SAP PRESS

Intelligent Finance with SAP®

AI-Based Automation and Custom Solutions

- Explore SAP's AI capabilities for finance, from journal entry automation to predictive analytics
- Assess key AI tools such as SAP AI Launchpad and Joule
- Design custom AI solutions that incorporate data retrieval, content transformation, smart assistants, and more

Darius Bikulcius

Rheinwerk Publishing

Contents

Chapter 3
Accounts Payable

AI is no longer a separate add-on to finance in SAP; it's becoming part of the way core processes are designed and operated. We've seen the role AI plays in accounts receivable—but what about the other side of the coin, accounts payable?

In the procure-to-pay space, automation shows up very clearly in supplier onboarding, invoice processing, down payment handling, goods receipt/invoice receipt (GR/IR) reconciliation, and the way cash is ultimately managed. SAP Build Process Automation, machine learning (ML) services, situation handling, SAP Invoice Management by OpenText, and SAP Cash Application together form a toolbox that can turn historically manual activities into integrated, insight-driven flows.

In this chapter, we walk through that toolbox from the perspective of accounts payable. We begin with master data, because nothing works reliably without good suppliers and business partners. From there we follow the life of a supplier invoice: how it arrives, how it's processed, how exceptions are resolved, how special cases like down payments are handled, and how mismatches between goods and invoices are cleared. Finally, we look at how all of this affects cash: how quickly you can safely pay suppliers, how confidently you can forecast liquidity, and how reliably you can close the books.

3.1 Master Data

Supplier onboarding is the first place where automation and AI quietly transform finance. Poorly maintained business partner master data shows up later as blocked invoices, payment errors, and reconciliation issues. For many organizations, the reality before automation is a jumble of spreadsheets, emails, and manual entry into multiple SAP and non-SAP systems.

SAP Build Process Automation changes this by providing a way to design end-to-end onboarding flows in which bots, application programming interfaces (APIs), and human tasks all play their part. Standard content from SAP already includes bots that can read Excel templates and create business partners in SAP S/4HANA, update sensitive fields based on employee master data, and check whether essential information such as bank details, payment terms, accounting clerks, and dunning settings is missing. What used to be a manual exercise in data cleansing becomes an automated, repeatable process embedded in day-to-day work.

Imagine a new vendor onboarding wave during an SAP S/4HANA rollout. Instead of keying hundreds or thousands of suppliers directly into the system, the project team uses a governed Excel template. SAP Build Process Automation schedules a job that reads that template, applies validation logic, and calls SAP S/4HANA APIs to create business partners. Failed records are flagged automatically and sent to a master data steward in an SAP Fiori inbox with explanatory messages. The steward corrects only what truly needs human judgment—ambiguous addresses, unclear tax numbers, missing bank details—rather than reentering entire records.

The same pattern applies to ongoing master data maintenance. When the employee responsible for a supplier relationship changes, a bot can confirm and update sensitive business partner details based on HR data. When finance wants to enforce new dunning strategies or payment term policies, a completeness check process can run across the supplier base and generate targeted tasks to fill gaps. Instead of discovering missing data when the first invoice arrives and fails, the system ensures records are usable before they participate in transactions.

From a process design perspective, the onboarding journey can be viewed as a small workflow in itself. Data entry may start in a self-service form for procurement or even for the supplier. SAP Build Process Automation then triggers a series of steps: basic validation of mandatory fields, checks against blacklists or duplicate records, the invocation of a master data check bot, and finally the creation of the supplier in SAP S/4HANA. At each stage, exceptions are routed to the right role. Finance thus moves toward a *first-time-right* approach to supplier data, which pays dividends in every subsequent step of procure-to-pay.

Note

We covered the master data for business partners in detail in Chapter 2 when discussing automations in accounts receivable. The master data process is identical as both use the business partner functionality, so we're keeping that part brief here.

3.2 Supplier Invoices

Once suppliers are set up, the focus shifts to their invoices. Historically, this is where accounts payable teams spend much of their time: registering invoices, manually checking them against purchase orders and goods receipts, releasing blocks, answering status queries from suppliers, and hunting for missing documents. The combination of SAP Build Process Automation and SAP Invoice Management by OpenText fundamentally changes that workload.

Where AI comes in most visibly is at the document capture and validation stage. SAP Invoice Management by OpenText integrates with SAP S/4HANA to ingest invoices from various channels: scanned paper, email attachments, electronic data interchange (EDI) messages, XML documents, or submissions via SAP Business Network and third-party networks.

Invoices are archived, and optical character recognition (OCR) combined with ML extracts header and line item data without the need for manual template training. Over time, the system "learns" the formats of the organization's suppliers and increases extraction accuracy, often to a level where manual validation is only required for low-confidence or high-risk items.

Once the data is captured, business rules perform the traditional checking work. The system determines document type, company code, and vendor; looks up open purchase orders matching the invoice; validates quantities and prices against tolerances; searches for duplicate invoices; enriches missing information where possible; and decides whether the invoice can be posted automatically, needs approval, or must be returned. These rules can be configured to reflect the organization's policies and chart of authority, meaning that exceptions are routed to the right person with all relevant context attached.

SAP Build Process Automation acts as the glue around these capabilities. It can subscribe to events raised by OpenText when an invoice is flagged with a particular exception reason, then instantiate a workflow that assigns the case to a buyer, a cost center manager, or an accounts payable specialist, sets due dates and escalation paths, and tracks resolution time. It can also orchestrate steps outside SAP S/4HANA, such as updating procurement systems, posting messages in collaboration tools, or sending standardized queries back to suppliers. The result is an invoice process in which straight-through processing is maximized and human attention is reserved for genuinely ambiguous or policy-sensitive decisions.

In this section, we'll cover supplier invoice processing with SAP Build Process Automation and embedded AI capabilities in SAP S/4HANA. We'll discuss automation with SAP Invoice Management by OpenText in Section 3.5.

3.2.1 Automating Supplier Invoices Without a Purchase Order

A practical way to automate non-purchase order vendor invoices in an SAP-centered finance landscape is to use SAP's Manage Invoices without Purchase Order template in SAP Build Process Automation. The template is designed to automate the creation of non-purchase order vendor invoices in a transparent, workflow-driven manner, combining an SAPUI5 request form, validation logic, approval routing, and ERP posting into one end-to-end process. It supports invoice submission, tax calculation, ERP validation, approval, and final posting while also giving users visibility into errors, statuses, and approvals.

The process begins with the requester entering invoice information in the provided SAPUI5 application form. The requester captures header details, item details, attachments, and comments, and the submitted form triggers the deployed process through an API trigger in SAP Build Process Automation. The application also includes search help, duplicate invoice checks, and automatic population of dependent values such as payment terms from the vendor master data and due date calculation from the invoice date and payment terms. This means the user experience is not just a static form; it's a controlled invoice-entry layer designed to reduce manual error before the workflow even begins.

Technically, the solution depends on a broader SAP Business Technology Platform (SAP BTP) setup. The prerequisites are as follows:

- SAP Build Process Automation
- SAP Build Work Zone
- SAP Cloud Identity Services
- SAP Integration Suite
- SAP Connectivity service with the cloud connector
- Cloud Foundry runtime
- SAP Business Application Studio

In addition, the process relies on integration content in SAP Integration Suite to call SAP ERP through remote function call (RFC)– or Business Application Programming Interface (BAPI)–based integration flows for tax calculation, validation, and invoice posting. In other words, SAP Build Process Automation provides the workflow and decision framework, while SAP Integration Suite and the cloud connector connect the process securely to the ERP backend.

The configuration starts in the SAP BTP cockpit, where destinations must be defined. The solution requires a `CPI` HTTP destination for SAP Integration Suite, an RFC destination for SAP, and an email destination for notifications. The cloud connector must also expose the required backend BAPIs and RFCs, including services for tax calculation, invoice checking, posting, vendor master lookups, exchange rates, and organizational master data. These backend calls are essential because the process does not merely collect invoice details; it actively validates and enriches them against ERP data before routing them onward.

Once the platform setup is complete, the implementation team imports the Manage Invoices Without Purchase Order package from the SAP Build store. The **Project Type** field should be filtered to **Process Automation**, after which the template can be added from the store to the lobby. The main process, Manage Non PO Vendor Invoice, then can be used and adapted. The process itself doesn't require explicit configuration to run, but the forms, decisions, email templates, and optional branching logic can all be modified to fit local policies and approval requirements.

A key part of the design lies in the delivered decisions. The template includes three decision artifacts: Determine Item Group, Determine Approvers, and Validate Invoice Items. These decisions allow the organization to externalize business rules from the core process. For example, invoice items can be grouped for approval by cost object, approvers can be assigned based on invoice details, and invoice line items can be validated against business-specific conditions such as company code, currency, plant, or cost center. This is especially important in finance because it means the approval logic can evolve without rewriting the overall process flow.

The process itself follows a clear sequence. First, the submitted invoice data is prepared for tax calculation and sent to ERP-related services. Next, the invoice is checked through ERP validation and optionally through decision-based validation rules. If any validation fails,

the requester is notified with a corresponding error form or email, such as failed tax calculation, failed ERP validation, or invalid invoice notification. If the invoice passes validation, then the process prepares the approval data, determines the approver, and sends an approval task to the responsible person through My Inbox.

Approval is handled through an item-level approval subprocess. The responsible approver receives the invoice details, can review attachments and comments, and can either approve or reject the request. Once approved, the process records the approval status and moves forward; on rejection, it ends with an appropriate notification to the requester and approvers. After all approvals are complete, a final automation prepares the invoice for posting and then posts it to the ERP system, after which the requester is informed of the result and the invoice number via email. This makes the template particularly useful for non-purchase order invoices, for which policy control and approval routing are often more important than in tightly matched purchase order scenarios.

The template also includes a set of automations that perform the heavy lifting behind the scenes. These include automations to prepare invoice data, fetch tax calculations, validate the invoice in the ERP system, prepare approval data, prepare item groups, and post the invoice to the ERP system. SAP explains that these automations use custom script logic in JavaScript and web service calls through destinations, with comments included in the automation steps to help implementers understand and adapt the logic. This is a useful design choice because it makes the package both reusable and extensible for organization-specific invoice requirements.

To make the solution operational for business users, the project must be released and deployed, with destination environment variables mapped to the real SAP BTP destinations during deployment. If this destination mapping isn't completed correctly, then the process will enter an erroneous state when it attempts to call external systems. After deployment, the SAPUI5 application can be imported into SAP Business Application Studio, adjusted with the correct process definition ID, built, and then deployed for access through SAP Build Work Zone. This gives business users a governed launch point from which to submit and track non-purchase order vendor invoices.

From a finance transformation perspective, this template shows how SAP Build Process Automation can turn a traditionally manual non-purchase order invoice process into a structured digital workflow. It combines form-based data capture, ERP-backed validation, configurable decision rules, approval routing, automated posting, and user notifications in one integrated pattern. For organizations looking to improve control and efficiency in non-purchase order invoice processing without building everything from scratch, it provides a strong starting point for a governed side-by-side automation on SAP BTP.

3.2.2 Invoicing with Joule

Companies are now looking into how they use AI assistants in the purchase-to-pay and record-to-report processes. In many finance organizations, the apparent "automation" of procure-to-pay and record-to-report still hide a great deal of manual effort. Operational

buyers click their way through several SAP Fiori apps just to find the right purchase order. General ledger accountants spend time clearing single items one by one, nervously double-checking that they have chosen the correct document and period. Accounts payable consultants jump between supplier invoices, journal entries, payment transactions, and workflow monitors every time a supplier sends an impatient reminder.

Joule, SAP's digital assistant, changes the texture of this work. Instead of navigating the system, users talk to it. They describe what they are trying to achieve in everyday language, and Joule orchestrates the underlying SAP S/4HANA capabilities on their behalf. The core logic, configuration, and controls remain exactly where they belong in SAP S/4HANA, but the user experience becomes conversational rather than transactional. There are three main capabilities that Joule assists with, which we'll discuss in the following sections:

- AI-assisted operational purchasing, which dramatically accelerates the creation and maintenance of purchase orders
- AI-supported processing of supplier invoice reminders, which pulls together fragmented information and standardizes communication with suppliers and approvers
- AI-assisted open item clearing, which applies cash optimally to vendor open items to improve reporting on cash and cash liquidity

Although these capabilities sit in different corners of the finance landscape, they share a common design pattern: a thin AI layer on top of robust standard functionality, governed by the same roles, authorizations, and audit trails as any other SAP S/4HANA process.

Operational Purchasing

The operational purchaser capability is a good starting point because it addresses one of the most basic frustrations in procurement: the time it takes to find and interpret the status of purchase orders and requisitions. In a classical setup, an operational buyer opens the Manage Purchase Orders app, enters a range of filters, executes the search, drills into the relevant items, then repeats the process in the Manage Purchase Requisitions – Professional app for the next question. The mechanics are well understood, but they fragment attention.

With Joule, the buyer stays in a single conversational context. The assistant is authorized just like any other user, typically via business roles such as `SAP_BR_PURCHASER`, `AP_BR_PROJECTMGR_COMMPRJ`, `SAP_BR_INVENTORY_MANAGER`, or `SAP_BR_WAREHOUSE_CLERK`. These roles determine which data the buyer may see and which apps they may ultimately navigate to.

> **Security Note**
>
> Joule never bypasses the existing security model. If a user lacks the `SAP_BR_PURCHASER` role, they won't be able to mass-update purchase order delivery dates; if they don't hold `SAP_BR_GL_ACCOUNTANT`, they cannot clear general ledger items; without the correct accounts payable roles, they can't remove payment blocks or inspect invoice details. Segregation-of-duties rules, workflow approvals, and change logs continue to apply exactly as before.

The conversation usually starts with a broad request. An operational buyer might simply say, “Show me the purchase orders,” and Joule responds with a structured list of orders that the user is authorized to access. The buyer refines the query in natural language: “Show purchase orders with purchasing organization 0001,” or “Show me purchase order 1123456789.” The assistant understands attributes such as purchase order ID, purchasing organization and group, supplier, material, plant, product type group, currency, and delivery status, and it shapes the response accordingly.

Crucially, the user does not need to remember transaction names or arcane filter sets. An operational buyer can ask whether there are any purchase orders currently in approval or request all items for a particular plant and material combination, in words that mirror their business questions. Joule translates that intent into the appropriate queries and returns the result set, from which the user can jump directly into the underlying SAP Fiori apps if more detailed work is required.

The same pattern applies to purchase requisitions. Instead of opening a separate app and defining filters, the buyer asks for requisitions with a given ID, purchasing group, supplier, material, material group, plant, product type group, delivery status, total value, or currency. They might search for all requisitions with overdue status or for requisitions exceeding a certain value in euros. The assistant returns the list, with the possibility of drilling down to individual items such as 12345678/0010.

From a process perspective, this conversational interface doesn’t replace the logic of requisition-to-order conversion, but it makes it far easier to identify which demands should be converted and when. The buyer can rapidly home in on overdue requisitions for a particular material group and plant, then open the professional requisition app directly from the result set to create the corresponding purchase orders using standard SAP S/4HANA functionality. Joule acts as the intelligent concierge, steering buyers to the right objects at the right time.

As organizations gain confidence with this way of working, they start to view the operational purchaser as the first step toward automated purchase order creation. Its automation is not in the sense of a black box that posts purchase orders without human approval; rather, it removes the friction between demand identification and purchase order creation.

Example

Consider the daily routine of a category buyer responsible for IT hardware. Each morning, they might ask Joule to show requisitions for the IT hardware material group in plant 1234, limited to those with overdue status and a total value above a certain threshold. The assistant assembles this view across the relevant purchasing groups and suppliers. The buyer scans the list, decides which requirements should be converted, and then, with a click, enters the standard conversion flow in the Manage Purchase Requisitions – Professional app. The technical act of finding and filtering the data, the part that used to consume the most time, is now automated.

Over time, the same pattern can be extended. Buyers can ask for requisitions grouped by supplier to support consolidation of orders or by purchasing group to balance workload. They can start their day with a conversational worklist that immediately shows where action is needed. The automation therefore isn't bypassing decision-making; it's ensuring that those decisions are applied quickly and consistently across all relevant documents.

Once purchase orders exist, they rarely stay static. Delivery dates shift, logistics providers report delays, and suppliers request changes across multiple lines. Traditionally, updating those dates meant opening each purchase order or schedule line and editing them individually. The Change Purchase Orders capability addresses this by allowing purchasers to drive mass updates through Joule.

This capability focuses on the schedule lines of purchase order items. Joule helps purchasers determine which schedule lines are affected and then updates the delivery dates for all of them in a single action. The precondition is straightforward: The user must have the `SAP_BR_PURCHASER` role, and the relevant SAP Note (3527180) should be implemented in the system.

The conversation often begins with a description of the problem rather than a technical filter. A buyer might say "Update the delivery date for purchase order 12345678" when a specific supplier notifies a delay. Joule responds by analyzing the order, determining the schedule lines, and showing the count of affected items. The buyer then confirms the new date, and the assistant updates every schedule line of that purchase order item accordingly.

In a more complex scenario, a freight issue might affect several purchase orders across a company code or for a particular supplier. Here the buyer asks to update the delivery date for multiple purchase orders. Joule interprets attributes such as company code, supplier, current delivery date, new delivery date, and (optionally) material. It returns the relevant schedule lines and, once the buyer approves, applies the new dates en masse. For cross-company events or supplier-wide delays, this capability removes hours of manual adjustment.

From a finance perspective, the impact is significant. Delivery dates drive goods receipt timing, accruals, and the behavior of GR/IR clearing. Having a conversational, traceable way of applying consistent date changes reduces the risk that some orders will be forgotten or that dates will be adjusted inconsistently. Joule does not change the underlying posting rules; it simply ensures that the master data that feeds those rules is easier to keep accurate.

Process Invoice Reminders

If operational purchasing is about keeping internal records clean, then the process invoice reminders capability addresses an outward-facing pain point: the constant stream of reminders from suppliers asking why their invoices have not been paid. For the accounts payable function, each reminder can trigger a small detective story. The consultant must verify whether the invoice exists in the system, whether it has been posted correctly, whether a payment has been initiated, whether any blocks are in place, and whether the invoice is stuck in workflow approval.

Without Joule, this means jumping between supplier invoice apps, general ledger line item displays, payment transactions, and workflow tools. With Joule, the entire investigation lives in a single conversation. Accounts payable consultants, who typically hold roles such as `SAP_BR_AP_ACCOUNTANT` or `SAP_BR_AP_ACCOUNTANT_PROCUREMT`, as well as general ledger accountants with the `SAP_BR_GL_ACCOUNTANT` role, can use this capability once the relevant SAP Note (3537323) is implemented.

The starting point is usually the supplier's own reference. A consultant might ask, "Show invoices with reference 123456789 from supplier XYZ." Joule searches for invoices based on the combination of reference and supplier name, returning any matches. The user can then request header details for a particular invoice, again using natural language: "Display details for invoice 123456789 from supplier XYZ." Behind the scenes, Joule accesses the same invoice data and apps that the consultant would normally navigate to manually.

Once the consultant has confirmed that the invoice exists, the next question is how it has been posted and whether it has been cleared. At this point, they can ask Joule to display the line items for the relevant journal entry or to show the clearing information for a particular journal entry and line item. The assistant responds with the appropriate line item views, revealing whether the invoice has already been cleared by a payment, which payment document was involved, and on what date.

If the invoice has not yet been paid, the consultant needs to understand why. Joule can show payment details for a given journal entry and line item, including the company code and fiscal year context. It can check whether there are any payment blocks on the invoice, again by referring to those same attributes, and it can remove a block when the user has the authority to do so. A short sequence of prompts is often enough: first "Check payment blocks," then, once the cause is known, "Remove payment block."

Workflow status is another common bottleneck. The consultant can ask whether a workflow is active for a given invoice number, and, if so, request the details of the approver. Joule returns the name of the approver and the relevant fiscal year context, making it immediately clear who needs to act.

What makes the process invoice reminders capability particularly attractive is that it doesn't stop at data retrieval. After working through the investigation conversationally, the consultant often needs to respond both to the supplier and, if a workflow is involved, to the internal approver. Joule assists by drafting those messages based on the steps that have been taken and the current status of the invoice. For example, once the consultant has removed a payment block and confirmed that the invoice will be included in the next payment run, they can instruct Joule, "Write an email to the supplier with a summary of what was done so far." The assistant uses the context of the conversation—invoice reference, supplier, payment status, and any changes applied—to generate a concise explanation that can be pasted into an email client. Likewise, the consultant can request a reminder email be sent to the workflow approver that summarizes the findings and politely asks for an update.

The effect is a subtle but important standardization of external communication. Instead of each consultant composing their own narrative, the message reflects the actual system status and uses a consistent structure. This is particularly valuable in large shared services centers, where dozens of people handle supplier queries across many company codes and regions.

From the supplier's perspective, the quality of answers improves. Rather than vague responses such as "your invoice is in process," they receive a concrete explanation: The invoice has been posted, it's awaiting approval from a named individual, a previous payment block has been removed, or a payment has been scheduled for a specified date. The internal process may not have changed, but the transparency around it has.

Open Item Clearing

Joule assistants can allow for the automated clearing of single general ledger open items, which transforms a tedious manual chore into a guided, reversible action.

While operational purchasing lives at the front end of the procure-to-pay process, general ledger open item clearing sits much closer to the record-to-report cycle. Despite all the automation in payment runs and reconciliation, there remain countless situations in which a single open item needs to be cleared manually: temporary postings, suspense accounts, corrections, technical clearing accounts, or late adjustments.

Traditionally, clearing a single item required navigating to the Clear G/L Accounts – Manual Clearing app, filtering the open items, selecting the correct line, and then working through a clearing transaction. If the accountant mis-keys the offset account or posting date, they must locate the clearing document and reverse it. None of this is conceptually difficult, but it's easy to make an error under pressure.

The clearing single G/L open item capability provides a conversational layer over this process. The accountant, who must hold the `SAP_BR_GL_ACCOUNTANT` role, still uses the standard open item app to display the list of items for a given general ledger account. However, instead of manually navigating through menus, they initiate clearing via Joule.

Once the list of open items is visible, the accountant selects the relevant item and chooses the **Clear Items** action. Joule asks only for the minimum additional information: the offsetting general ledger account and the posting date. Clearing is intentionally constrained to a single open item in the same currency as the item being cleared, preserving the integrity of the account. If the accountant proposes a posting date that lies earlier than the date of the open item, and if the period is still open, then the system automatically uses the open item's posting date to avoid backdated anomalies.

In natural language, the accountant might say, "Clear accounting document 1001760839 in company code 0001, fiscal year 2025, line item 001, with offset account 113101 and posting date 2025-09-19." Joule interprets this prompt, validates the inputs against authorization and period controls, and executes the clearing using standard posting logic. The result is a normal SAP S/4HANA document with all the usual audit information.

Clearing is one half of the story; errors and changing decisions are the other. Joule therefore also supports resetting a clearing and, where required, resetting and reversing the clearing document. If the accountant realizes that the wrong item was cleared or that the business wants to treat it differently, they can select the cleared document from the list of cleared items, choose the **Reset Clearing** action, and ask Joule to reset the clearing. The assistant removes the clearing information from the entry view, restoring the document to its open state. A simple conversational request such as "Reset clearing of document 101760864 in company code 0001 for fiscal year 2025" is sufficient.

There are cases, however, where the clearing should be not only undone but explicitly reversed with a new posting. For those situations, the accountant can use the reset-and-reverse capability. After selecting the document, they specify the reversal reason, the fiscal period in which the reversal should be recorded, and the reversal posting date in the standard YYYY-MM-DD format. Because these are standard accounting parameters, they remain fully compatible with existing reporting and audit practices. A typical prompt might describe all these details in one sentence, and Joule takes care of orchestrating the technical steps.

What emerges is a pattern that recurs throughout AI-assisted finance: The automation does not change the fundamental accounting principles; it simply lowers the effort required to apply them correctly. General ledger accountants spend less time on navigation and more time on judgement, yet every action they take remains visible and reversible in the system of record.

Data Quality Is Key!

Joule's ability to return meaningful results depends on consistent purchase order, requisition, supplier, and material master data, as well as clear invoice references, structured reversal reasons, and well-maintained payment block codes. Where data is messy or inconsistent, the assistant may still function, but the user will find themselves refining prompts and dealing with edge cases. AI does not eliminate the classic master data challenge; it simply makes the consequences of poor data more visible as the conversation exposes them directly.

From a cash management perspective, faster and more reliable invoice processing has direct impact. Clean master data and automated validation help ensure that liabilities are recognized promptly, that early payment discounts are taken when planned, and that payment runs are based on accurate, up-to-date information. Controllers gain a much clearer picture of upcoming cash outflows and can steer payment strategies more confidently.

3.3 Supplier Downpayment Requests

Supplier down payments sit at the intersection of procurement, treasury, and accounts payable. They appear less frequently than standard invoices but often involve large amounts and have a heightened risk. In many organizations, the process is fragmented:

Purchase orders may contain down payment terms, the treasury may track the promised prepayments in spreadsheets, and accounts payable may manually create down payment requests and postings in SAP. Each step relies on someone remembering what to do.

SAP provides a dedicated automation, implemented via a low-code solution available in SAP Business Accelerator Hub to industrialize this process. This automates the creation of supplier down payment requests and the corresponding postings in SAP S/4HANA, relieving users from navigating multiple transactions and screens. SAP Build Process Automation wraps this technical capability in a business-friendly workflow.

The workflow proceeds as follows:

1. **Initiation of down payment**
   A typical automated down payment process begins with initiation. This can happen when a buyer creates a purchase order with a down payment condition, when a project manager triggers a down payment request via the Post Supplier Down Payments app, or when an external contract is approved that requires an upfront payment. Instead of expecting accounts payable to discover these cases late in the process, the workflow captures them explicitly and starts an approval flow.
2. **Approval routing**
   Approvals are routed according to the organization's delegation-of-authority rules. The treasury may need to confirm that the prepayment aligns with cash forecasts and risk limits. A project or cost center owner may need to confirm that the goods or services warrant such a commitment. SAP Build Process Automation makes it straightforward to design these paths and adjust them as policies evolve, without resorting to custom code.
3. **Down payment request**
   Once approvals are complete, the down payment bot takes over. It logs into SAP S/4HANA or calls appropriate APIs and creates the down payment request, posts the down payment, and sets references on the purchase order and supplier line items so that the final invoice can be cleared correctly. The bot produces a log of all steps taken, which supports both operational monitoring and audits.
4. **Final invoicing**
   When the final invoice arrives, the earlier automation of invoice processing and OpenText comes into play. The invoice is captured and matched not only against the purchase order and goods receipts but also against the existing down payment. Any deviations, such as an invoice lower than anticipated or a missing purchase order reference, are surfaced as exceptions for accounts payable to resolve. The lifecycle of the down payment, from request through posting to final clearance, is therefore visible and controlled.

From a cash management standpoint, this is particularly valuable. Approved down payments are visible early in liquidity planning; treasurers can see committed outflows and can adjust funding strategies accordingly. Because approvals and postings are consistent and traceable, the organization reduces the risk of unauthorized prepayments and strengthens its control environment.

3.4 GR/IR Reconciliation

If supplier onboarding and invoice capture are about getting data entered correctly, then GR/IR reconciliation is about aligning that data with reality before closing the books. The GR/IR clearing account reflects timing differences between goods receipts and invoice receipts. In a perfect world, these would match automatically at the line level. In practice, differences arise from partial deliveries, timing lags, price changes, and simple errors. Clearing these differences has long been a painstaking exercise that especially flares up at month end.

The Reconcile GR/IR app offers ML capabilities embedded in SAP S/4HANA to make intelligent recommendations for exception handling. The ML component allows the user to explore intelligent recommendations for exception handling. Figure 3.1 shows an example of the overview screen.

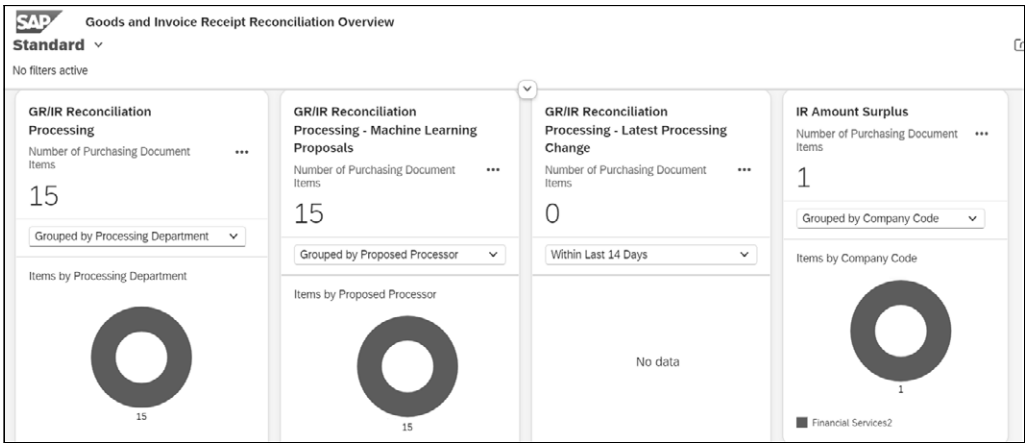


Figure 3.1: Goods and Invoice Receipt Reconciliation App—Overview

Here you can see a good example of the ML proposals. From here, a user can drill down further to perform an account reconciliation, as shown in Figure 3.2.

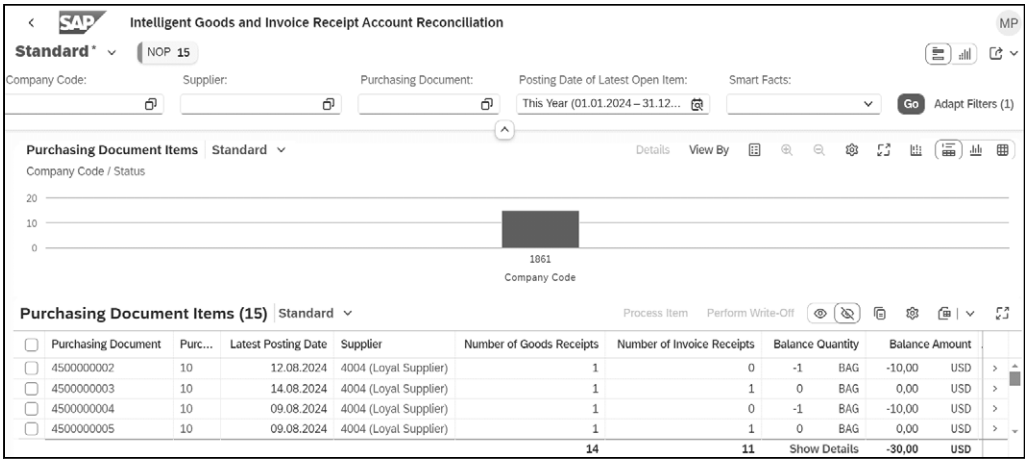


Figure 3.2: Intelligent Goods and Invoice Receipt Account Reconciliation

In this example, you can see the proposed goods receipt and invoice receipt matches, leaving the user to focus on those unmatched items.

3.5 Invoice Automation with OpenText

Although bots and workflows orchestrate much of the process, SAP Invoice Management by OpenText is the backbone for industrial-scale invoice automation in many SAP S/4HANA landscapes. It's tightly integrated with the ERP systems and provides the core services of capturing, archiving, interpreting, validating, and routing invoices.

A workflow is built into SAP Invoice Management by OpenText to handle approvals and exceptions. Documents can be routed to individuals or roles, and the system records who acted on them and when. Approvers can see the archived invoice image side by side with SAP data, compare it with the purchase order and goods receipt, and add annotations or comments. In many cases, the information someone needs to make a decision is consolidated on one screen, reducing the back-and-forth emails that typically accompany invoice disputes.

SAP Build Process Automation adds value by stretching these workflows beyond the boundaries of the core solution. Where SAP Invoice Management by OpenText focuses on what happens inside SAP S/4HANA, SAP Build can incorporate external systems and communication channels. For example, when an invoice requires a purchase order change, an automated process can create a task for the buyer in a procurement tool or even propose the change automatically if policy allows. When documentation from the supplier is missing, a standardized email can be generated and sent without manual effort. When approvals are overdue, escalations can be orchestrated in a consistent way rather than handled ad hoc.

Reporting and analytics round out the picture. SAP Invoice Management by OpenText provides operational views of the process status and service-level agreement (SLA) adherence. Combined with SAP S/4HANA's embedded analytics and SAP Analytics Cloud, organizations can build dashboards that reveal how long invoices take to process, which suppliers tend to generate exceptions, what proportion of invoices flow straight through versus needing a human touch, and where bottlenecks arise. These insights feed back into process improvement: adjusting tolerance levels, refining business rules, training staff on recurring issues, or encouraging suppliers to adopt more structured channels.

Although the solution's primary purpose is to reduce manual effort and speed up processing, its effect on cash and risk is equally important. Faster and more reliable invoice handling allows organizations to take advantage of early payment discounts when it makes sense, enforce payment terms consistently, identify fraudulent or duplicate invoices before payment, and reduce late payment penalties and relationship damage. The accounts payable function shifts from a reactive "post-and-push" role to a proactive guardian of working capital and compliance.

Table 3.1 shows the key capabilities of SAP Invoice Management by OpenText, focused on payables automation.

Capability	Description
Invoice ingestion	■ Support invoice scanning through scanning software that can directly integrate with network scanners, network shares, and so on. ■ Use invoice image-processing capabilities like rotation, contrast, and brightness optimization to improve OCR readability.
Invoice archiving	■ Archive digitized invoices in a secured and compliant repository that supports encryption, compression, and retention features. ■ Archive purchase orders received for sales order posting.
Optical character recognition	■ Extract invoice metadata from the faces of digitized invoices with minimum manual intervention and push the data to SAP for further processing. ■ An OCR solution powered by AI and ML allows you to extract data without training.
Duplicate check	■ Automatically identify duplicate invoices and route those invoices to the duplicate checker role as suspected duplicates. The duplicate checker role can confirm whether an invoice is really a duplicate
Automatic checks and validation	■ Perform automatic checks and validations on invoices based on preconfigured business rules such as three-way matching, among numerous other preconfigured business rules. ■ Route invoices to appropriate responsible persons for issue resolution if any business rules are violated. ■ Send automatic notifications to assigned users for action items pending in their queue.
Invoice approvals and straight-through processing	■ Approve non-purchase order invoices. ■ Automatically approve and pay purchase order-based invoices if three-way matched or if three-way match exceptions are cleared. ■ Automatically pay invoices post approval.
Reporting	Several varieties of reporting are available: ■ Operational and management reporting ■ SLA-based monitoring, aging reports, and the like ■ Reports to monitor bottlenecks and make decisions or take corrective actions to improve efficiency

Table 3.1: Key OpenText Capabilities for Payables Automation

The architecture starts with input channels. Invoices may arrive as scanned images via a scanning solution, as email attachments, as electronic data interchange (EDI) or XML messages, or through electronic invoicing networks such as SAP Business Network or partners like Tungsten. All these channels feed into a central processing engine. Documents are archived in a compliant way, ensuring that an organization can meet its legal retention requirements and that accounts payable can always retrieve the original image.

SAP Invoice Management by OpenText is installed on SAP S/4HANA and tightly integrated with the core ERP system. The following input channels can be used for invoice capture:

- **Scanning**
  Corporate scanners are connected to the OpenText scanning client called Enterprise Scan. This allows for basic image processing before archival in a central archive repository, and it can use barcode recognition capabilities.
- **Email**
  Suppliers send invoices as email attachments to an email address configured with the customer's SAP system. OpenText downloads this email attachment and starts processing it.
- **Other channels**
  Other options include the exchange of invoice data through an EDI channel such as IDocs, ingestion of invoices from SAP Business Network, or through XML and other formats using data mapping. The Tungsten Network ecosystem connects buyers and suppliers for easier, faster, and safer global trade. It facilitates the electronic transmission of invoices to a buyer's accounts payable system. It allows you to upload invoice data directly from your billing systems so you can track invoice delivery to the accounts payable system, eliminating the need for paper invoices.

SAP Invoice Management by OpenText can also store and validate data extracted through OCR. The solution contains approximately 90 standard business rules that businesses can adopt for performing automatic checks and validation on both purchase order-based and non-purchase order-based invoices. The posting is automatic for purchase order-based invoices if three-way matching and other business rules are validated. Users can park, post, and block an invoice for payment and route the invoice to a preconfigured agent for issue resolution. Approval of invoices can be based on a preconfigured delegation of authority depending on organizational hierarchy, cost elements, and other SAP master data.

SAP Invoice Management by OpenText can also capture information using OCR technology. The solution is powered by AI and ML to process various invoice forms. It has a powerful learning engine that trains itself and doesn't need to be trained by personnel. For instance, the solution can be trained to process nonstandard invoice forms. Validation of extracted data becomes optional after a period, as the solution has an accuracy rate of up to 85 percent in data extraction.

Finally, SAP Invoice Management by OpenText has an archiving capability as well. It provides secure and compliant storage for digitized invoices, irrespective of the incoming

channel. The solution can render invoice documents next to SAP data in the navigation pane for easier invoice processing. This means that accountants can view the original purchase order against the vendor invoice to identify any discrepancies. It also provides an annotation feature for reading handwritten information.

SAP Invoice Management by OpenText allows for automation of purchase order-based and non-purchase order-based invoices through standard business rules. Table 3.2 summarizes the key capabilities of those business rules.

Capability	Description
Document type determination	■ It handles the automatic determination of company code, vendor, and purchase order number. ■ Purchase orders from SAP can be synchronized with OCR where OCR tries to extract a purchase order number from the face of the invoice and match it with an existing purchase order number in SAP.
Data enrichment	■ Standard data enrichments exist to enrich the invoice before processing by SAP Invoice Management by OpenText.
Duplicate check	■ It has the ability to flag duplicate invoices based on configurable parameters such as invoice reference number, invoice date, gross amount, and so on.
Exception handling	■ Standard business rules are present out of the box for both purchase order-based and non-purchase order-based invoices for exception handling.
Handling of price/quantity discrepancies	■ The price/quantity discrepancy is one of the most important checks available out of the box in SAP as part of the three-way match process. ■ It has the ability to raise exceptions if there is a price/quantity mismatch and route the invoice to appropriate agent for resolution of the exception. ■ Options exist to cancel the invoice, short pay the invoice, pay as invoiced, cancel and reenter the invoice, create a new purchase order, and change an existing purchase order. ■ For quantity discrepancy, it has the ability to create invoices or a credit memo.

Table 3.2: Key Capabilities of Purchase Order–Based Invoice Automation

Capability	Description
Approval configuration	■ Typically, non-purchase order-based invoices require an account/cost element assignment, which is picked up from the purchase order for purchase order-based invoices. ■ There is a standard role in SAP Invoice Management by OpenText that allows a business user to enter the cost center and general ledger account details. The role is called *coder* and is the first level of approval in the system. ■ Once the coding step is completed, an invoice will be sent for approval based on a preconfigured chart of authority based on a combination of cost elements, company codes, expense type, and organization structure. ■ Importing the delegation of authority from an HRIS is possible through customizations and is covered later.
Posting	■ It offers automated invoice posting. ■ It has the ability to post invoices with payment blocks. ■ It allows a user to route to an agent to cancel the invoice, short pay the invoice, pay as invoiced, cancel and reenter the invoice, create a new purchase order, and change an existing purchase order.

Table 3.2: Key Capabilities of Purchase Order–Based Invoice Automation (Cont.)

3.6 Summary

Seen individually, the automations described in this chapter may look like point solutions: a bot for creating suppliers from Excel, a workflow for invoice exceptions, an ML model for GR/IR, an integration between OpenText and SAP S/4HANA. The real power emerges when you view these as parts of a coherent operating model for SAP S/4HANA Finance.

At the starting point, master data automation ensures that suppliers and business partners are created and maintained with the completeness and consistency that downstream processes need. This reduces avoidable invoice blocks and payment errors. As invoices arrive, AI-driven capture and validation, combined with SAP Build Process Automation workflows, handle most documents without human intervention and route the rest intelligently. Supplier queries, once a drain on accounts payable resources, can be answered automatically based on real-time system data.

Special processes such as supplier down payments no longer rely on expert knowledge embedded in a handful of people. Instead, they follow standardized workflows with approvals, bot-driven postings, and clear traceability. GR/IR reconciliation, historically a month-end pain point, is supported by ML and situation handling so that discrepancies are

identified early and addressed collaboratively. This spreads the workload across the period and improves the quality of financial reporting.

Throughout, SAP Invoice Management by OpenText acts as the industrial engine for invoice processing, while SAP Build Process Automation serves as the orchestration layer that connects bots, APIs, ML services, and human tasks via end-to-end flows. On the receivables side, solutions such as SAP Cash Application apply the same principles—machine learning, automation, and workflow—to the matching of incoming payments with open invoices. Together, these capabilities make both sides of the cash conversion cycle more predictable and transparent.

The cumulative impact on cash management is significant. When supplier onboarding is reliable, invoices are processed quickly and accurately, down payments are controlled, GR/IR balances are kept in check, and customer payments are matched automatically, finance gains a near real-time view of cash positions and commitments. Controllers and treasurers can plan liquidity using hard data rather than approximations. Working capital can be optimized more systematically: deciding when to take early payment discounts, where to negotiate better terms, when to accelerate collections.

Perhaps the most important shift, however, is cultural. As SAP Build Process Automation and AI take over repetitive tasks, finance professionals can spend more time on analysis, stakeholder engagement, and strategic decisions. Automation is not only about speed and cost; it is about elevating the role of finance within the enterprise. In an SAP S/4HANA environment that embraces intelligent technologies across master data, invoices, down payments, GR/IR reconciliation, and cash, the finance organization moves closer to its long-stated goal: to be a true business partner, supported by a digital backbone that quietly does the heavy lifting in the background.

In the next chapter, we'll cover automations in the area of fixed asset processing.

Chapter 12

Turning Static Outputs into Dynamic Stakeholder Narratives

Financial value is not created by publishing one more static pack. It's created when trusted numbers are recast for the people who must act on them. This chapter examines how AI can turn management reports, budget decks, board materials, and investor relations content into dynamic, evidence-based narratives while preserving governance.

In this chapter, we'll show how a report, a budget deck, a board pack, or an investor relations (IR) brief can be treated not as an end product but as a dynamic narrative asset. We'll focus on the following custom solution patterns: additional context, market intelligence, real-time KPIs, and sentiment and peer analysis.

Figure 12.1 offers a conceptual lens for this chapter. Static financial outputs aren't wrong; they're simply incomplete. The missing element is the ability to layer numerical, business, and narrative context around the governed core.

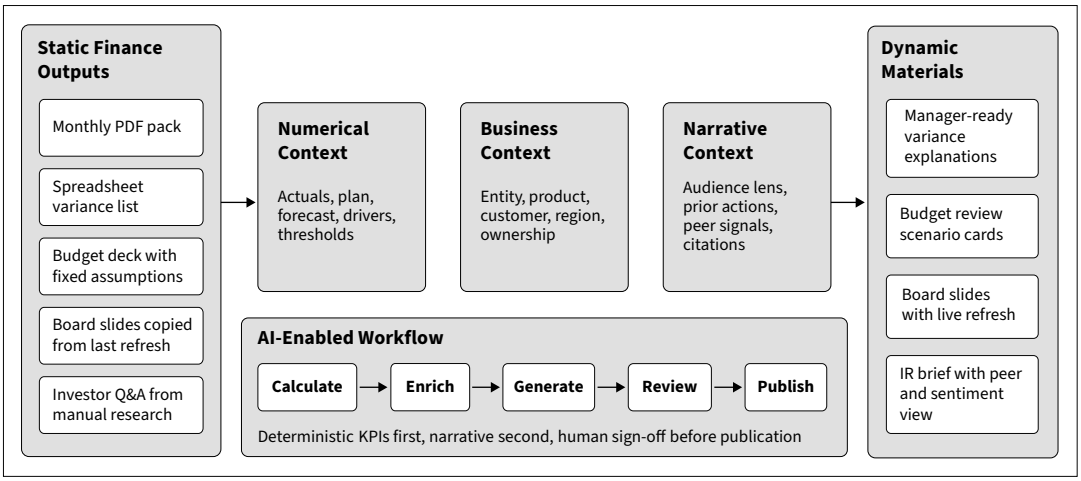


Figure 12.1: AI Turns Static Finance Outputs into Dynamic Narrative Assets

Composite Case Study

Throughout this chapter, we use a composite case study of Aurelia Industrial Group to show how the patterns work in practice. Aurelia is a publicly listed industrial manufacturer with global operations, volatile commodity exposure, and a service business that

stabilizes cash flow. Like many organizations, it already has trusted finance processes but struggles with commentary quality, market-aware planning, late board refreshes, and fragmented investor research. The examples are illustrative, but the design principles are directly reusable.

Aurelia Industrial Group is a fictional enterprise assembled from common finance transformation patterns seen in manufacturing, consumer products, and capital-intensive industries. All figures, timelines, and outcomes in the case study are illustrative.

The technology combinations discussed in this chapter are grounded in publicly documented product capabilities, but the operating model and metrics shown here are the chapter's own design example rather than a vendor reference implementation.

For additional context to set the stage for this chapter, Table 12.1 shows how the same financial truth is recast for different stakeholder materials, time horizons, artificial intelligence (AI) contributions, and human owners.

Stakeholder Material	Primary Question	Time Horizon	AI Contribution	Human Owner
Management report	Why did performance move?	Week/month	Driver explanation, action draft, issue clustering	Financial planning and analysis (FP&A) manager
Budget review	Which assumptions now look weak or strong?	Quarter/year	Market signal retrieval, scenario cards, assumption challenge	Planning lead
Board deck	What changed since the last cut of the pack?	Month/quarter	Refresh narrative, exception summaries, red flag detection	Chief financial officer (CFO) office
IR brief	How will the market interpret our message versus peers?	Quarter/event	Sentiment analysis, peer comparison, question-and-answer (Q&A) preparation	IR lead with legal review

Table 12.1: Same Financial Truth, Recast for Different Audiences

12.1 Additional Context

Variance reporting is the first place where most finance teams feel the ceiling of static output. A report can tell you that revenue is 3.1 percent below plan in a region, that the margin compressed by 140 basis points, and that cash conversion improved—but numbers alone rarely explain whether the change is temporary, structural, controllable, or already addressed. Managers then spend the first 15 minutes of every review reconstructing context that could have been delivered with the report itself.

In practice, additional context falls into three layers. The first is numerical context: plan, actual, forecast, prior period, materiality thresholds, and driver calculations. The second is business context: hierarchy, product family, customer cluster, plant, salesperson, budget owner, and workflow status. The third is narrative context: what the organization said last month, what actions were agreed upon, what exceptions are considered acceptable, and how the audience wants the story framed. An AI-assisted report becomes genuinely useful only when all three layers are available.

Aurelia Industrial Group: The Reporting Problem

At Aurelia Industrial Group, the monthly management pack had all the classic strengths of a disciplined close: reconciled actuals, a robust planning model, and consistent page structure. The weakness was interpretive speed. Regional finance teams exported data from the planning model, wrote commentary in slide notes or email, and then passed a merged narrative to the FP&A group. By the time the pack reached the executive committee, some explanations were already outdated because freight costs, service backlog, or plant restart issues had changed during the final days of consolidation.

The CFO didn't ask for more commentary. She asked for *better* commentary. The group team needed short explanations that surfaced only the most material drivers, pointed to ownership, and clearly indicated whether a pressure point should reverse, persist, or accelerate. It also needed a way to pull in operational clues, such as warranty cases or production deviations, without forcing corporate FP&A to read plant-level ticket systems line by line.

This is exactly where SAP Analytics Cloud and SAP Databricks create a productive division of labor. SAP Analytics Cloud remains the governed frontend for planning, analytical pages, and workflow. SAP Databricks becomes the workspace for enriching a variance object with broader signals and feature sets that don't belong inside the core planning model.

In this section, we'll explore how two solutions, SAP Analytics Cloud and SAP Databricks, assist in utilizing all three layers via AI capabilities.

12.1.1 SAP Analytics Cloud

We'll begin with a look at SAP Analytics Cloud in this section, including its integrated role in the custom AI solution for additional context and core capabilities. We'll discuss its implementation and benefits alongside SAP Databricks later, in Section 12.1.2.

Integration

SAP Analytics Cloud is well suited to the first two layers of additional context because it already combines planning, analytics, workflow, and presentation in one environment. Stories sit at the center of the SAP Analytics Cloud experience and allow teams to capture insights and visualizations across multiple pages for other users to explore. For planning scenarios, SAP positions the product as an all-in-one application for plans, budgets, forecasts, and reports, with live connectivity, scenario modeling, collaboration, and multistep planning processes.

The reference pattern starts with a governed planning and analytics layer in SAP Analytics Cloud. Here, the team defines the report grain, calculation logic, versions, hierarchies, calendars, and thresholds. Because SAP Analytics Cloud combines planning and analysis, the same environment can hold the management pages, commentary placeholders, calendar tasks, and review workflow. The planning features also support scenario modeling, collaboration, and multistep planning processes, which means the report is already linked to ownership and process state rather than floating as an isolated presentation artifact.

Core Capabilities

In this use case, SAP Analytics Cloud contributes the governed reporting layer that turns a raw variance into a reviewable management explanation before any language is generated. The core capabilities are as follows:

- Planning models, versions, and hierarchies keep actuals, plan, forecast, and prior-period values at a common granularity so that the explanation starts from reconciled finance logic.
- Stories and analytic pages bring tables, charts, filters, and commentary together in one presentation surface, allowing the same variance object to be viewed through different stakeholder lenses.
- Calendars, tasks, and workflow tie each commentary item to an accountable owner, due date, and approval state so that the narrative remains part of the finance process rather than an external slide exercise.
- Comments, collaboration features, and version management preserve approved explanations and reduce copy-forward behavior across close cycles.
- Multiactions and related planning automations can refresh calculations, trigger post-close steps, and prepare the governed input that downstream narrative services summarize.

Design Principle

Keep deterministic logic upstream from the language model. The model should summarize a precomputed variance object, not decide whether a variance exists or whether it's material.

That separation preserves auditability. It also makes the generated commentary easier to review because the reviewer is validating interpretation, emphasis, and wording rather than reperforming finance logic in prose form.

Taken together, these capabilities make SAP Analytics Cloud the system that defines the numerical and business context while preserving control over how management commentary is reviewed and published.

12.1.2 SAP Databricks

Now let's turn our attention to SAP Databricks. We'll cover its role in the custom solution in the following sections, discuss how it's implemented alongside SAP Analytics Cloud, and close with a look at our case study to showcase its benefits.

Integration

SAP Databricks complements the governed SAP Analytics Cloud layer when finance teams need more than structured planning data. Because SAP Databricks is embedded natively in SAP Business Data Cloud, teams can access curated SAP data products directly and combine them with third-party, semistructured, and unstructured data through zero-copy delta sharing. This keeps the enrichment layer close to trusted business data while giving advanced users a pro-code environment for notebooks, SQL, and machine learning (ML).

Core Capabilities

In the additional context pattern, the most relevant SAP Databricks capabilities are the following:

- Zero-copy delta sharing to curated SAP data products so that driver engineering can reuse trusted finance semantics instead of copying data into disconnected marts.
- Notebook, SQL, and Spark workloads for joining finance data with ticket text, shipment exceptions, supplier alerts, service narratives, and other operational signals.
- End-to-end AI and ML tooling so that candidate drivers can be ranked, monitored, and improved over time rather than hardcoded once and forgotten.
- A pro-code environment for advanced data users who need broader joins, feature engineering, and model experimentation that don't belong inside the core planning model.

Implementation

The model of work is straightforward, as depicted in Figure 12.2. SAP Analytics Cloud provides the deterministic report object. SAP Databricks enriches that object and ranks candidate drivers. A generation service then produces short commentary in a predefined schema that can be written back for review. Reviewers don't see a black box; they see a governed explanation with source references, confidence indicators, and an approval state.

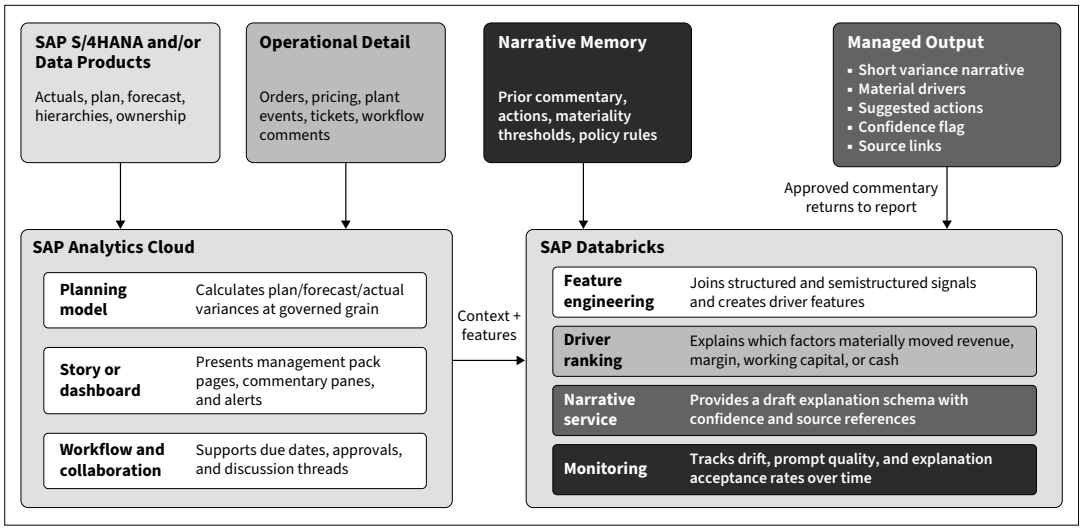


Figure 12.2: Governed Variance-Explanation Pattern Keeps Calculations in SAP Analytics Cloud and Enrichment Logic in SAP Databricks

A practical implementation usually follows these steps:

1. Calculate the base variance object in SAP Analytics Cloud at the granularity required for review—for example, by company, region, business unit, account, and month.
2. Attach business metadata such as accountable owner, materiality threshold, last approved action, and process status from the planning calendar or related workflow tables.
3. Rank candidate drivers using explicit business rules first and statistical or ML signals second so that the final explanation remains interpretable.
4. Generate a draft narrative in a fixed explanation template containing a headline, the major drivers, the outlook, the required action, the confidence level, and the supporting source references.
5. Route the draft back to the appropriate reviewer in SAP Analytics Cloud so that approval, override, and publication remain part of the finance process.

The explanation template, outlined in Table 12.2, matters more than polished prose. It ensures that the report designer, workflow owner, or downstream application receives a consistent set of fields that can be reviewed, approved, and reused.

Field	Purpose	Typical Content
Headline	One-sentence explanation of the variance	Revenue below plan as delayed distributor orders outweighed service outperformance
Top drivers	Two to four quantified contributors	Volume -3.4M; mix -1.2M; price +0.8M; FX -0.5M
Persistence flag	Signal whether issue is likely temporary or structural	Likely temporary pending April shipments
Action	Required management follow-up	Confirm recovery schedule with regional sales and logistics
Confidence	Review prioritization	High/medium/low
Source references	Evidence for review	Order log, ticket cluster, plant downtime note, prior board action

Table 12.2: Useful Template for Machine-Assisted Variance Commentary

Example

Presented in business language, the same variance record would read more naturally as a short review note rather than as raw JSON:

- *Variance reference:* North America Q1 distributor revenue.
- *Headline:* Revenue finished below plan because two March distributor orders shifted into April, while service revenue remained ahead of plan.
- *Primary drivers:* Order timing reduced revenue by USD 3.4 million, product mix reduced it by USD 1.2 million, and price realization added USD 0.8 million.
- *Persistence assessment:* The miss is currently judged temporary, subject to confirmation that the delayed shipments convert in April.
- *Required action and evidence:* Validate shipment dates, refresh the April recovery assumption, and retain the backlog snapshot, sales log, and service revenue bridge as review evidence.

What to Avoid

Don't ask a model to summarize thousands of line items without a controlled reduction step. Commentary quality drops sharply when the prompt contains too much raw detail and too little modeled structure.

Don't let free-text explanations become the only record of a driver. Every important driver should still map back to a reconciled number, a business object, or an explicit source reference.

Benefits

To understand the benefits, let's revisit our case study. Aurelia piloted the pattern first on regional revenue and gross margin commentary. The group FP&A team chose twelve report pages that consumed the most time after close and defined a strict explanation template for each one. SAP Analytics Cloud remained the source of actuals, plan, forecast, hierarchy, and ownership. SAP Databricks was used to enrich the records with order timing, backlog aging, service ticket clusters, and warranty themes. The language layer saw only the curated explanation record, not the full lake of unfiltered data.

The practical change was visible immediately. Instead of writing every paragraph from scratch, regional teams reviewed short draft explanations ranked by confidence. High-confidence items were accepted or lightly edited. Medium-confidence items were escalated when the model surfaced contradictory signals, such as positive volume but negative mix. Low-confidence items were rewritten by hand and fed back into prompt and rule tuning. Even when the text was discarded, the ranked driver object still saved time because the analyst started from a narrowed evidence set rather than an empty page.

Table 12.3 summarizes the pilot month-end bridge and the short explanations produced from the governed variance record.

Metric	Plan	Actual	Variance	AI-Supported Explanation
North America revenue	$154.2M	$149.6M	-$4.6M	Two large distributor shipments slipped into April; service revenue partially offset the gap
Gross margin	31.4%	29.8%	-1.6 pts	Energy and warranty accruals in two plants outweighed positive pricing
Operating expenses	$42.0M	$41.1M	+$0.9M	Hiring freeze and lower travel more than offset digital project spend
Operating cash flow	$158M	$182M	+$24M	Collections in service business and delayed inventory receipts lifted cash

Table 12.3: Simplified Month-End Bridge Used in Aurelia Pilot

Example Commentary

North America revenue finished USD 4.6 million below plan. The largest driver was order timing rather than demand erosion; two March distributor shipments moved into the first week of April because of transport availability. Service revenue remained ahead of plan and partially offset the gap. At this stage, the miss is assessed as temporary, but management should validate April conversion before carrying the full recovery into the next forecast.

12.2 Market Intelligence

Budget reviews often fail for the same reason variance commentary fails: They assume the internal plan is the whole story. In reality, a budget assumption becomes credible only when it's tested against events outside the chart of accounts: Commodity curves move. Competitors change prices. Freight markets tighten. Regulation affects cost or demand. Supplier health deteriorates. If those signals are captured only in disconnected emails, articles, or ad hoc analyst notes, the review process becomes a negotiation of opinions rather than an evidence-based challenge of assumptions.

Market intelligence doesn't mean flooding finance with news. It means selecting external signals that could reasonably alter an internal budget line and presenting them in a way that an owner can evaluate. This is where the pairing of SAP Knowledge Graph and Amazon Bedrock is valuable. SAP Knowledge Graph provides business context by connecting company data and metadata so that AI can understand which internal objects a signal might affect. Amazon Bedrock provides a managed path for retrieval-augmented generation, source attribution, and summarization across private and external content used in the review workflow. We'll discuss both solutions in more detail in the following sections.

12.2.1 SAP Knowledge Graph

SAP Knowledge Graph is a solution that connects a company's data fabric to AI for accurate and contextual responses and grounds AI in the right data to avoid hallucinations. We'll explore its role in the context of market intelligence in this section, and we'll discuss how it's implemented alongside Amazon Bedrock in Section 12.2.2.

Integration

A market-aware budget review starts by changing the unit of analysis. Instead of reviewing budget lines in isolation, the finance team reviews assumption packages. A package might consist of a raw material index assumption, an energy cost assumption, a price realization assumption for a region, or a service attachment rate for a product family. Each package has an owner, a value in the plan, a sensitivity range, and a set of external signals that could move it.

This sounds simple, but it requires semantic alignment. A signal such as "European steel coil prices rise 8 percent over six weeks" isn't directly useful to finance until it's mapped to the plants, products, contracts, and timing windows that matter. A competitor article about discounting in one product category matters only if the category maps to the same buying center or regional sales organization. That mapping layer is where SAP Knowledge Graph earns its place: It preserves the relationships between the company's data objects so that the downstream AI service can reason with context rather than keywords alone.

Core Capabilities

In the market intelligence pattern, SAP Knowledge Graph contributes the semantic layer that tells the system which internal objects a signal could affect. The most relevant capabilities are the following:

- Business object mapping across plants, products, suppliers, contracts, regions, and other entities so that an external signal can be tied to the right exposure points inside the plan.
- Grounding of AI interactions in customer data and metadata so that downstream summarization starts from business context instead of keywords alone.
- Cross-solution access to SAP data and metadata, including prepopulated SAP Knowledge Graph content that reduces the effort required to establish relationships between business objects.
- Natural language access to contextual business information, which helps planning teams understand not only what changed externally but also where that change is likely to land internally.

12.2.2 Amazon Bedrock

Amazon Web Services (AWS) positions the Amazon Bedrock knowledge base as a fully managed way to implement end-to-end retrieval-augmented generation (RAG), including session context management, source attribution, and even natural language-to-SQL access for structured sources. In the following sections, we'll discuss how, together with SAP Knowledge Graph, Amazon Bedrock supports a pragmatic finance pattern: linking market evidence to budget objects before asking a model to write anything.

Integration

Once the mapping provided by SAP Knowledge Graph exists, Amazon Bedrock can retrieve and summarize the most relevant evidence for a given budget topic. The output is not a news digest but a scenario card with three elements: what changed externally, which internal assumptions or objects could be affected, and what action the budget owner should take. In this role, Bedrock acts as a structured challenger to the budget rather than a generic commentator on market headlines.

Core Capabilities

For market intelligence enrichment, the most relevant Amazon Bedrock capabilities are the following:

- Knowledge bases for retrieval-augmented generation so that approved private and external content can be retrieved with source attribution and fed into a constrained summary.

- Choice of foundation models and prompt orchestration so that the same evidence set can be summarized, classified, and converted into scenario cards with a consistent output schema.
- Session context and structured retrieval patterns that help keep an assumption package, its owner, and its evidence set together through the review workflow.
- Enterprise security boundaries for private content, which is important when the research layer is handling licensed sources and sensitive internal planning material.

Implementation with SAP Knowledge Graph

In practice, SAP Knowledge Graph is used to map each assumption package to the business objects that define exposure, while Amazon Bedrock handles retrieval, summarization, and constrained scenario-card generation. A robust implementation usually contains seven design decisions:

1. Define the planning assumptions that are worth external challenge, such as raw material costs, freight, volume growth, price realization, churn, or regulatory cost.
2. Use SAP Knowledge Graph to map each assumption to the internal business objects that determine exposure, such as legal entity, plant, customer cluster, product family, supplier, region, contract type, or channel.
3. Select the external content universe that the organization is licensed and willing to use, including market reports, approved news feeds, analyst summaries, or internal research notes that Amazon Bedrock can retrieve from.
4. Use Amazon Bedrock retrieval to pull only the content relevant to the assumption package, rather than dumping a broad set of articles into the prompt.
5. Require the model to produce a constrained output with a signal summary, the affected assumptions, the likely direction of impact, confidence, and a dated source list.
6. Route the scenario card to the responsible planning owner so the response is recorded as *accept*, *reject*, *monitor*, or *revise* within the planning process.
7. Store both the evidence and the owner response so future planning cycles can learn which signals mattered and which were noise.

Table 12.4 provides a practical taxonomy of the signal families, source types, mapped internal objects, and budget questions used in a market-aware review.

Signal Family	Typical Source	Mapped Internal Objects	Budget Question
Commodity and input costs	Index feeds, procurement notes, supplier alerts	Plant, BOM, contract, inventory policy	Should standard cost or hedge assumptions change?

Table 12.4: Taxonomy for Market Intelligence Signals in Budget Reviews

Signal Family	Typical Source	Mapped Internal Objects	Budget Question
Demand and pricing	Competitor news, market reports, channel updates	Product family, region, key account, sales motion	Do volume or price assumptions still hold?
Macro and foreign exchange (FX)	Economic indicators, central bank commentary	Country, currency, business unit	Should regional growth or FX assumptions be revised?
Regulatory and policy	Industry bodies, legal notes, public announcements	Product line, geography, compliance cost center	Will compliance cost or demand timing change?

Table 12.4: Taxonomy for Market Intelligence Signals in Budget Reviews (Cont.)

Figure 12.3 shows how SAP Knowledge Graph links external signals to internal business objects while Amazon Bedrock turns the retrieved evidence into reviewable scenario cards.

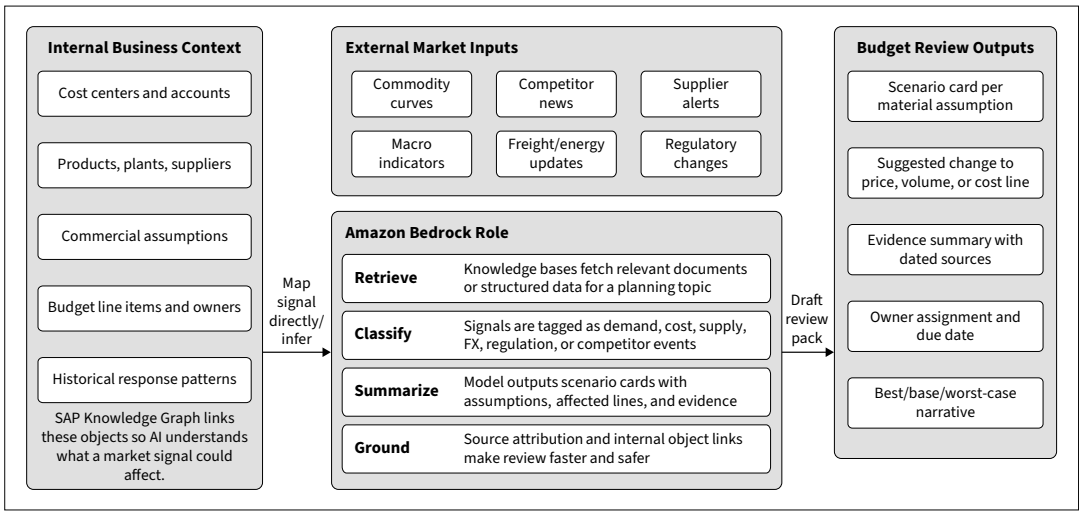


Figure 12.3: SAP Knowledge Graph and Amazon Bedrock Interaction

Controls, Licensing, and Source Discipline

Market-aware planning introduces a different risk profile than internal commentary generation. The main challenge is not just hallucination but evidence hygiene. Finance must know which sources are licensed, what the recency requirement is, how citations are preserved, and whether the same source can be quoted externally or only used for internal decision support.

For that reason, many organizations separate the pipeline into two layers. The first layer retrieves and normalizes source material under approved content policies. The second layer asks the model to summarize only the normalized, approved excerpt set. This reduces the chance that the final scenario card contains unsupported claims, stale evidence, or references to a source the organization cannot redistribute.

In governance terms, the market intelligence layer should behave like a research service feeding the planning process, not like an autonomous planner. The budget owner remains accountable for whether an assumption changes. AI simply makes the challenge process faster, more complete, and more traceable.

Benefits

The benefits are easiest to see in Aurelia's second-half budget review, where the goal was not to read more market material but to force a better planning response.

Aurelia applied the pattern during its second-half budget review. The original plan assumed modest steel inflation, stable freight cost, and no major competitor price disruption in the coatings business. Those assumptions looked reasonable when the draft budget was assembled. By the time of the review, however, three things had changed: Steel pricing had accelerated in Europe, freight availability into one North American corridor had tightened, and a competitor had announced an aggressive promotional program in a segment where Aurelia was trying to recover margin.

The planning team did not want a general market report. It wanted to know which assumptions to challenge and where. The solution was to create assumption packages for steel, freight, and coatings price realization, each linked to the relevant plants, products, suppliers, and regions through the business context layer. Retrieval then focused on those packages. The Amazon Bedrock–generated output highlighted the strongest evidence, mapped it to affected internal objects, and suggested whether the impact should be modeled through cost, volume, price, or timing.

The most valuable aspect of the exercise was not that the model summarized articles faster. It was that the review pack now forced a business response. Each scenario card ended with a recommended action and an owner status: revise now, monitor, or reject. This turned market intelligence from background reading into part of the planning control cycle.

Table 12.5 shows three example scenario cards from the Aurelia budget cycle and the owner response each one triggered.

Scenario Card	Illustrative Evidence	Potential Impact	Owner Response
Steel input cost	European coil prices up over prior review window; supplier notes cite tight availability	Increase cost assumption for two product families and revisit margin floor	Revise now

Table 12.5: Example Budget Review Scenario Cards for Aurelia

Scenario Card	Illustrative Evidence	Potential Impact	Owner Response
Freight corridor pressure	Carrier delays and surcharge notices in North America	Shift shipment timing and temporary logistics cost for April–May	Monitor with weekly check
Competitor discounting	Promotional campaign in coatings segment	Reduce price realization assumption or increase volume risk reserve	Challenge sales assumption before sign-off

Table 12.5: Example Budget Review Scenario Cards for Aurelia (Cont.)

Tip for Finance Design

Use the scenario card as the unit of governance. It's easier to review 10 high-quality cards linked to assumptions than to read 40 pages of generic market commentary.

Each card should name the affected assumption explicitly. If a reviewer can't tell whether the signal changes volume, price, cost, timing, or risk reserve, the card isn't yet useful.

Warning for Executive Use

Don't present market intelligence outputs as if they were external forecasts endorsed by the company. They are structured prompts for management judgment.

Where the source material itself is uncertain or contradictory, the output should state that openly. Ambiguity handled explicitly is more valuable than false precision expressed confidently.

12.3 Real-Time KPIs

Board decks are among the most visible examples of static finance communication. The problem is familiar: The deck is assembled days before the meeting because refresh and validation take time, so the most senior decision-making forum often reviews a carefully designed package that is already partially out of date. Teams then spend the opening minutes of the meeting explaining what changed after the deck was frozen.

A more mature approach is to treat the deck as a governed presentation layer on top of live or near-real-time KPI services. This doesn't imply that every slide should be live. A board deck still contains judgment, narrative framing, and carefully curated exhibits. The goal is more selective: Make the pages most exposed to last-minute movements refreshable, and make the explanatory layer fast enough that the CFO office can reissue a small number of pages on the morning of the meeting instead of rebuilding the pack overnight.

A board-ready KPI isn't merely a measure in a dashboard. It's an information product with five properties:

- The KPI has a settled definition with no ambiguity about the numerator, denominator, timing, or source.
- The KPI is tied to a board question, so the slide exists to support a decision rather than to display data.
- It has a materiality threshold that determines whether commentary is needed.
- It has a refresh cadence and ownership path.
- It has a narrative instruction that tells the language layer how to summarize movement.

Table 12.6 summarizes a compact operating model for board-level KPIs, including the board question, refresh cadence, escalation threshold, and narrative cue attached to each metric.

KPI	Board Question	Refresh Cadence	Escalation Threshold	Narrative Cue
Revenue run-rate	Are we landing the quarter?	Daily in close window	>1.5 percent versus latest outlook	Explain timing versus demand
Gross margin	Is the earnings quality improving?	Daily/weekly	>0.8 points movement	Separate price, mix, cost, one-offs
Operating cash flow	Is liquidity resilient?	Daily	>$15M movement	Explain collections, inventory, payables
Backlog conversion	Will demand turn into revenue?	Twice weekly	>5 percent shift in aged backlog	Flag slippage and concentration risk
Capex delivery	Are strategic projects on plan?	Weekly	>10 percent schedule or spend variance	State milestone and risk to outcome

Table 12.6: Compact Operating Model for Board-Level KPIs

SAP Analytics Cloud is central here because stories are designed to present trusted visualizations and commentary, and the SAP Analytics Cloud add-in for Microsoft PowerPoint allows those widgets to be embedded directly in PowerPoint and refreshed without copying and pasting. If the organization also uses SAP Snowflake for structured and unstructured operational data and Azure OpenAI for narrative generation inside enterprise boundaries, the board pack can move much closer to decision time while remaining controlled.

We'll explore how these tools combine into a custom AI solution for real-time KPIs in more detail in the following sections.

12.3.1 SAP Snowflake

SAP Snowflake provides the semantic and compositional data layer for board-ready KPIs when the reporting view must combine ERP data with operational, customer, service, or other non-SAP context.

Integration

In the overall architecture, SAP Snowflake sits between the source systems and the presentation layer, exposing board-ready semantic views that can be consumed by SAP Analytics Cloud. This is especially useful when a KPI depends on structured and unstructured signals that must be harmonized before reaching the board page.

Core Capabilities

The most relevant SAP Snowflake capabilities in this pattern are semantic views that define consistent metrics and business relationships, Cortex Analyst for governed natural language access to structured data, and Cortex Agents for orchestrating across structured and unstructured data when additional context is needed.

12.3.2 Azure OpenAI

Azure OpenAI supplies the narrative-generation layer for the board-ready KPI pattern. Its role is not to invent a management message but to summarize the latest KPI delta against the prior approved state using a tightly controlled prompt.

Integration

Integration typically occurs after the KPI snapshot is assembled. The service receives the current KPI values, the prior approved cut, the board question, the narrative cue, and any relevant action carryovers, then returns a short draft explanation for human review within enterprise boundaries.

Core Capabilities

Key capabilities include concise narrative generation from structured prompts, support for grounding through enterprise retrieval services when additional context is required, and privacy boundaries suitable for board content as prompts, completions, and training data are not used to retrain the base Azure OpenAI models.

12.3.3 SAP Analytics Cloud

A board member isn't asking for data exhaust. The board wants a small set of measures that answer strategic questions: Are we delivering the quarter? Has risk increased? Is liquidity resilient? Are we converting backlog? Are capex and restructuring actions landing as planned? Those questions remain stable even when underlying data changes, which is precisely why the architecture can be industrialized.

The finance design task is to define a board semantic layer. Every KPI needs a stable definition, owner, refresh cadence, threshold, and narrative rule. Once those are established, the presentation layer can be dynamic without becoming chaotic. The KPI card still looks the same. What changes is the data behind it and the short commentary generated from its current state versus the prior approved state.

SAP Analytics Cloud is the governed presentation layer for real-time board KPIs. It holds the stories, widgets, filters, and presentation flow that board materials require, while the PowerPoint add-in allows the CFO office to refresh selected pages without rebuilding slides.

Integration

Integration centers on SAP Analytics Cloud stories and the SAP Analytics Cloud add-in for Microsoft PowerPoint, which allows teams to find widgets, add them to slides, filter them, and refresh them on demand. In this design, SAP Analytics Cloud consumes the semantic views coming from SAP Snowflake or other trusted models and turns them into board-ready pages.

Core Capabilities

The most relevant SAP Analytics Cloud capabilities in this pattern are story-based presentation, governed widget refresh, planning and collaboration context, and a PowerPoint integration that keeps the board page tied to trusted source content instead of copy-and-paste snapshots.

Implementation

The implementation begins with semantic modeling. Finance and data teams define the board KPIs at a granularity that is both stable and explainable. Some organizations do this directly in SAP Analytics Cloud live models; others expose a semantic layer from SAP Snowflake, especially when the KPI depends on service, customer, or operational data beyond the ERP core. SAP Snowflake's newer integration patterns also emphasize zero-copy, bidirectional access to semantically rich SAP data products, which is helpful when the organization wants to avoid yet another copied board mart.

Once the semantic layer is stable, the team designs the board presentation. SAP Analytics Cloud stories are a strong fit because they are already built to combine tables, charts, filters, and comments into a presentation flow. With the PowerPoint add-in, the team can embed selected widgets directly into the board deck and refresh them as the data changes. This eliminates the copy-and-paste failure mode in which the chart and the commentary diverge because one was updated and the other was not.

The final layer is narrative generation. Azure OpenAI is often chosen here because enterprises want controlled deployment, integration with Azure AI Search or other grounding services, and documented privacy boundaries (Azure OpenAI doesn't use customer data to retrain models). The prompt itself should be narrow: Provide the latest KPI snapshot, the

prior approved state, the board question, the narrative cue, and any relevant action carry-overs. The output should be limited to one paragraph and one risk flag.

Figure 12.4 shows how SAP Snowflake semantic views, SAP Analytics Cloud widgets, and Azure OpenAI narrative generation fit together in a governed board-reporting flow.

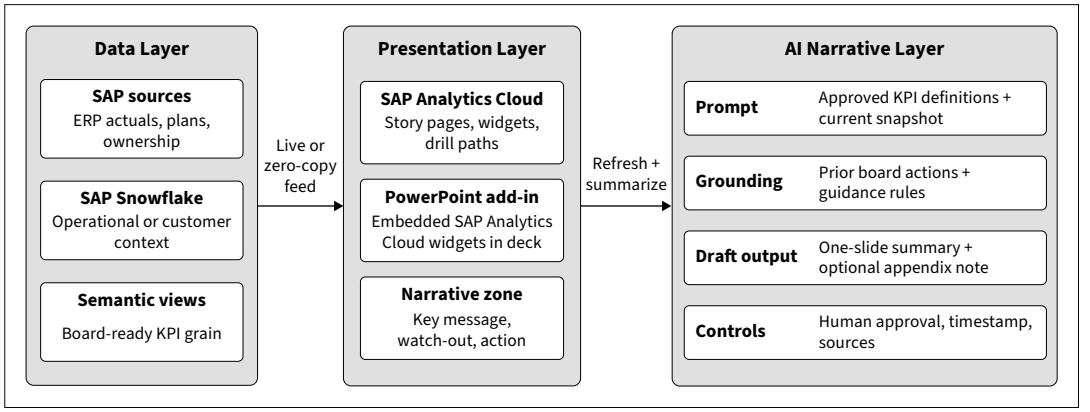


Figure 12.4: KPI Widgets, Semantic Views, and Narrative Generation Designed as Governed Flow

Control Framework and Failure Modes

The main failure mode in dynamic board reporting isn't technical latency; it's governance slippage. If a slide can refresh, then the organization must know who can refresh it, how the prior state is preserved, whether the narrative is regenerated automatically or on request, and where the latest timestamp appears. None of that is difficult, but it must be designed intentionally.

A second failure mode is semantic drift. If one region changes a KPI definition or if the board cue changes but the prompt isn't updated, then the narrative becomes inconsistent. That's why the KPI operating model should be managed like a product catalog with defined stewardship.

A third failure mode is overautomation: Board members aren't asking for a chatbot in the deck. They're asking for relevance. The right implementation feels boring in the best possible sense: current data, concise explanation, consistent structure, and obvious accountability.

Benefits

Before redesigning, Aurelia froze most of its board pack on Friday for a Monday meeting. That created a familiar tension. The chief executive officer (CEO) wanted the latest order intake, backlog conversion, and cash view; the CFO office wanted stability. The result was a hybrid compromise: Some numbers were updated manually in the notes while others remained as they were in the original cut. Board members then asked which figures were current and which belonged to the previous version.

The new pattern separated stable narrative from refreshable evidence. The board pack still had an approved storyline, but five slides were rebuilt around embedded SAP Analytics Cloud widgets. Those widgets drew from board-ready views that combined SAP and non-SAP data. On the morning of the meeting, the CFO office refreshed only those pages. Azure OpenAI produced a short commentary delta for each refreshed page, explicitly stating what had changed since the prior approved version and whether the movement altered the recommendation.

This reduced not only manual work but also ambiguity. As Figure 12.5 illustrates, the board no longer saw a mixed document in which part of the pack was live only in conversation. Instead, the pack itself showed the current widget, the data timestamp, and a concise narrative that compared the latest state with the previous board cut.

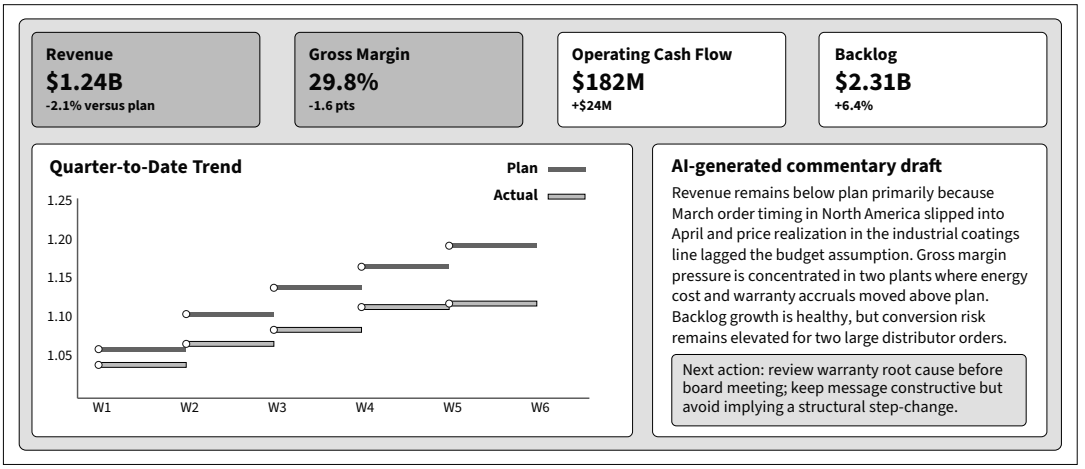


Figure 12.5: Board-Ready Page Combining Refreshed Widgets with Concise AI-Generated Commentary

Board-Ready KPI Snapshot

In practice, it's clearer to describe the semantic view in business terms than to show SQL. A practical board snapshot would usually expose the following fields:

- **Context fields**
  The reporting date and business unit so that every widget is tied to a precise cut of the business.
- **Landing indicators**
  The quarter-to-date revenue and revenue versus the latest outlook so that directors can see whether the quarter is tracking to plan.
- **Quality-of-earnings indicators**
  The gross margin and gross margin versus plan so that the narrative can separate price, mix, cost, and one-off effects.

- **Cash and conversion indicators**
  The operating cash flow and aged backlog so that the deck surfaces liquidity resilience and the risk that demand won't convert cleanly into revenue.
- **Execution indicators**
  The capex spend and capex schedule variance so that strategic projects can be discussed in the same refresh cycle as financial performance.

Board Reporting Note

Real time doesn't mean *unrestricted*. Some board materials should remain intentionally fixed between approval and meeting distribution, especially where legal, audit, or transaction sensitivity is high.

The more practical target is *late-binding freshness*: Refresh the handful of KPI pages most exposed to change, preserve clear timestamps, and require a human sign-off on any regenerated narrative.

12.4 Sentiment and Peer Analysis

Investor relations (IR) content is where internal precision meets external interpretation. The challenge is rarely a lack of internal knowledge. Companies usually know their guidance bridge, operating drivers, and approved messages. The challenge is knowing how those messages will be received in the context of market sentiment, peer commentary, and current investor concerns. Without that external lens, earnings preparation and investor presentations can become internally coherent but externally tone-deaf.

This is why sentiment and peer analysis deserves its own architecture. Internal truth should remain the anchor, ideally through a retrieval-augmented pattern that grounds answers in approved internal material. SAP's own description of RAG for Joule emphasizes context-rich answers and reduced hallucinations by grounding large language models (LLMs) in relevant data. The external market voice can then be layered in through a research platform such as AlphaSense, which markets its next-generation generative search as connecting qualitative insights with structured financial data and producing executive-ready reports with citations.

The result is a better prepared IR team. The team can see which themes are rising in the peer set, which concerns dominate recent coverage, how management's draft message compares with peer emphasis, and which questions are most likely to surface in an earnings call or roadshow.

We'll explore how Joule and AlphaSense work together to enable sentiment and peer analysis in the following sections.

12.4.1 Joule with Retrieval-Augmented Generation

Joule provides the internal grounding layer for IR preparation. In this use case, the priority is not open-ended drafting but retrieval of approved internal messages, guidance bridges, KPI glossaries, and prior script language so that every suggested adjustment starts from the company's own facts.

Integration

Integration begins with a curated internal IR knowledge base: prepared remarks, prior transcripts, guidance bridges, KPI glossaries, board-approved phrases, and disclosure policy. Joule with a retrieval-augmented pattern uses that corpus so that questions are answered against current, approved internal material rather than generic model memory.

This internal-first design ensures that any later comparison with peer or market evidence begins from the company's authorized facts, terminology, and messaging boundaries.

Core Capabilities

In this pattern, the most relevant Joule capabilities are document grounding against approved internal sources, context-rich question answering with reduced hallucination risk, role-aware access to enterprise content, and concise summarization of internal talking points into reviewable drafts with clear traceability back to source material.

12.4.2 AlphaSense

AlphaSense supplies the external market voice layer, giving the IR team a faster way to retrieve peer earnings commentary, filings, research, news, and expert perspectives around a defined theme.

Integration

The design sequence matters. Start with the internal knowledge base: prepared remarks, prior transcripts, guidance bridge, KPI glossary, board-approved phrases, and disclosure policy. This is the material that should ground any internal draft or recommended response. Joule with a retrieval-augmented pattern is well suited here because the question-answering layer can be constrained to approved, current source material.

Only after the internal base is in place should the team add peer and market voice. AlphaSense is useful not because it replaces analyst judgment but because it dramatically improves retrieval across peer transcripts, filings, news, research, and other market content. When this external layer is connected to the internal grounding layer, the output becomes much more practical: "Our message on backlog is internally consistent, but peers are seeing greater investor sensitivity on conversion quality and pricing durability" is a useful finding; "Backlog sentiment is mixed" is not.

A good sentiment architecture, therefore, separates three questions: What can we say, based on internal facts? What is the market saying, based on external evidence? Where do

the two diverge enough that the earnings message or Q&A should be adjusted? Figure 12.6 summarizes this internal-truth-first, external-voice-second pattern.

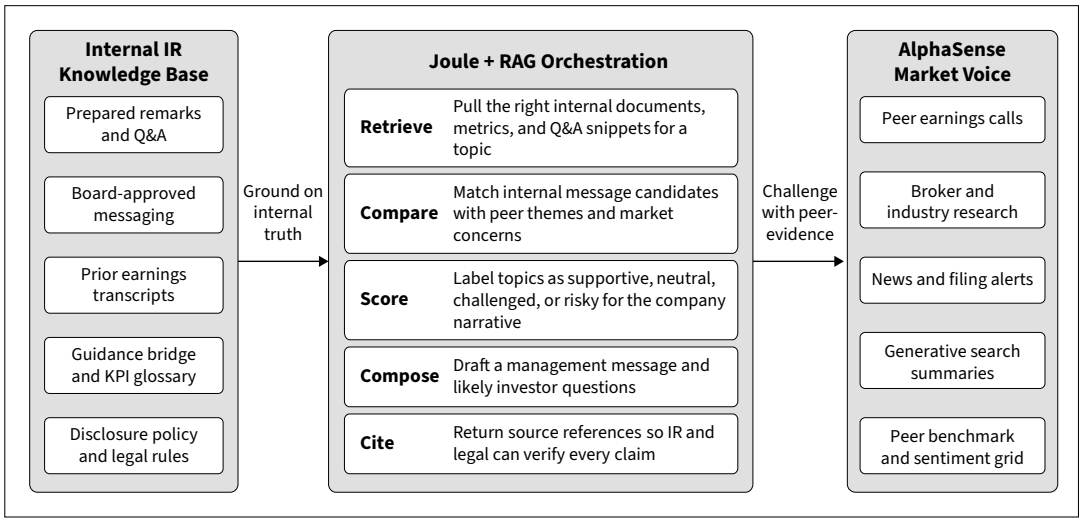


Figure 12.6: Joule with RAG Grounds the Internal Message; AlphaSense Supplies the External Market Voice

Core Capabilities

The most relevant AlphaSense capabilities in this pattern are generative search across a broad market intelligence corpus, AI summaries and sentiment cues over transcripts and research, alerts and monitoring for emerging themes, and the ability to turn findings into executive-ready reports or slides with citations.

These capabilities do not replace analyst judgment; they accelerate retrieval and synthesis so the IR team can spend more time calibrating the message and less time manually stitching sources together.

Implementation

A practical implementation of Joule and AlphaSense together follows a clear sequence: Build the approved internal IR corpus in Joule, define the peer basket and themes to be monitored in AlphaSense, retrieve internal and external evidence separately, compare the two at the theme level, generate a reviewable briefing note, and route the result through finance, legal, and executive review before any wording is approved for market use. That is, follow these steps:

1. Load the approved internal corpus into Joule or the relevant grounding layer so that guidance bridges, prior scripts, KPI definitions, and disclosure policy are available for retrieval.
2. Define the peer basket, themes, and evidence types to be monitored in AlphaSense, such as transcripts, filings, research, news, and expert content.

3. Retrieve internal and external evidence separately, then compare the two at the theme level instead of blending them into one undifferentiated prompt.
4. Generate a constrained briefing output that lists the theme assessment, evidence, likely investor questions, and any recommended message adjustment.
5. Route the output through the normal chain of interpretation, legal review, and executive approval before any external use.

Investor Relations Briefing Template

A practical way to present the same logic is to describe the briefing discipline in plain business language:

- **Purpose**
  Support IR preparation only; the output remains an internal drafting aid and is never a publish-to-market channel.
- **Evidence boundary**
  Use approved internal IR materials and approved external research evidence only, with citations preserved for review.
- **Required assessment**
  For each theme, state whether the market posture is supportive, neutral, challenged, or high-risk, and explain why.
- **Required coaching output**
  List the evidence, the likely investor questions, and the message adjustment that would make management's wording more credible without changing guidance.
- **Themes and audience**
  In this example, the review covers backlog quality, pricing durability, cash generation, and portfolio simplification for buy-side investors ahead of quarterly earnings, using a peer basket of industrial motion control and diversified capital goods companies.

In addition to the solution-focused details, there are some key considerations to keep in mind when implementing sentiment and peer analysis in your organization. Peer analysis fails when the peer basket is treated as a static list. For IR, peers should be grouped by relevance to the question at hand. Product line peers help when discussing demand, margin, or pricing. Capital market peers help when discussing valuation narratives, leverage tolerance, or investor base behavior. Geographic peers help when the company's exposure is region-specific. One company may appear in multiple baskets, but the analysis should state which basket is being used and why.

The IR team should also decide which evidence types matter for each basket. Earnings calls are useful for message comparison. Filings are useful for disclosed risk emphasis and accounting language. Research and news are useful to see how third parties frame the company and its sector. Expert interviews or channel checks, where licensed and appropriate,

may be useful for theme discovery. The key point is that the system should know what it's comparing, not just which company names appear in a watch list.

Table 12.7 shows a theme-based framework that links the internal message anchor to the external evidence the IR team should compare and the question it's trying to answer.

Theme	Internal Message Anchor	Peer/Market Evidence to Compare	What IR Wants to Know
Backlog quality	Aging, conversion profile, concentration	Peer call language, analyst questions, order cancellation commentary	Will investors challenge whether backlog converts cleanly?
Pricing and margin	Price realization bridge, input cost pass-through	Peer margin remarks, discounting themes, sector research	Is our pricing story credible versus the market?
Cash discipline	Working capital actions, capex pacing, collections	Peer cash conversion commentary, balance sheet focus	Will the market reward or dismiss the cash message?
Portfolio actions	Restructuring, disposal, or integration plan	Peer execution examples and market reaction	Which proof points will investors expect next?

Table 12.7: Theme-Based Approach to Peer Analysis

Executive users don't need a black box sentiment score with two decimal places. They need an interpretable view that ties a theme to evidence. A practical output might classify each theme as supportive, neutral, challenged, or high-risk, then list the evidence behind the label. For example, if the market is supportive of cash generation but challenged in terms of pricing durability, the earnings message should reflect that asymmetry rather than aiming for a uniformly optimistic tone.

This approach also avoids a common IR error: overindexing on the loudest recent article or call question. By retrieving a broader and more structured evidence set, the team can see whether a theme is persistent, isolated, emerging, or already fading. The output becomes less emotional and more portfolio-like.

When designed well, the sentiment layer also improves message rehearsal. Management can test draft statements against the current evidence set and ask where the statement is likely to be challenged. The result is better-prepared Q&As, cleaner script revisions, and fewer surprises during the call.

Table 12.8 translates the same method into an illustrative earnings-prep view for Aurelia, showing how internal draft language can be adjusted once market evidence is reviewed.

Theme	Aurelia Draft Message	Market Signal	Assessment	Recommended Adjustment
Backlog	Backlog remains robust and positions us for the second half	Peers increasingly discuss conversion timing and order quality	Challenged	Acknowledge quality and timing, not only volume
Margin recovery	Pricing discipline is restoring margin	Competitor discounting noted in selected segments	Neutral to challenged	Differentiate portfolio pockets where pricing is holding
Cash generation	Cash execution is improving structurally	Peers also emphasize collections and capex restraint	Supportive	Keep the message but avoid implying one quarter proves a trend
Portfolio simplification	Restructuring will sharpen returns	Market rewards proof of milestone delivery	Neutral	Add milestone evidence and timing

Table 12.8: Illustrative Theme-Level Sentiment View for Earnings-Prep Cycle

Governance for Investor-Facing Content

IR use cases demand the highest discipline in the chapter because the output can influence public messaging. For that reason, the sentiment and peer-analysis layer should never be allowed to publish externally. Its purpose is to help humans prepare, not to bypass legal, finance, or executive review.

The most effective governance pattern is to preserve a strict chain: retrieved evidence, generated comparison, human interpretation, legal review, and approved wording. In that chain, AI accelerates the first two steps and improves the third. It doesn't collapse the chain into a single automatic output.

That discipline also makes the use case more credible to executives. Management teams are far more willing to use AI for earnings preparation when they see that the system returns citations, shows both internal and external evidence, and flags uncertainty instead of pretending to possess certainty.

Benefits

In the quarter before Aurelia's earnings call, management wanted to emphasize three points: backlog strength, improving cash performance, and the benefits of portfolio simplification. Internally, all three points were supportable. Externally, however, the peer picture was uneven. Several peers had begun to receive skeptical questions about backlog conversion quality, and research notes were increasingly focused on whether price held up in selected end markets.

Using the internal grounding layer, the IR team first assembled the approved factual base: backlog aging, order concentration, pricing bridge, cash drivers, and restructuring milestones. It then pulled peer evidence through AlphaSense for the defined basket. The combined analysis showed that Aurelia's cash message was likely to land well, but the backlog message needed revision. The original script celebrated backlog volume without addressing conversion quality. That left a gap between internal messaging and current market sensitivity.

The revised script kept the positive backlog story but added two confidence builders: a statement on the share of backlog scheduled within the next two quarters and a note on order quality by channel. It also prepared a Q&A response on competitor discounting—not because Aurelia had a broad pricing problem but because the peer evidence suggested that investors would ask for sharper differentiation.

Example Executive Prep Question

Based on the latest internal guidance bridge and the current peer basket, which two elements of our earnings script are most likely to be challenged by investors, and what evidence-backed adjustment would make the message more credible without changing guidance?

12.5 Summary

The four use cases in this chapter all start from the same premise: Finance communication becomes more valuable when trusted numbers are paired with the right context for the right audience. In management reporting, the priority is explanatory depth and actionability. In budget reviews, it's the disciplined use of market signals to challenge assumptions. In board decks, it's freshness without governance loss. In IR, it's the controlled combination of internal truth and external market voice.

SAP Analytics Cloud provides the natural frontend for many of these patterns because it already combines stories, planning, collaboration, and presentation in one governed environment. SAP Databricks extends that environment when feature engineering and broader data enrichment are required. SAP Knowledge Graph strengthens the semantic layer that grounds AI in business context. SAP Snowflake supports composite KPI and semantic views where SAP and non-SAP data need to meet. Joule with RAG provides a strong internal

grounding pattern. Amazon Bedrock and Azure OpenAI offer managed generative AI capabilities for retrieval, summarization, and narrative generation. AlphaSense strengthens the external market intelligence and peer analysis layer.

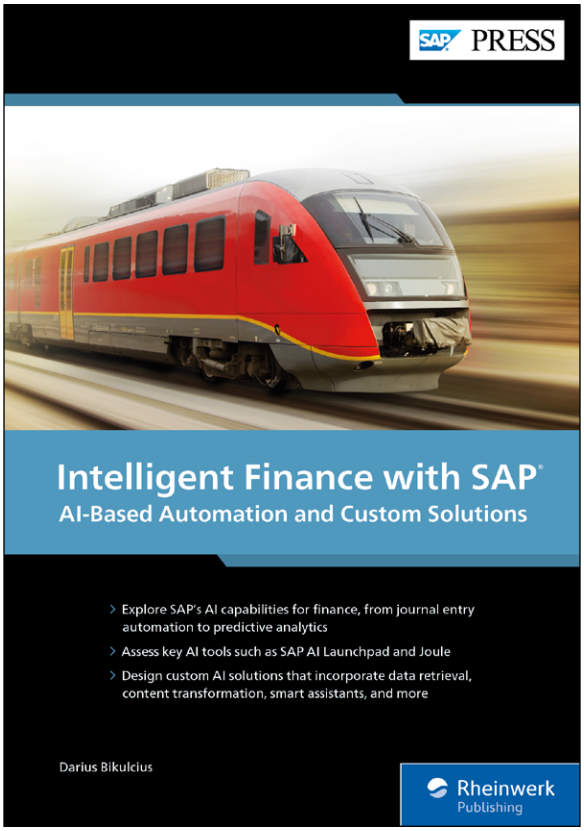
For finance leaders, the practical lesson is not to start with a chatbot. Start with an information product that already matters: a management page, an assumption package, a board KPI card, or an earnings theme brief. Define its granularity, owner, evidence set, and review workflow. Then add AI only where it reduces manual synthesis or improves interpretive speed. That sequence preserves trust.

A final thought is worth emphasizing: Dynamic stakeholder materials are not merely a formatting improvement. They change the role of finance. The finance team moves from being a producer of static output to an orchestrator of decision-ready narratives. That is a significant shift in capability and credibility, and it's one of the most compelling reasons to treat AI as part of the finance communication architecture rather than as a novelty on top of it.

Adoption Checklist

- Choose one high-friction artifact first, such as the monthly variance pack or the five slides in a board deck that are most exposed to late movement.
- Define a strict explanation template and evidence model before introducing generation.
- Keep deterministic calculations, materiality logic, and ownership outside the language model.
- Preserve source attribution and timestamps so reviewers know what changed and why.
- Expand only after the first use case proves both efficiency and trust.

In the next chapter, we'll discuss how AI assistants can be incorporated into your SAP finance workflow.



Darius Bikulcius

Intelligent Finance with SAP

AI-Based Automation and Custom Solutions

- Explore SAP's available AI capabilities for finance, from automating journal entries to incorporating predictive analytics
- Design custom AI solutions that leverage data retrieval, content transformation, smart assistants, and more
- Preview agentic AI as the catalyst for a truly intelligent ERP

www.sap-press.com/6279

The Author
Darius Bikulcius is a managing director in PwC's SAP financials practice with 25 years of experience in financials design and implementation. He specializes in automating processes for invoice management, journal entry management, and month-end close.

ISBN 978-1-4932-2827-0 • 535 pages • 07/2026
E-book: $84.99 • **Print book:** $89.95 • **Bundle:** $99.99

Rheinwerk Publishing