



One Event Backbone for SAP: Universal Change Data Capture for Applications and Analytics

Why application-level CDC from the SAP core should feed both your event-driven process landscape and your data lake — in both directions, from SAP ECC to SAP S/4HANA, from SAP Advanced Event Mesh to Microsoft OneLake.

Publication

Type	Whitepaper
Practice	Integration + Technology
Published	July 2026
Pages	10
Product	ASAPIO Integration Add-on

Contents

Executive Summary	2
The Integration Gap	3
What This Means for the C-Suite	4
The Challenge	5
The ASAPIO Approach	6
Consumption Patterns	7
Architectural Assessment	8
Recommendations	9

Executive Summary

Two integration worlds, one source of change — and one capture layer that can serve them both.

Enterprises running SAP face the same architectural question twice. Applications, automation platforms and microservices need to know **when** something changes in SAP. Analytics platforms and data lakes such as **Microsoft OneLake** or Kafka-fed lakehouses need a continuous feed of **what** changed, in business context. And increasingly, AI systems running outside SAP need both.

Most organizations answer these questions with disconnected toolchains: point-to-point interfaces, custom ABAP and iPaaS polling on the application side; batch extractors or database-log replication on the analytics side. The result is duplicated capture logic, inconsistent data and rising maintenance cost — at exactly the moment SAP's **Clean Core** strategy and the ECC-to-S/4HANA transition demand simplification.

This paper argues for a different pattern: capture business change **once**, at the application level, inside the SAP system — and fan it out to every consumer. The **ASAPIO Integration Add-on** implements this as a code-less, Clean Core-compliant event enablement layer for SAP ECC and SAP S/4HANA, emitting standardized CloudEvents to brokers, automation platforms and analytics destinations — and processing inbound events from the non-SAP world back into the ERP.

Key findings

- **AI has moved outside SAP; the data it needs has not.** In productive use, non-SAP AI dominates by a wide margin — making real-time, semantically intact SAP data an AI prerequisite, not an analytics nicety.
- **Application-level CDC preserves business semantics.** Database logs see table rows; the application layer sees business objects, statuses and context.
- **Operational and analytical consumers can share one capture layer.** The event that triggers a process is the same change that must land in the lakehouse; capturing it twice is architectural debt.
- **Integration is bidirectional.** The same layer that publishes SAP events can post inbound events from webshops, sensors and partner systems into SAP.

Bottom line

- Treat SAP change data as a **product with many consumers**, not a per-interface problem: one application-level capture layer, one event standard (CloudEvents), many pluggable destinations — from SAP Advanced Event Mesh and Kafka to Microsoft OneLake.

The Integration Gap

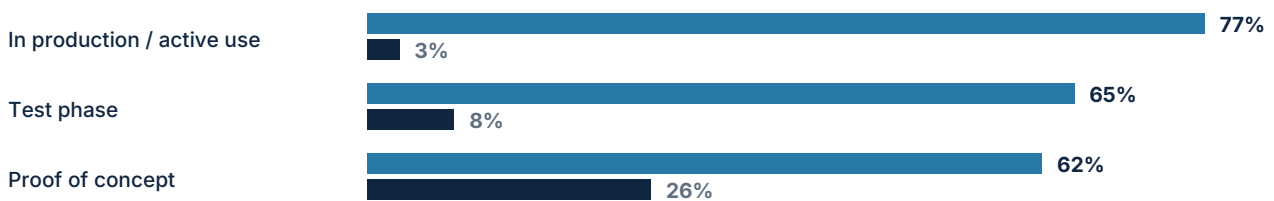
AI has moved outside SAP faster than SAP data can follow it.

The most consequential shift in the SAP landscape is not which AI models enterprises adopt — it is where they run. Survey evidence from the German-speaking SAP user group shows that among companies with AI use cases already in productive use, the overwhelming majority run them on non-SAP platforms. At the same time, integration has become the fastest-growing investment area on the SAP Business Technology Platform, nearly doubling in two years.

Where AI use cases run — by maturity stage

Respondents with implemented AI use cases, by platform

■ Non-SAP solutions
■ SAP solutions

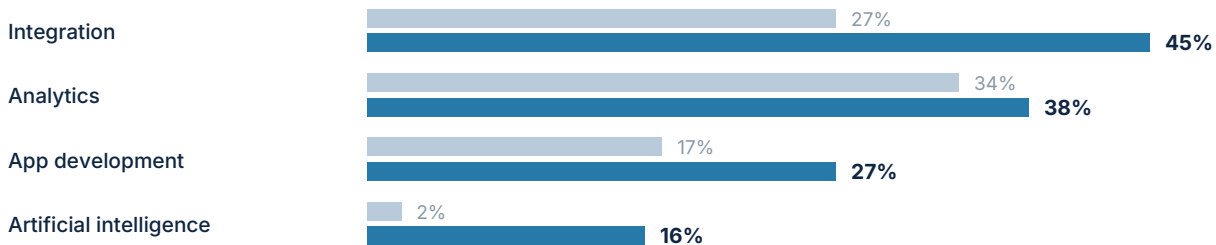


Base: n=85 — the 43 % of 198 respondents who had implemented AI use cases. Multiple answers possible; stages are not mutually exclusive.

Integration is the fastest-growing SAP BTP investment area

Share planning high or medium investment, by BTP area

■ 2024
■ 2026



Base: n=198 (2026). Comparison values from DSAG-Investitionsreport 2024; different sample.

Figure 1. Source: DSAG-Investitionsreport 2026, Deutschsprachige SAP-Anwendergruppe e. V., press release of 26 February 2026. Survey of 198 SAP user organizations in Germany, Austria and Switzerland, conducted 8 December 2025 to 21 January 2026; one respondent per member company, in CIO, competence-centre lead or equivalent roles. Chart compiled by ASAPIO from the published figures; percentages as reported.

What This Means for the C-Suite

Seven questions executives ask about SAP event enablement — and the short answers.

Why this matters for SAP customers

Three consequences follow. The ERP is no longer where intelligence is applied — it is where the truth is recorded, and that truth must reach outside systems quickly and with meaning intact. Landscapes remain hybrid, so any answer must work on SAP ECC and SAP S/4HANA alike. And SAP's **Clean Core** direction rules out solving it with more custom code in the core. The gap is therefore not a modelling or licensing problem: it is an **integration** problem — the one area where SAP customers are already increasing spend fastest.

AI readiness	Your AI runs outside SAP. This makes SAP business objects available to it within seconds, as curated events rather than reverse-engineered table extracts.
Time to value	Capture is configured, not coded. A first productive event feed is a customizing exercise measured in weeks, not an ABAP development project measured in quarters.
Cost of change	A new consumer is a destination entry, not a project. Fan-out to an additional broker, webhook or lake path leaves the capture side untouched.
Transition risk	Feeds built today on SAP ECC carry over to SAP S/4HANA. The integration layer stops being the largest single workstream of the transformation.
Audit and upgrade	Code-less customizing on released, upgrade-stable interfaces. There are no core modifications to defend at the next upgrade or in the next architecture review.
Data consistency	Process events and lakehouse records originate from one capture point. Reconciliation between the operational and the analytical view of SAP disappears by construction.
Vendor lock-in	CloudEvents as the format, pluggable adapters as the transport. The target landscape — SAP BTP-, Microsoft- or Kafka-centric, or all three — stays yours to choose and to change.

The Challenge

Fragmented capture: four ways to get SAP data, none of them universal.

Most SAP landscapes have accumulated several extraction and integration mechanisms in parallel. Each was a reasonable answer to a single requirement; together they form a fragile, expensive estate.

Criterion	Batch extractors / scheduled jobs	Database-log CDC (table level)	iPaaS / API polling	Application-level CDC (ASAPIO approach)
Latency	Hours (schedule-bound)	Seconds	Minutes (poll interval)	Near real time (on change)
Business semantics	Partial — extract structures	Low — raw table rows, no business context	Partial — API-shaped	High — business objects, statuses, context
Clean Core fit (S/4HANA)	Varies — often custom ABAP	Bypasses application layer ; licensing exposure	Good if released APIs used	Designed for it — code-less, released interfaces
ECC support	Yes	Product-dependent	Limited (API coverage)	Yes — same product on ECC and S/4HANA
Feeds processes (events)	No	Without semantics	Pull only	Yes — native push, CloudEvents
Feeds analytics / AI	Yes , but stale	Yes , without context	Inefficient at volume	Yes — same events, lake-ready payloads
Inbound into SAP	No	No (read-only)	Yes	Yes — posting via released APIs
Capture logic instances	One per interface / pipeline			One, shared by all consumers

Table 1. Comparison of SAP change-data approaches against dual-use (operational + analytical) requirements. Qualitative assessment by ASAPIO; individual products may vary.

The hidden cost: capturing the same change many times

A single business event — a change to a material master, say — is typically detected independently by a nightly extractor, a replication tool, two custom interfaces and an API poller: five mechanisms, five failure modes, five interpretations of “what changed”, five things to migrate on the way to S/4HANA — and permanent reconciliation between the operational and analytical views of the same data.

The ASAPIO Approach

Capture once at the application level — then fan out to every consumer.

The **ASAPIO Integration Add-on** is installed inside the SAP system and event-enables the ERP itself. Changes to business objects are detected at the **application level** — where documents, statuses and business context exist — and published as standardized **CloudEvents** through pluggable protocol adapters. Configuration is code-less: which objects, which fields, which payload, which destination, all defined in customizing. On SAP S/4HANA the runtime follows Clean Core principles and relies on released, upgrade-stable interfaces.

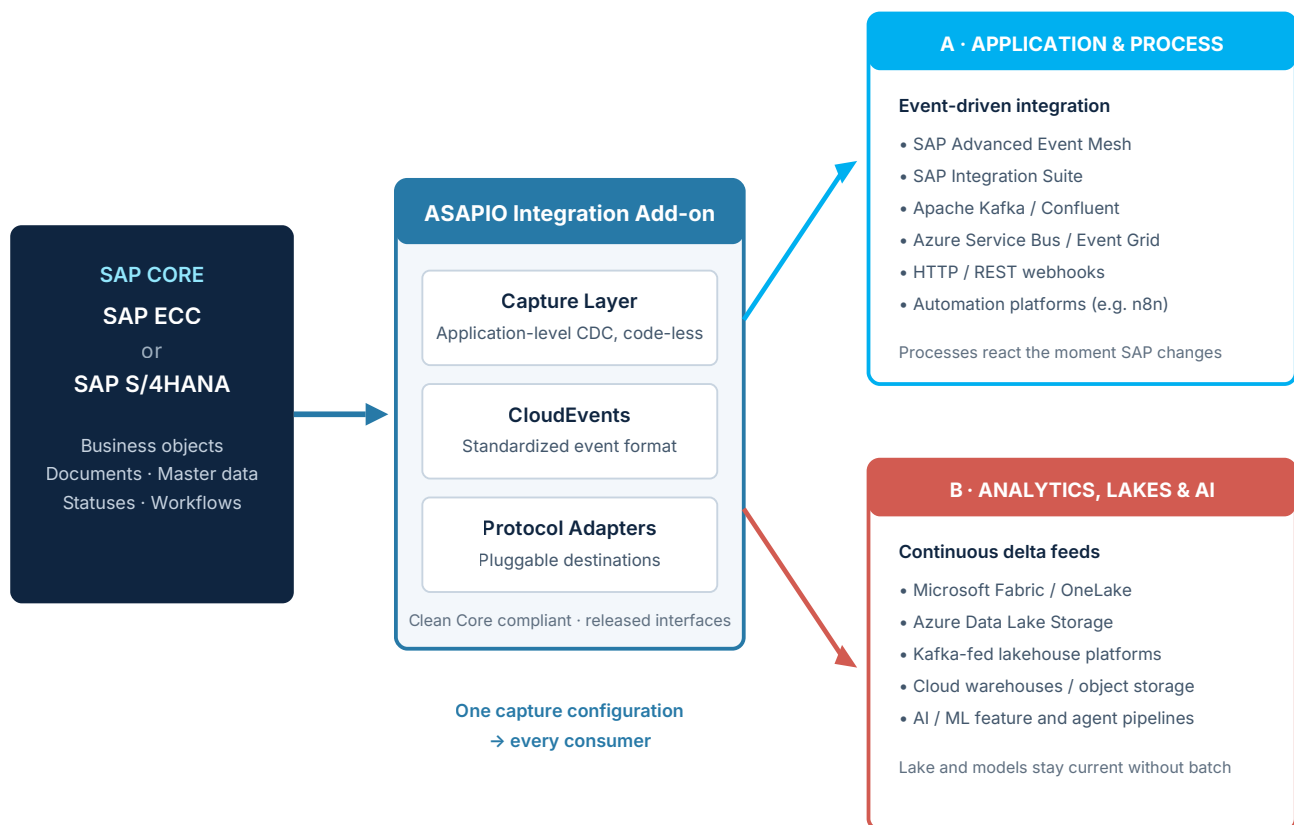


Figure 2. Universal event backbone: the ASAPIO Integration Add-on captures change once at the application level and serves both operational (A) and analytical (B) consumers.

Because the same product runs on SAP ECC and SAP S/4HANA, the backbone is **transition-proof**: feeds built today on ECC continue to operate after the move to S/4HANA.

Consumption Patterns

The same event, two audiences — and a return path from the non-SAP world.

Pattern A — Event-driven processes

Operational consumers need small, fast, semantically precise signals: sales order 4711 created, business partner X changed bank data, goods receipt posted. These are published as CloudEvents to messaging backbones or directly to HTTP endpoints and automation platforms. Typical outcomes: real-time master data distribution to e-commerce and CRM; microservices that subscribe to SAP events instead of calling into the ERP; workflow automation triggered the moment SAP changes; and PI/PO modernization, replacing polling with push-based eventing.

Pattern B — Continuous feeds for analytics and AI

Analytical and AI consumers need the same changes with fuller payloads, delivered continuously. Because capture happens at the application level, payloads arrive as **curated business objects** rather than raw table fragments. Landing zones include Microsoft Fabric / OneLake, Azure Data Lake Storage and Kafka topics. The results: a lakehouse that reflects the ERP within minutes; retired batch extractors; AI pipelines and agents consuming fresh, semantically labeled SAP data; and a single source of truth shared with the operational side.

Pattern C — Inbound: closing the loop

A universal backbone is bidirectional. The same add-on processes **inbound** events from webshops, marketplaces, IoT sensors or partner platforms and posts them into SAP through released APIs and BAPIs. Figure 3 shows a retail Click & Collect flow: the webshop order enters SAP as a posted sales order with store-level stock reservation within seconds; confirmation and pickup-ready events flow back out to webshop and store app.

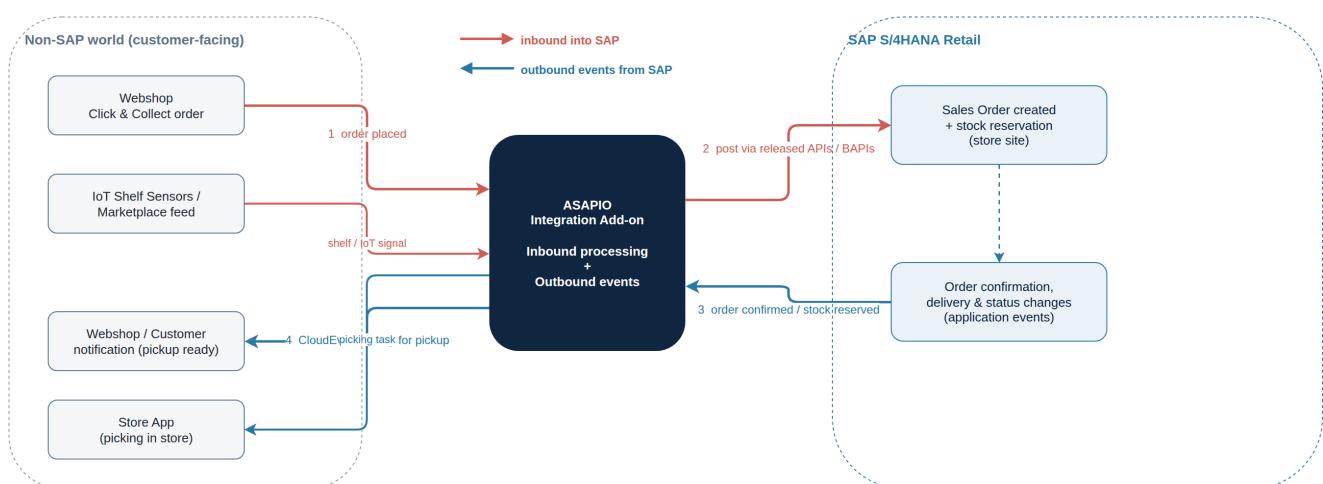


Figure 3. Bidirectional integration in retail (Click & Collect): inbound webshop orders and IoT signals are posted into SAP via released APIs (red), while confirmation and pickup-ready events flow back as CloudEvents (blue).

Architectural Assessment

Why one backbone beats two toolchains.

Consistency by construction

One capture point means one definition of “what changed”. The operational and analytical views of SAP cannot drift apart, because they are the same stream.

Clean Core alignment

Code-less configuration on released interfaces keeps the core clean and upgrades predictable — on S/4HANA by design, on ECC as a disciplined bridge toward it.

Migration de-risking

Feeds defined on ECC carry over to S/4HANA, so the integration layer stops being the largest workstream of the transformation.

Vendor-neutral distribution

CloudEvents and pluggable adapters keep the target landscape open: SAP BTP-centric, Microsoft-centric, Kafka-centric — or all three in parallel.

Selected capability summary

Dimension	ASAPIO Integration Add-on
Deployment	ABAP add-on inside SAP ECC or SAP S/4HANA; no separate middleware server required for capture
Change detection	Application-level CDC on business objects (master data and transactional documents), configuration-driven
Event format	CloudEvents-based, standardized and consumer-neutral
Destinations	SAP Advanced Event Mesh, SAP Integration Suite, Apache Kafka / Confluent, Microsoft Azure (messaging, Data Lake, Fabric / OneLake), AWS, Google Cloud, HTTP/REST, automation platforms
Inbound	Event- and message-driven inbound processing; posting into SAP via released APIs / BAPIs
Clean Core	Code-less customizing; S/4HANA runtime aligned to Clean Core principles and released interfaces
Dual use	Identical capture serves event-driven integration and continuous analytics, lake and AI feeds

Table 2. Capability overview. Details depend on release and edition; see product documentation at asapio.com.

Recommendations

An action plan for SAP integration leaders.

- 1 Inventory your capture points.**
 List every mechanism that currently detects SAP change. The count is your architectural debt metric; most organizations find the same objects captured three to five times.
- 2 Adopt application-level CDC as the default.**
 Standardize on business-object events in CloudEvents format as the canonical representation of SAP change, for operational, analytical and AI consumers alike.
- 3 Start with one dual-use object.**
 Publish a high-value object once and connect both a process consumer and an analytics consumer. Measure latency, load and reconciliation effort against the legacy path.
- 4 Retire batch extraction where events suffice.**
 Convert schedule-bound extracts to continuous delta feeds object by object. Keep batch only where full-snapshot semantics are genuinely required.
- 5 Close the loop with inbound events.**
 Where non-SAP systems create SAP documents today via files or point-to-point calls, route them through the same backbone.
- 6 Make it part of your S/4HANA transition plan.**
 Establish the capture layer on ECC now, so feeds carry over and Clean Core compliance is a starting condition on S/4HANA rather than a remediation project.

Conclusion

Application integration and analytics have been treated as separate SAP data problems for two decades; AI has now made the separation untenable. All three consume the same underlying change. Capturing it once — at the application level, code-less and Clean Core-compliant — distributing it as standardized events to any broker, platform or lake, and accepting events back in the other direction, is the simplest architecture that satisfies all of them. This is the design principle behind the **ASAPIO Integration Add-on: the universal integration solution for SAP.**

**ASAPIO**

The universal integration solution for SAP

ASAPIO GmbH · Landsberger Str. 400 · 81241 Munich · Germany
info@asapio.com · asapio.com

About ASAPIO

ASAPIO is a Munich-based SAP software vendor and consulting partner, founded in 2003, with practices in SAP S/4HANA + Ariba, Integration + Technology, and SAP Fieldglass. The ASAPIO Integration Add-on event-enables SAP ECC and SAP S/4HANA for real-time cloud connectivity — code-less and Clean Core-compliant.

Sources

DSAG-Investitionsreport 2026, Deutschsprachige SAP-Anwendergruppe e. V., 26 February 2026 (n = 198, DACH). Press release and charts available free of charge at dsag.de.

This document is provided by ASAPIO GmbH for informational purposes only and does not constitute a promise or legal obligation to deliver software; forward-looking statements involve risks and uncertainties. Third-party survey data is reproduced for illustrative purposes and remains the property of its publisher. SAP, SAP S/4HANA, SAP BTP, SAP Ariba, SAP Fieldglass and other SAP product names are trademarks or registered trademarks of SAP SE in Germany and other countries. Microsoft, Microsoft Fabric, OneLake and Azure are trademarks of the Microsoft group of companies. Apache Kafka is a trademark of the Apache Software Foundation. All other product and company names may be trademarks of their respective owners; their use does not imply affiliation or endorsement.