Cloud Platform à Business Technology Platform

A trajetória da BTP remonta ao lançamento, em 2012, da SAP HANA Cloud Platform, posteriormente renomeada SAP Cloud Platform (SCP) em 2017, e finalmente reposicionada como SAP Business Technology Platform em 2021, consolidando sob uma única marca comercial quatro pilares até então oferecidos de forma relativamente segmentada: banco de dados e gestão de dados (Database and Data Management), analytics, integração de aplicações e dados (Integration Suite) e extensibilidade de aplicações (Extension Suite), ao qual se somaram, mais recentemente, os serviços de Inteligência Artificial reunidos sob a marca SAP Business AI.

Essa consolidação reflete um movimento estratégico mais amplo da SAP, orientado pelo conceito de RISE with SAP (2021) e, posteriormente, GROW with SAP, segundo o qual o S/4HANA passa a ser comercializado e implantado como núcleo digital padronizado, ao redor do qual gravitam serviços de extensão, integração e inteligência hospedados na BTP —

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

reproduzindo, em escala organizacional e comercial, o mesmo princípio arquitetural de núcleo coeso mais periferia distribuída já identificado, em escala técnica, na relação entre RAP e microsserviços.

Serviços de Inteligência Artificial: SAP AI Core, AI Foundation e Business AI

No âmbito da BTP, o SAP AI Core constitui o serviço responsável pela execução e pelo gerenciamento do ciclo de vida de modelos de aprendizado de máquina em ambiente Kubernetes, oferecendo capacidades de treinamento, versionamento e disponibilização de modelos (model serving) por meio de APIs REST. Esse serviço é complementado pelo SAP AI Launchpad, interface centralizada para monitoramento e governança de cenários de IA, e pelo conjunto mais amplo denominado SAP AI Foundation, que reúne acesso a modelos de linguagem de grande porte (Large Language Models) de terceiros — como os disponibilizados por meio de parcerias com provedores especializados — e a ferramentas de orquestração de fluxos de inteligência artificial generativa.

Sob a marca SAP Business AI, a empresa posiciona comercialmente um conjunto de capacidades de IA já embarcadas em processos de negócio específicos do S/4HANA e de outras soluções da linha de produtos, como recomendações preditivas em planejamento de demanda, detecção de anomalias em lançamentos financeiros e assistentes conversacionais de interação em linguagem natural (SAP Joule). Do ponto de vista arquitetural, o aspecto mais relevante para a automação industrial é a possibilidade de os modelos hospedados no SAP AI Core consumirem dados operacionais expostos por serviços OData construídos com o ABAP RAP, processarem esses dados por meio de algoritmos de aprendizado de máquina, e retroalimentarem o núcleo transacional com resultados — como probabilidades de falha de equipamento ou classificações de qualidade —, novamente por meio de operações padronizadas do próprio objeto de negócio RAP, fechando um ciclo de retroalimentação entre dados operacionais, inferência de IA e ação transacional.

Integration Suite, Event Mesh e arquitetura orientada a eventos

A comunicação entre o núcleo digital S/4HANA, os serviços de IA hospedados na BTP e sistemas de automação industrial externos (MES, SCADA, historiadores de processo) depende de mecanismos robustos de integração, providos pelo SAP Integration Suite. Esse conjunto de

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

serviços inclui o Cloud Integration, responsável pela orquestração de fluxos de integração ponto a ponto e mediados por padrões como REST, SOAP e OData; o API Management, responsável pela governança, versionamento e segurança de APIs expostas externamente; e o SAP Event Mesh, um barramento de mensageria baseado no paradigma de publicação e assinatura (publish/subscribe), que viabiliza arquiteturas orientadas a eventos em escala.

É o Event Mesh que permite a um sensor industrial — ou, mais precisamente, ao sistema de automação que o supervisiona — publicar um evento de anomalia sem necessidade de conhecimento prévio sobre quais serviços consumirão essa informação, sejam eles serviços RAP dentro do S/4HANA, aplicações CAP hospedadas na BTP, ou modelos de IA em execução no SAP AI Core. Esse desacoplamento temporal e topológico entre produtores e consumidores de eventos constitui condição arquitetural necessária para que a automação de processos industriais opere em tempo real, sem a criação de dependências síncronas rígidas entre o chão de fábrica e o núcleo transacional do ERP — risco que comprometeria tanto o desempenho quanto a resiliência do sistema como um todo.

CONVERGÊNCIA: MICROSSERVIÇOS, RAP, BTP E IA NA AUTOMAÇÃO DE PROCESSOS INDUSTRIAIS

As seções anteriores demonstraram, de forma isolada, como a evolução histórica das arquiteturas SAP (seção 2), os fundamentos teóricos de microsserviços, Indústria 4.0 e IA (seção 3), o ABAP RAP (seção 5) e a SAP BTP (seção 6) constituem peças de um mesmo quebra-cabeça arquitetural. Esta seção propõe uma arquitetura de referência que articula essas peças em torno de um cenário aplicado de automação industrial, seguida de discussão sobre seus desafios de implementação.

Arquitetura de referência proposta

A arquitetura de referência proposta organiza-se em quatro camadas lógicas. A camada de borda industrial (edge layer) compreende sensores, controladores lógicos programáveis e sistemas MES/SCADA responsáveis pela coleta primária de dados de processo — temperatura, vibração, pressão, velocidade de linha — tipicamente agregados por um gateway de borda (edge gateway) antes de sua transmissão à nuvem, de modo a reduzir volume de tráfego e latência.

A camada de integração e eventos, sustentada pelo SAP Integration Suite e pelo SAP Event

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

Mesh, recebe os dados da camada de borda e os distribui, de forma assíncrona, aos consumidores interessados. A camada de inteligência, hospedada no SAP AI Core, consome os eventos relevantes, aplica modelos de aprendizado de máquina previamente treinados — por exemplo, um modelo de classificação de risco de falha de rolamento industrial treinado sobre séries históricas de vibração — e publica seus resultados de volta ao barramento de eventos ou os disponibiliza por meio de API REST.

Por fim, a camada transacional, constituída pelo núcleo digital S/4HANA, recebe os resultados da inferência de IA por meio de objetos de negócio RAP especificamente projetados para esse fim — como uma extensão do objeto de negócio de ordem de manutenção (Maintenance Order), capaz de criar automaticamente uma ordem preventiva quando a probabilidade de falha calculada pelo modelo de IA ultrapassa determinado limiar. Todo esse fluxo opera sem que a lógica de inferência de IA resida dentro do núcleo ABAP, e sem que a lógica transacional de negócio resida fora dele — preservando, simultaneamente, o princípio do clean core e a modularidade distribuída característica de microsserviços.

Figura 1 — Fluxo conceitual da arquitetura de referência: Borda Industrial → SAP Event Mesh → SAP AI Core (inferência) → Objeto de Negócio RAP (S/4HANA) → Ordem de Manutenção Preventiva. Fonte: elaborado pelo(a) autor(a) (2026).

Caso ilustrativo: manutenção preditiva em linha de produção

Para ilustrar a arquitetura proposta, considere-se um cenário hipotético de manutenção preditiva em uma linha de envase industrial. Sensores de vibração instalados em motores elétricos críticos transmitem leituras a cada segundo a um gateway de borda, que agrega essas leituras em janelas de sessenta segundos e as publica como eventos no SAP Event Mesh. Um serviço hospedado no SAP AI Core, previamente treinado com dados históricos de falhas rotuladas, consome esses eventos e calcula, a cada janela, a probabilidade de falha iminente do equipamento nas próximas setenta e duas horas.

Quando essa probabilidade ultrapassa um limiar predefinido — por exemplo, setenta e cinco por cento —, o serviço de IA publica um evento de alerta, consumido por uma aplicação CAP hospedada na SAP BTP, responsável por enriquecer o alerta com dados contextuais adicionais (histórico de manutenções anteriores do equipamento, disponibilidade de peças de reposição em estoque) obtidos por meio de chamadas a objetos de negócio RAP expostos pelo S/4HANA. Confirmada a relevância do alerta, a aplicação CAP invoca uma ação (action) do objeto de

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

negócio RAP de ordem de manutenção, criando automaticamente uma ordem preventiva, atribuída à equipe de manutenção responsável, com prioridade calculada em função da criticidade do equipamento e da probabilidade de falha estimada.

Esse ciclo completo — da leitura do sensor à criação da ordem de manutenção no ERP — ilustra de forma concreta a tese central deste artigo: a automação de processos industriais orientada por Inteligência Artificial, no ecossistema SAP contemporâneo, não decorre de uma única tecnologia isolada, mas da orquestração entre microsserviços de integração e inteligência, hospedados na BTP, e um núcleo transacional modular e governado, viabilizado pelo ABAP RAP.

Desafios: governança, integração legada e segurança

A implementação da arquitetura proposta não está isenta de desafios significativos. Em primeiro lugar, a governança de APIs torna-se elemento crítico: à medida que o número de objetos de negócio RAP expostos como APIs liberadas cresce, organizações necessitam de processos formais de gestão de ciclo de vida de API — versionamento, depreciação controlada e documentação —, sob pena de reproduzir, em escala distribuída, o mesmo problema de acoplamento rígido que a arquitetura de microsserviços busca evitar.

Em segundo lugar, a integração com sistemas legados de automação industrial — muitos dos quais operando protocolos proprietários ou padrões industriais anteriores à popularização de APIs REST, como OPC-UA, Modbus ou Profibus — frequentemente exige camadas adicionais de conversão de protocolo na borda, antes que os dados possam ser incorporados ao fluxo de eventos baseado em padrões web descrito na seção 7.1. Em terceiro lugar, a superfície de exposição criada pela integração entre chão de fábrica, nuvem pública e núcleo transacional amplia consideravelmente os riscos de segurança da informação, exigindo controles rigorosos de autenticação, autorização e segmentação de rede entre as camadas de borda, integração, inteligência e transação — especialmente relevante em setores industriais sujeitos a regulação de infraestrutura crítica.

DISCUSSÃO

Os achados sistematizados neste artigo permitem retomar a hipótese inicial de que a adoção de microsserviços no ecossistema SAP constitui processo evolutivo cumulativo, e não ruptura abrupta. A análise histórica da seção 2 evidencia que cada transição arquitetural — do R/2 ao

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

R/3, do R/3 ao NetWeaver, do ECC ao HANA, do HANA ao S/4HANA — respondeu a limitações técnicas específicas da geração anterior, ao mesmo tempo em que preservou elementos estruturais fundamentais, como a centralidade da linguagem ABAP e o princípio de integração de dados sobre base única. Essa continuidade histórica confirma-se tecnicamente na própria concepção do RAP, que a documentação oficial da SAP descreve como sucessor evolutivo — e não substituto disruptivo — do ABAP Programming Model for SAP Fiori.

Do ponto de vista teórico, este artigo contribui para a literatura ao propor uma distinção conceitual entre modularidade lógica interna ao núcleo transacional — viabilizada pelo RAP sob a forma de um monólito modular — e distribuição física de serviços em uma arquitetura de microsserviços propriamente dita — viabilizada pela combinação entre RAP e SAP BTP. Essa distinção mostra-se relevante porque a literatura corrente sobre microsserviços (NEWMAN, 2021; FOWLER; LEWIS, 2014), concebida majoritariamente a partir de sistemas nativos de nuvem construídos desde sua origem sob esse paradigma, não trata adequadamente do caso de sistemas ERP legados de grande escala, cuja transição para microsserviços não pode prescindir de um núcleo transacional coeso, sob risco de comprometer garantias de integridade de dados críticas para operações financeiras e de manufatura.

Sob a perspectiva da automação industrial, os resultados sugerem que a principal contribuição da convergência entre RAP, BTP e IA não reside na velocidade bruta de processamento — já assegurada, em grande medida, pela computação in-memory do SAP HANA desde 2011 —, mas na governabilidade da integração entre inteligência artificial e processos transacionais de negócio. Isto é, a arquitetura proposta na seção 7 não torna a manutenção preditiva tecnicamente possível pela primeira vez — modelos de aprendizado de máquina para esse fim já são amplamente empregados em plataformas de IoT industrial independentes de ERP —, mas torna possível que os resultados dessa predição se traduzam, de forma auditável, governada e integrada, em ações transacionais dentro do sistema de gestão empresarial, sem exigir integração pontual e não padronizada entre sistemas heterogêneos.

Como limitação deste estudo, reitera-se seu caráter eminentemente teórico-conceitual: a arquitetura de referência proposta não foi submetida a validação empírica em ambiente produtivo, e a literatura acadêmica revisada por pares sobre o ABAP RAP e a SAP BTP permanece incipiente, o que exigiu recurso substancial à documentação técnica oficial do fabricante — fonte legítima, mas sujeita a viés comercial inerente à própria natureza da SAP como parte interessada na promoção de suas tecnologias proprietárias. Estudos futuros de natureza empírica, incluindo estudos de caso em ambientes industriais reais e avaliações

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

quantitativas de desempenho e custo, são necessários para corroborar ou qualificar as proposições aqui apresentadas.

CONSIDERAÇÕES FINAIS

Este artigo analisou, sob perspectiva histórica e evolutiva, a trajetória das arquiteturas SAP desde o R/2 até o S/4HANA, demonstrando como cada geração tecnológica criou condições técnicas cumulativas para a transição, hoje observável, em direção a arquiteturas de microsserviços orientadas por Inteligência Artificial. Verificou-se que o ABAP RAP opera como instrumento de modularização governada do núcleo transacional, materializando o princípio do clean core e viabilizando extensibilidade sem comprometer a integridade e a atualizabilidade contínua do sistema padrão. Verificou-se, complementarmente, que a SAP BTP fornece o ambiente de execução distribuído — integração, dados e Inteligência Artificial — no qual essa modularidade se expande para uma arquitetura de microsserviços fisicamente distribuída, capaz de sustentar cenários de automação de processos industriais em tempo real.

A arquitetura de referência proposta na seção 7, ilustrada por meio de um caso hipotético de manutenção preditiva, demonstrou de que forma esses elementos — camada de borda industrial, integração orientada a eventos, inferência de Inteligência Artificial e núcleo transacional modular — podem ser articulados de forma coerente e governada. Conclui-se que a principal contribuição dessa convergência tecnológica não reside isoladamente em nenhum de seus componentes, mas na capacidade de integrá-los de forma padronizada, auditável e resiliente a ciclos de atualização contínua — capacidade que distingue a proposta arquitetural aqui discutida de integrações pontuais e não governadas historicamente comuns em ambientes de automação industrial.

Como agenda de pesquisa futura, recomenda-se a condução de estudos de caso empíricos em organizações industriais que tenham efetivamente implementado arquiteturas similares à proposta, com vistas a mensurar, de forma quantitativa, indicadores de desempenho, resiliência operacional e retorno sobre investimento. Recomenda-se, ainda, a investigação aprofundada dos desafios de governança de dados e segurança da informação inerentes à ampliação da superfície de integração entre chão de fábrica, nuvem pública e sistemas transacionais críticos, tema que, dada sua relevância crescente, extrapola os limites deste artigo e demanda tratamento específico em pesquisas subsequentes.

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.1 2026 | submissão: 25/08/2026 | aceito: 28/08/2026 | publicação: 31/08/2026

REFERÊNCIAS

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 6023: informação e documentação: referências: elaboração. Rio de Janeiro: ABNT, 2018.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 14724: informação e documentação: trabalhos acadêmicos: apresentação. Rio de Janeiro: ABNT, 2011.

FOWLER, Martin; LEWIS, James. Microservices: a definition of this new architectural term. martinfowler.com, 25 mar. 2014. Disponível em: https://martinfowler.com/articles/microservices.html. Acesso em: 02 jul. 2026.

GIL, Antonio Carlos. Como elaborar projetos de pesquisa. 6. 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. 2. ed. Sebastopol: O'Reilly Media, 2021.

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

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

SAP SE. ABAP RESTful Application Programming Model (RAP). SAP Community, 2024. Disponível em: https://pages.community.sap.com/topics/abap/rap. Acesso em: 05 ago. 2026.

SAP SE. Getting started with the ABAP RESTful Application Programming Model (RAP). SAP Community Technology Blog, jan. 2024. Disponível em: https://community.sap.com/t5/technology-blog-posts-by-sap/getting-started-with-the-abaprestful-application-programming-model-rap/ba-p/13420829. Acesso em: 12 ago. 2026.

SAP SE. SAP Business Technology Platform. Disponível em: https://www.sap.com/products/technology-platform.html. Acesso em: 12 ago. 2026.

SAP SE. SAP S/4HANA. Disponível em: https://www.sap.com/products/erp/s4hana.html. Acesso em: 17 ago. 2026.

DEUTSCHE SAP-ANWENDERGRUPPE (DSAG). Das ABAP RESTful Application Programming Model (RAP). ABAP-Leitfaden, 2024. Disponível em: https://1dsag.github.io/ABAP-Leitfaden/abap/restful_abap/. Acesso em: 19 ago. 2026.

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.