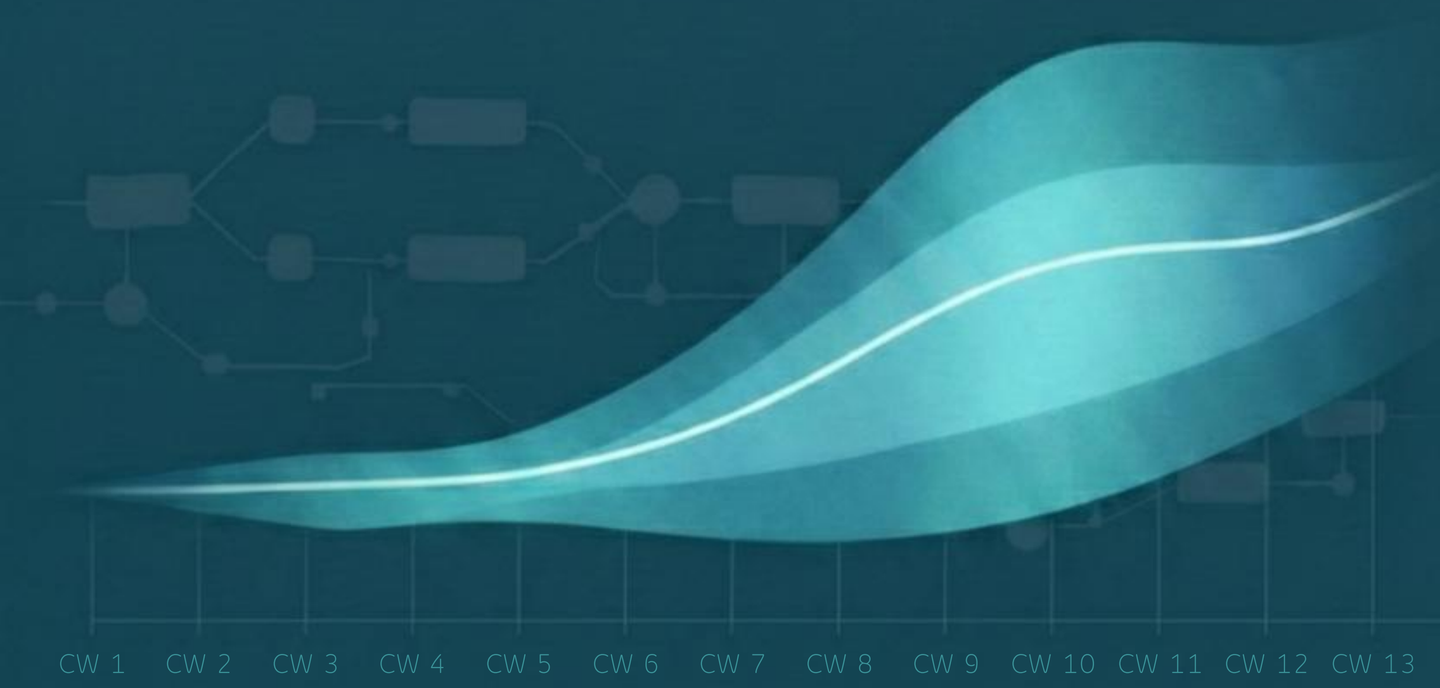


Patterns over Averages

WHITE PAPER

From average values to robust corridors in liquidity planning

June 2026
Axel Meythaler | Pirmin Mutter | Pascal Krawczyk

Contents

Executive Summary

Liquidity planning is one of the **most critical steering instruments** of a company. Despite the increasing digitalization of planning processes, **forecast accuracy** remains unsatisfactory in many companies. The cause rarely lies in the planning tool alone, but in the **assumptions that feed into the forecast**. In particular, the **payment behavior of customers** and **toward suppliers** is often a **driver of inaccuracies**.

The **interest rate turnaround** has forced the **shift** to **active liquidity management**. Every euro held unnecessarily as a **liquidity reserve** creates **opportunity costs**. Every euro the liquidity forecast misses can lead to expensive short-term **emergency financing**.

Process mining and AI methods can help **reduce inaccuracies** and enable planners to output **forecast intervals** that allow **liquidity reserves and credit lines to be managed on a risk-adjusted basis**. **Process mining** can start as early as order intake and use events from order creation, delivery, invoicing, and the approval process as context attributes to make the **drivers of downstream payment behavior visible**. On this data basis, **AI models** can be trained that generate, for each open receivable and payable, a **payment forecast with a probability distribution**.

This makes inflow and outflow dates not only more accurate, but above all more controllable. Companies that implement this approach can **improve forecast quality**, **size liquidity reserves more precisely**, and **reduce financing costs**.

01

Forecast uncertainty in operating cash flow

Many companies have invested in modern planning software in recent years. Process efficiency is rising, but **forecast accuracy** often falls short of expectations.[9]

Beyond process integration, the **forecast assumptions** themselves are a frequent source of error: the **discrepancy between assumed and actual payment behavior** is high in many companies.

Short-term liquidity planning

In the first planning weeks, **short-term liquidity planning** is based on the direct capture of cash-effective inflows and outflows in a rolling forecast. **Liquidity planning** overall follows a multi-stage structure that can be broken down into the following planning components:

- **Liquidity status:** Determination of immediately available liquid funds (in particular cash, bank balances, undrawn credit lines) and comparison with liabilities due as of the reporting date to identify shortfalls or surpluses (also referred to as "pro-forma liquidity"). Items not yet posted (and possibly already due) must also be included.
- **Open items (OPOS):** Integration of open receivables and payables that fall due after the reporting date.

- **Recurring payments:** Inclusion of regularly expected payments not covered by open items (e.g., payroll, loan repayments, rent, taxes).
- **Order backlog and purchase commitments:** Integration of existing, non-invoiced orders and open purchase orders.
- **Expected orders and purchase requisitions:** Inclusion of anticipated incoming orders and purchase requisitions.
- **Management assumptions:** Completion through assumptions derived from individual transactions, based on the available mid-term P&L, expense, and balance sheet planning.

The planning components are shown in Figure 1.

Schematic structure of short-term liquidity planning

Liquidity status

(illustrative values) as of t=0

Item	EUR m
Cash on hand	1,2
Bank balances (available accounts)	6,8
Undrawn credit lines	8,0
Drawn amount	(7,0)
Total available credit lines	15,0
Freely available liquid funds	**16,0**
Liabilities due (as of reporting date)	13,5
Pro-forma liquidity	**2,5**

Forecast

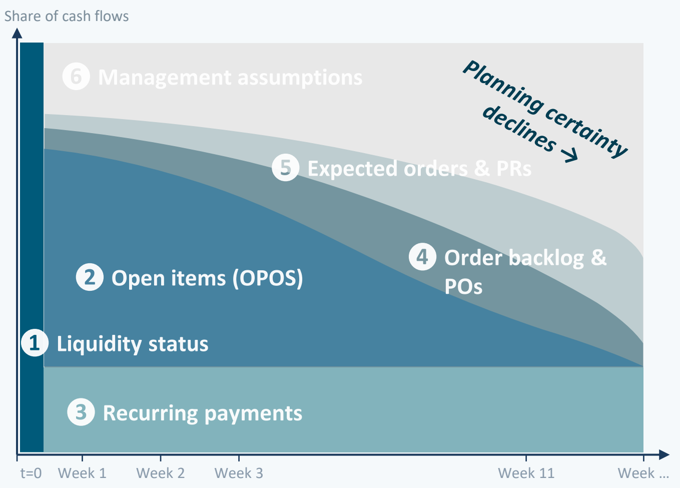


Fig. 1: Structure of short-term liquidity planning. As the time horizon extends, the share of certain inflows and outflows declines (source: own illustration)

Where forecast uncertainty arises

Because short-term liquidity planning provides the basis for day-by-day treasury decisions in the first planning weeks, forecast errors weigh particularly heavily under liquidity pressure. In the supposedly safe payments from open items (OPOS), unacceptable forecast uncertainty can arise.

The reason is that actual payment behavior can deviate significantly from contractually agreed payment terms. This has an immediate impact on results. An example: if a customer with EUR 10m gross monthly revenue and a 30-day contractual payment term shows a **dispersion of ±12 days** in actual payment dates, this corresponds to a **weekly forecast bandwidth of around EUR 4m**. A significant liquidity reserve must be held against this.

In practice, companies try to absorb this uncertainty through manual corrections or blanket safety buffers. Both have their limits. Manual corrections are person-dependent, poorly scalable, and their accuracy is rarely systematically reviewed. Blanket buffers tie up liquidity unnecessarily. On top of this comes a common misconception: the assumption that individual customer deviations cancel each other out at the portfolio level only holds if those deviations are genuinely uncorrelated. Seasonal effects, industry-wide payment cycles, or macroeconomic influences can however systematically push in the same direction and amplify the overall deviation rather than smoothing it.

Tools improve process efficiency and data integration, but do not address the quality of the underlying forecast assumptions. A decisive lever lies in an **empirically grounded data base on real payment behavior**. **Process mining** and **artificial intelligence (AI)** can deliver this data base. The forecasting object of this paper remains the timing of existing receivables and payables in short-term liquidity planning. The underlying data base, however, can extend much further upstream and incorporate signals from sales documents, delivery, invoicing, dunning, or approval processes. Sales risk (whether an expected order materializes) and default probability (whether payment is made at all) can likewise be modeled with related AI methods, but are not addressed in this paper.

“

All models are wrong, but some are useful.

George E. P. Box | Science and Statistics

02

What really drives payment behavior

The factors that determine payment behavior are the key to forecast quality in the short-term planning horizon.

Accounts receivable: Why customers pay differently than agreed

Payment behavior is influenced by temporal, contractual, and structural factors. Temporal patterns (no payment runs on weekends, holiday and vacation effects, differing bank calendars in international groups) are in principle well modelable, but increase the complexity of day-level forecasting.

Contractual payment terms and WAT. The simplest model of payment behavior is based on the contractually agreed payment terms. The weighted metric is referred to as WAT (Weighted Average Terms).[10]

Actual payment behavior as empirical improvement. Using the actual duration from invoice date to cash-effective receipt, referred to as WADTC (Weighted Average Days to Collect),[10] can produce an empirically measurable improvement in forecast quality. However, the metric only represents a weighted average across historical transactions and, when applied to the future, can therefore lead to significant distortions in individual cases.

Figure 2 shows a schematic of the KPI definitions for clarification.

Limits of average values. The higher the dispersion of payment behavior, the less accurately a single average value reflects reality.

Drivers of incoming payments. An incoming payment forecast that is as accurate as possible requires the consideration of additional aspects: for example partial payments, cash discount usage (and its dependence on invoice amount and payment timing), the customer's payment run patterns, invoice issuance timing, current dunning level, and the number of open payment discrepancies.

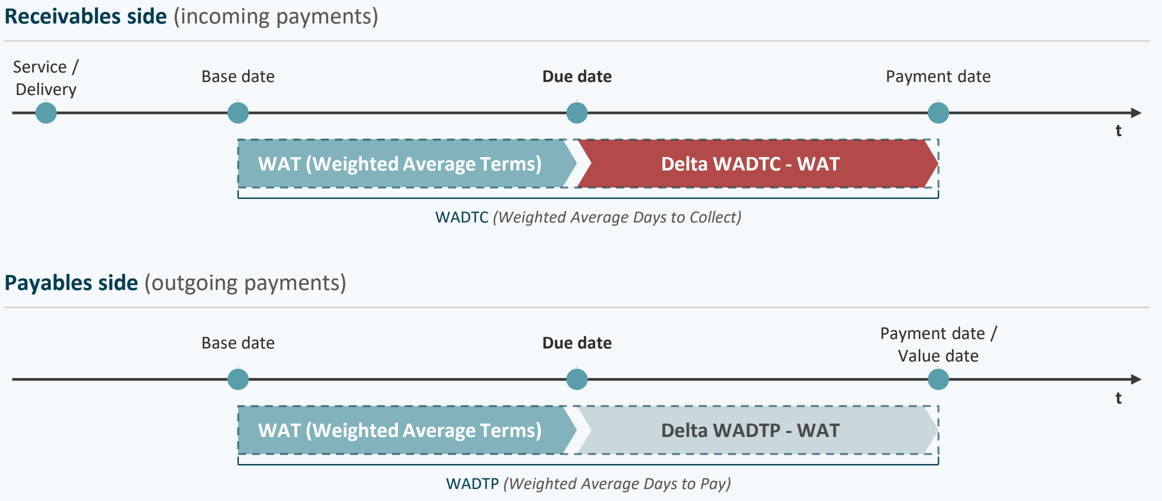


Fig. 2: Operational KPIs along the timeline (source: own illustration)

Structural factors come on top of this: regional payment cultures, industry affiliation, and seasonal effects (e.g., factory holidays, year-end business, budget outflows in Q4) can produce predictable but often unmodeled fluctuations.

Forecast relevance varies by business model. In project- and capex-heavy companies, partial invoicing and large individual transactions generate significant dispersion. In environments with frequent offsets and deductions, lump-sum payments and standardized payment runs systematically distort simple average assumptions. Seasonal business models add recurring but rarely modeled patterns at quarter-ends. In international B2B business, differing bank calendars and regional payment cultures further increase variance.

Accounts payable: Why your own payment behavior can hold surprises

In analogy to the incoming-payment forecast, an outgoing-payment forecast can be optimized based on historical experience values. The WADTP (Weighted Average Days to Pay)[10] measures the weighted average time span from invoice date to the company's own cash-effective payment, weighted by invoice amount.

Forecast quality on the outgoing-payment side should tend to be higher, as outgoing payments are more controllable. Nevertheless, systematic patterns also exist here that often arise unconsciously: invoice approvals stall at certain points in the month (for example due to internal reporting deadlines), payment runs only execute on certain weekdays, and cash discount usage is often more inconsistent than assumed. Many companies know their customers' payment behavior better than their own, because their own outgoing-payment patterns are distributed across different functions, approval levels, and payment runs and are never analyzed in a consolidated way.

Analyzing these patterns has a dual benefit: it improves forecast accuracy and simultaneously uncovers optimization potential (e.g., missed cash discounts due to slow approval processes). However, *relying solely* on historical averages cannot unlock the potential to lift forecast quality.

03

Five maturity levels: Where does your company stand?

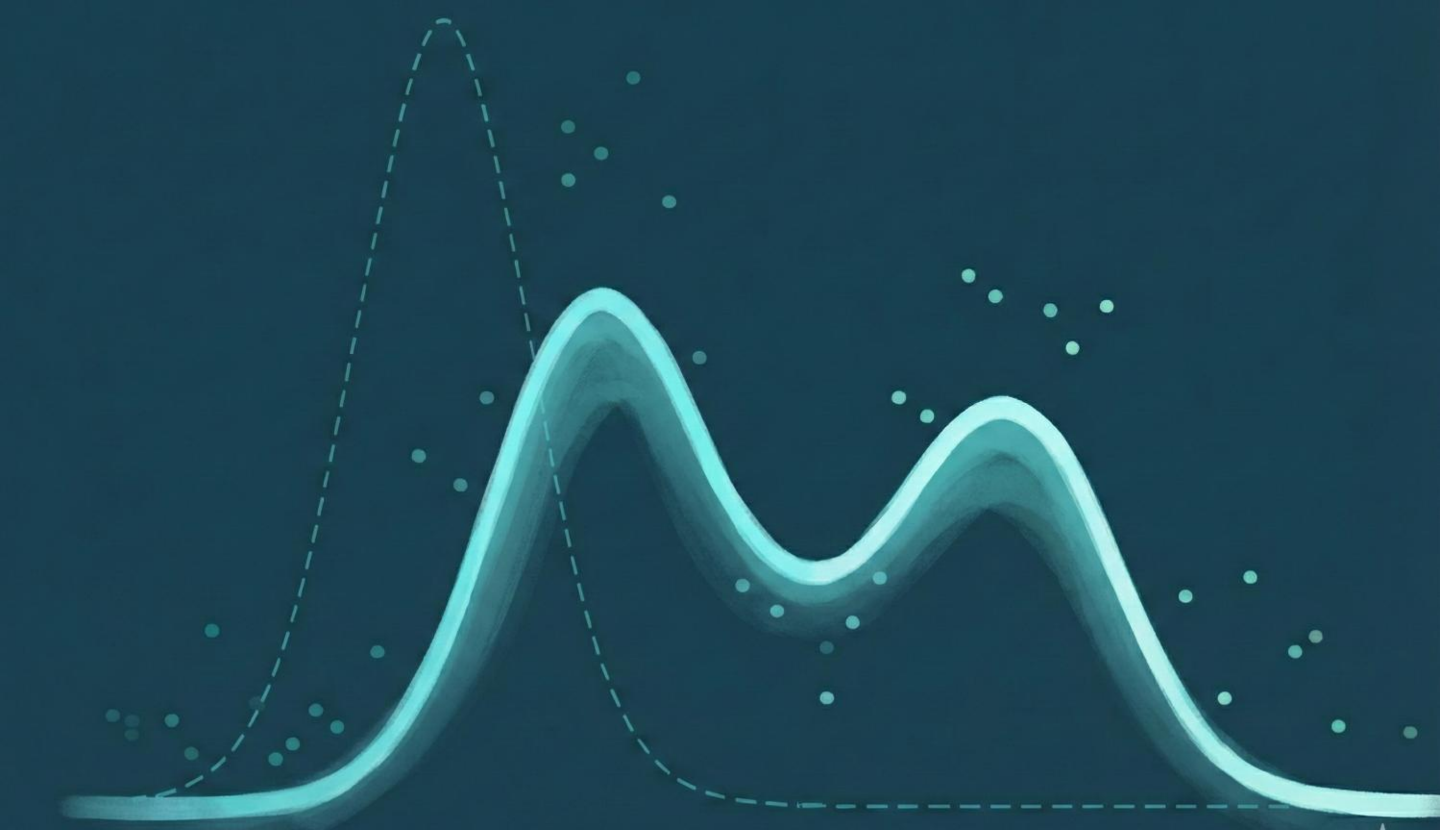
In practice, five maturity levels can be distinguished in liquidity forecasting. This classification helps assess the status quo and define a target path for improvement. The discussion below focuses on the incoming-payment forecast, where the greatest forecasting challenge lies: incoming payments depend on the behavior of external business partners and are not directly controllable. The payables side is conceptually analogous but, in practice, more controllable (see Chapter 2).

Figure 3 classifies the forecasting methodology into five maturity stages.

Level 1: Manual. The liquidity forecast is run predominantly in spreadsheets. Payment dates are planned based on contractual payment terms, without empirical correction. A systematic plan-vs.-actual comparison is either absent or only sporadic. Planning is person-dependent, poorly documented, and fragile when staff change. Typical hallmark: forecast deviations are not systematically analyzed. The causes of plan misses remain unknown.

Level 2: Standardized. The planning process is structured and documented. For the first time, empirically measured payment behavior is used as a correction: the highest-revenue customers receive an individual WADTC, while the remaining accounts are mapped via segment or industry averages. Updates typically take place as part of annual planning. A regular plan-vs.-actual reconciliation is established. Seasonal fluctuations, size-dependent effects, and dispersion within a segment remain invisible.

Level 3: Integrated. The data connection to the ERP system (Enterprise Resource Planning) is automated. Actuals flow into the planning software on a regular basis. The planning process is rolling and scenario-capable. Progress over Level 2 lies in granularity and timeliness: the WADTC is automatically calculated per customer and regularly updated, no longer only for top customers and no longer just once a year. It remains, however, a single-point estimator per customer: a single expected value that compresses the entire variance of payment behavior into one number.



Typical hallmark: planning is system-supported, but the planner regularly corrects "by gut feeling" because a single average value does not adequately reflect reality.

The leap from Level 3 to Level 4 requires a process-mining methodology that goes beyond aggregated ERP metrics.

Level 4: Data-driven. Building on the integrated data base from Level 3, the conceptual leap takes place: from single-point estimator to distribution function. Historical payment behavior is analyzed systematically at the individual transaction level. Process mining is used to extract real payment patterns from the ERP transaction data. Customers are no longer grouped by master data (industry, region, revenue class), but segmented based on their actual payment behavior. Unlike the individual WADTC calculation at Level 3, this segmentation captures not only the mean but the full distribution per cluster. These profiles are refreshed from the current transaction data with every model update.

Typical hallmark: dispersion within a segment is known and quantified for the first time. Already at this stage, forecast intervals can be derived and liquidity reserves can be sized on a data-driven basis rather than as a flat allowance.

Level 5: Predictive. The highest maturity level combines process-mining insights with context-based AI models. For every individual open receivable, an individual payment forecast is generated, drawing simultaneously on historical behavior, seasonal patterns, invoice attributes, and the current process status. The leap over Level 4 is that multivariate modeling captures interactions between factors that a univariate distribution per segment cannot represent. The forecast is probabilistic and delivers confidence intervals. However, this analytical maturity level only unfolds its full steering value when governance, operational responsibility, and operational integration with reserve logic and line management move along with it.

Maturity model of forecasting methodology

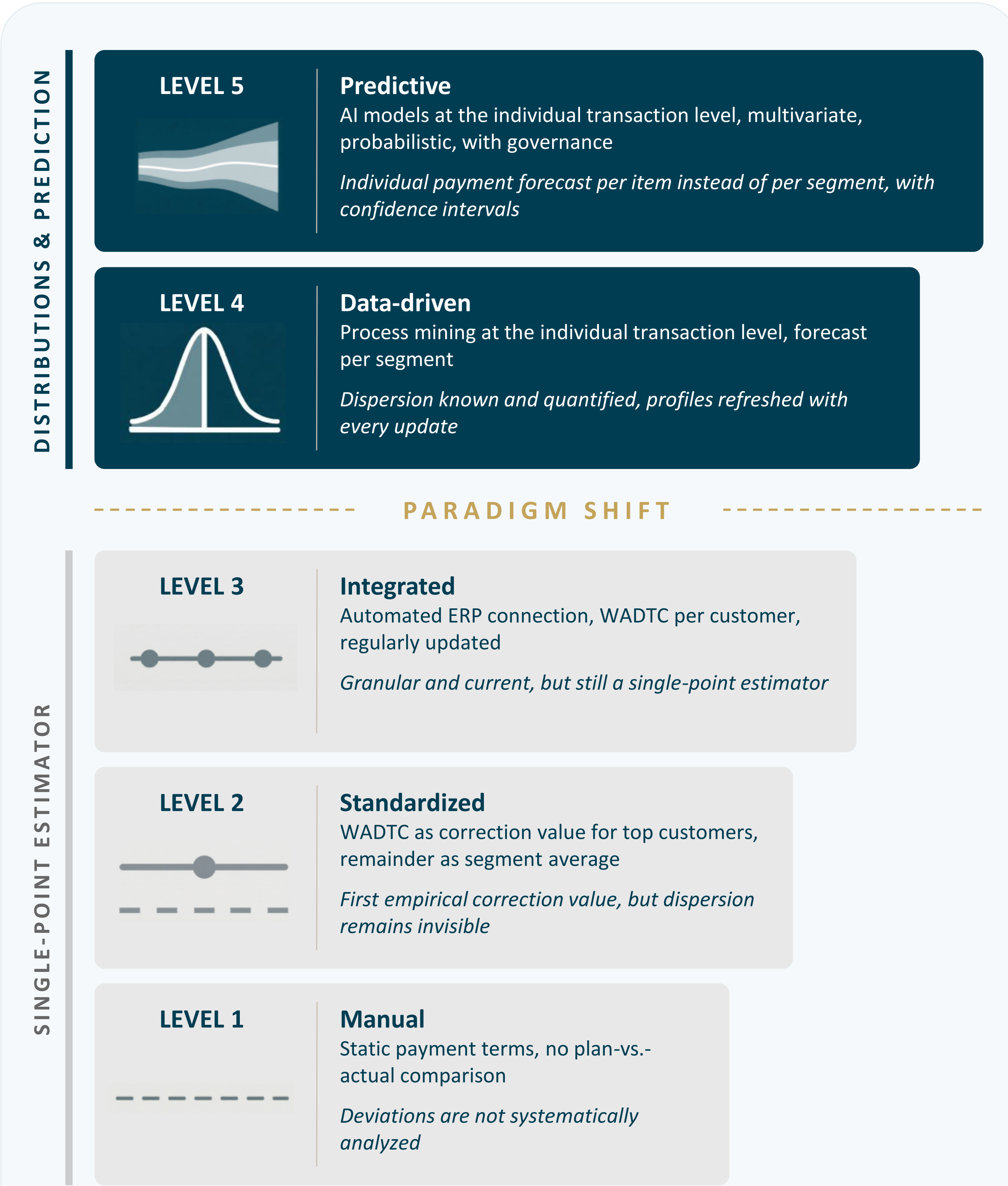


Fig. 3: Maturity model of forecasting methodology (own illustration)

04

Looking back: From transaction record to payment profile

What process mining does differently

Process mining is an analytical method that reconstructs the actual flow of business processes from structured process logs (event logs) in IT systems.[1]

A conventional analysis of payment transactions does yield the WADTC (down to the document level). Process mining goes further and links incoming payments to the process events along the entire document flow, reconstructing the process path from the sales document to the incoming payment. For liquidity forecasting, the section from invoicing onward is operationally the most relevant. Upstream process steps such as order type or partial delivery feed in as context attributes and improve forecast quality. On the disbursement side, the same logic applies from purchase order, goods receipt, and invoice approval onward. Through this linkage, the drivers of payment delays become visible. This chapter describes the descriptive part of the approach, namely the systematic analysis of historical payment patterns. The predictive modeling built on top of this using AI is the subject of Chapter 5.

From event log to payment profile

The path from raw transaction data to usable payment profiles follows a structured analytical process.

1. **Data extraction and event-log construction.** The relevant events are extracted from the ERP system and structured as an event log. Each entry contains at least a timestamp, an activity label (e.g., "Invoice created", "Payment received"), and a unique document identifier (Case ID).[1],[2] The Case ID is the individual invoice, since linking payment events to the invoice represents the most granular analyzable unit.

2

Process Discovery. A typical order-to-cash process often reveals hundreds of process variants identified by process-mining algorithms: standard flows, flows with partial deliveries, credit notes, dunning notices, and complaints. Each variant has its own lead-time profile.

3

Payment-behavior analysis. For each customer (or customer segment), the distribution of actual payment durations is calculated. The result is a complete distribution curve that also shows the median, the dispersion, possible clusters, and outliers.

4

Pattern identification. Process mining uncovers systematic patterns that remain invisible in a conventional analysis. Typical findings: seasonal fluctuations (e.g., accelerated payment behavior in Q4 due to budget outflows, slowdown in Q1), size-dependent effects (e.g., cash discount usage only on invoices below certain thresholds), weekday effects in payment runs, and correlations with internal process bottlenecks (e.g., delayed invoice approval around month-end).

5

Segmentation. Customers are grouped into clusters based on their payment-behavior profile. Typical segments: "on-time payers" (low deviation from the payment term), "systematic delayers" (consistently later payment, but predictable), "seasonal patterns" (strong dependence on quarter or month), "volatile payers" (high dispersion). This behavior-based segmentation forms the basis for differentiated forecast assumptions and at the same time provides a solution for new customers without their own payment history.

The result of Steps 1 to 5 is a well-founded payment profile per customer based on a probability distribution of the expected payment duration that accounts for seasonal, size-dependent, and process-related influencing factors. In environments with frequent partial payments or offsets, the forecast focuses on the dominant payment timing per document. These profiles are not an analytical end in themselves. In rolling liquidity planning, they replace rigid due dates, average values, and flat safety buffers with more robust assumptions and corridors.

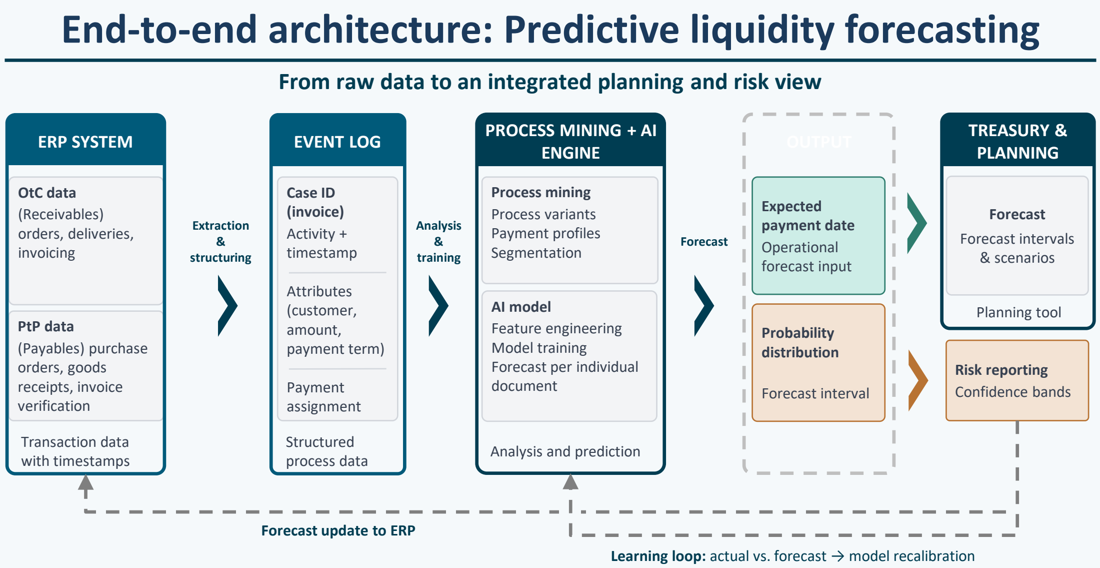


Fig. 4: End-to-end architecture: Predictive liquidity forecasting. From ERP document through process mining and AI forecasting to the planning interface with a governance feedback loop. (Source: own illustration)

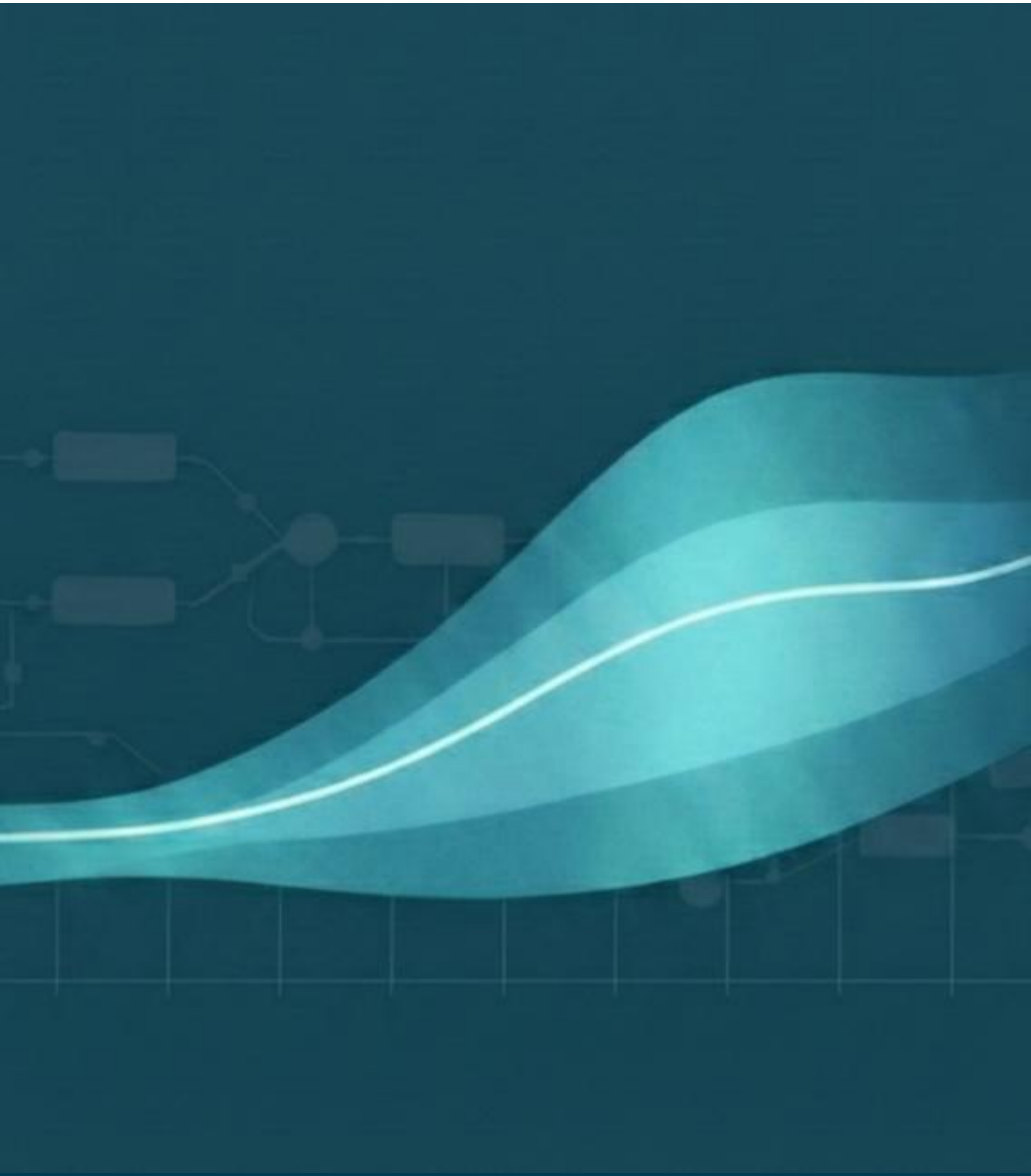
Integration into rolling planning

The integration of the process-mining results into the existing planning process is targeted specifically at the points where static assumptions have been used so far. Figure 4 shows the end-to-end logic from ERP data extraction through process mining and AI to planning integration. The individual components are explained in detail in Chapters 5 and 6.

For open items, the calculation is no longer based on the contractual payment term or WADTC, but on the empirical payment profile of the respective customer or segment. For operational implementation this means: the forecast model calculates, for each open receivable, an expected payment date that is carried into the planning tool as an updated plan date. The aggregation of the individual forecasts into the weekly cash inflow is based on the individual probability distributions and enables the calculation of confidence bands at the aggregate level. On this basis, confidence scenarios can also be defined: for example, a P90 scenario (the cash inflow amount that is reached with at least 90 percent probability) for conservative credit-line management, and a P50 scenario (median) for operational planning. The confidence level thereby becomes an explicit steering variable. Treasury deliberately decides how much forecast uncertainty it accepts and sizes reserves and credit lines accordingly.

On the disbursement side, the insights from the PtP process feed directly into the forecast of payables open items. Instead of the theoretical due date, a realistic disbursement date is determined that already accounts for systematic internal patterns (e.g., delayed invoice approvals). On the disbursement side, the prudence principle should apply. However, the forecast should not be based on perpetuating one's own process weaknesses. Internal delays do improve the short-term liquidity position, but are neither plannable nor sustainable. It is therefore advisable, on the payables side, to use the contractual due date as a lower bound and to capture delays only as a sensitivity.

The planning rhythm and the existing planning tools are retained, but reporting and decision logic are extended: confidence bands enable risk-adjusted management, and the bandwidth of the forecast becomes a steering variable for credit lines and reserves.



05

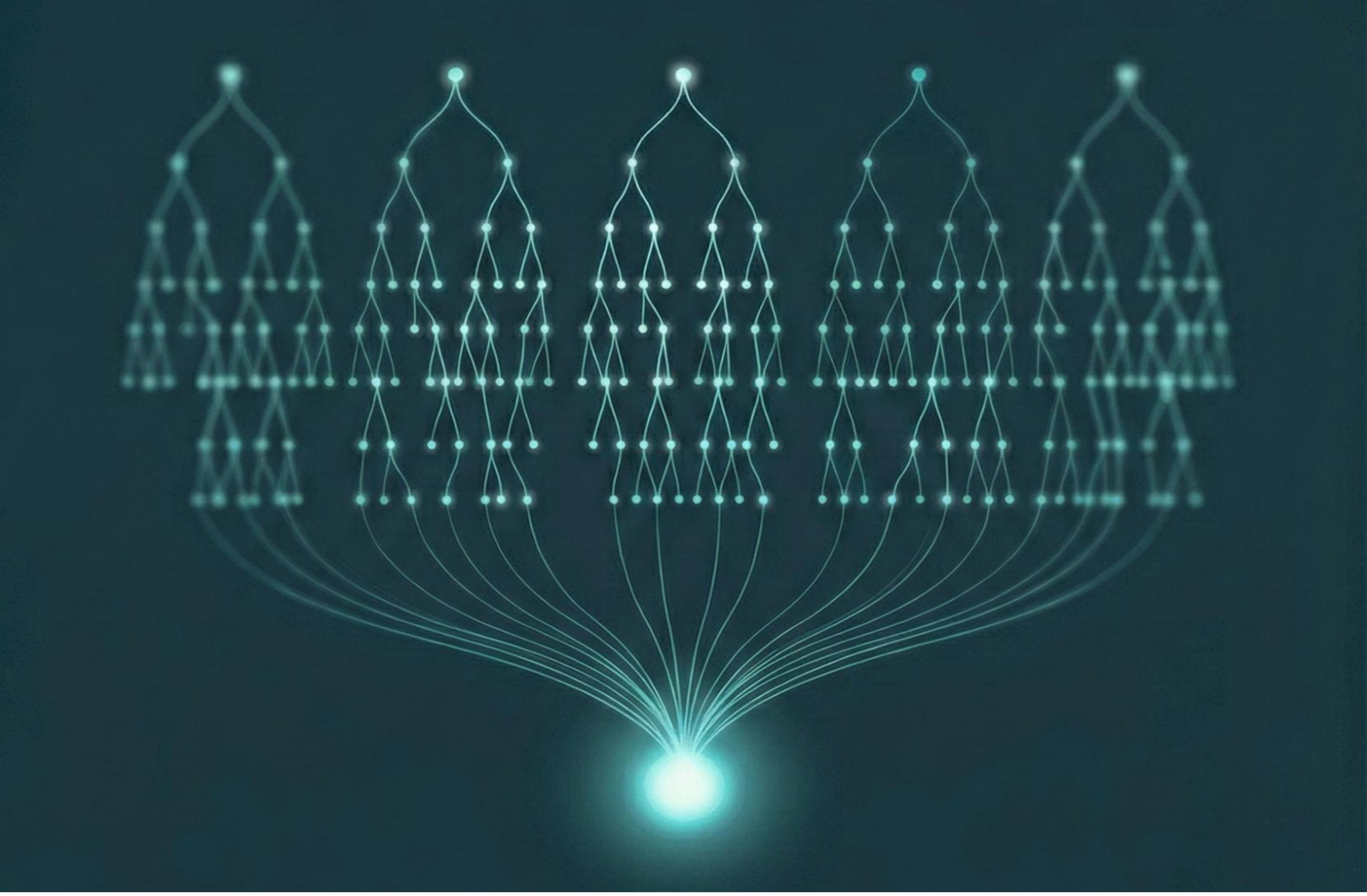
Looking ahead: Patterns over averages

Process mining provides the empirical basis: payment profiles, distributions, and segments (see maturity Level 4). Translating these into individual, context-based payment forecasts requires the next step: AI models that are trained on historical payment patterns and process data, and that generate, for each open receivable, an individual forecast of the expected payment date. On the payables side, the approach can be applied analogously.

From hindsight to prediction

A machine-learning-based (ML) forecast model for incoming payments typically uses input variables (features) from four categories: (1) the customer's historical payment behavior (WADTC, variance, trend over recent periods), (2) invoice attributes (amount, payment term, type, company), (3) context factors (month, quarter, holidays, dunning level, age of the receivable, open payment discrepancies), and (4) process characteristics from process mining (process variant, order type and delivery mode as context attributes, number of process loops). In practice, the algorithms used are primarily Gradient-Boosted Decision Trees (e.g., XGBoost, LightGBM) and Random Forests. These algorithms are well suited to structured, tabular business data and model non-linear relationships as well as interaction effects between variables. In the academic literature and in practical projects, such models at the invoice level consistently show clear improvements over simple average assumptions.[8] The achievable accuracy depends, however, heavily on data quality, customer structure, and model calibration. For treasury management, the mean absolute forecast error in days (at the individual-document level) and the calibration of the forecast intervals (at weekly aggregates) are the most relevant quality measures.[6] Introducing features that capture historical customer behavior has the potential to significantly improve the forecast over pure invoice attributes.

The advantage over a univariate distribution function from process mining: ML models account for the



interactions of several factors simultaneously. This multivariate modeling goes beyond simple average values or segment-based distributions. In particular, the process-mining perspective contributes features that cannot be derived from pure ERP master and transactional data: the process variant of a document (e.g., a partial delivery with subsequent invoicing) explains payment delays that would otherwise remain as unexplained variance in the model, without reconstruction of the process path.

From point estimator to distribution. Decision trees initially provide a point estimator (the expected payment date). For the probabilistic extension there are several established approaches: quantile regression trains the model directly on specific percentiles (e.g., "By when does the customer pay with 90% probability at the latest?") instead of estimating only the expected value. Conformal prediction uses the model's historical forecast errors to generate retrospectively calibrated forecast intervals without changing the model itself.[3] The empirical residual distribution from backtesting evaluates how strongly forecasts deviated from actuals in the past and derives bandwidths from this. In practice, the combination of point estimation and conformal prediction has proven a robust entry point:[4] the method generates calibrated forecast intervals on the basis of historical residuals, requires no model change, and is directly verifiable (e.g., "If the model outputs an 80% forecast interval, then in hindsight roughly 80% of the realized incoming payments should have fallen within this interval."). The formal calibration holds under sufficiently stable data distributions. In the case of significant shifts in payment behavior, the calibration must be continuously monitored and, where applicable, readjusted. For the steering decision this means: treasury no longer receives a single number, but a calibrated corridor with a clear statement of how much reserve buffer the current forecast uncertainty requires.

Ranges instead of point values

Confidence bands can already be derived from the process-mining distributions (maturity Level 4). The added value of AI modeling lies in the precision of these bands: because the model accounts for several influencing factors simultaneously, the intervals become narrower and thus more useful operationally. The operationally most valuable output of an AI-supported forecasting approach is the probabilistic forecast. Instead of a single plan figure ("In CW 24 we expect incoming payments from receivables of EUR 14.3m"), the model delivers a forecast interval ("With 90 percent probability, incoming payments from receivables in CW 24 lie between EUR 13.1m and EUR 15.8m"). This interval refers to the working-capital-related share of cash flow (incoming payments from receivables, outgoing payments to suppliers).

Figure 5 visualizes the difference between a point-value-based and an interval-based liquidity forecast.

Treasury steers along the percentiles of the aggregated weekly incoming payments. P90 for conservative credit-line sizing, P50 for operational planning.

In day-to-day operations, this changes the steering logic. The treasury team steers actions along the bandwidth. With a narrow band (high certainty), the liquidity reserve can be reduced or free capital invested short-term. With a wide band (high uncertainty), the credit line is increased as a precaution, or an upcoming disbursement is rescheduled.

While the model measures the forecast error at the individual-document level in days, the operational steering metric is the percentage forecast error on weekly aggregates, because treasury plans at this level. The quality of the probabilistic forecast can be checked directly: if the model outputs an 80% confidence band, then in backtesting roughly four out of five incoming payments should actually fall within this band.[3] This calibration rate is the central quality metric of the forecast.

Point-value-based vs. interval-based liquidity forecast

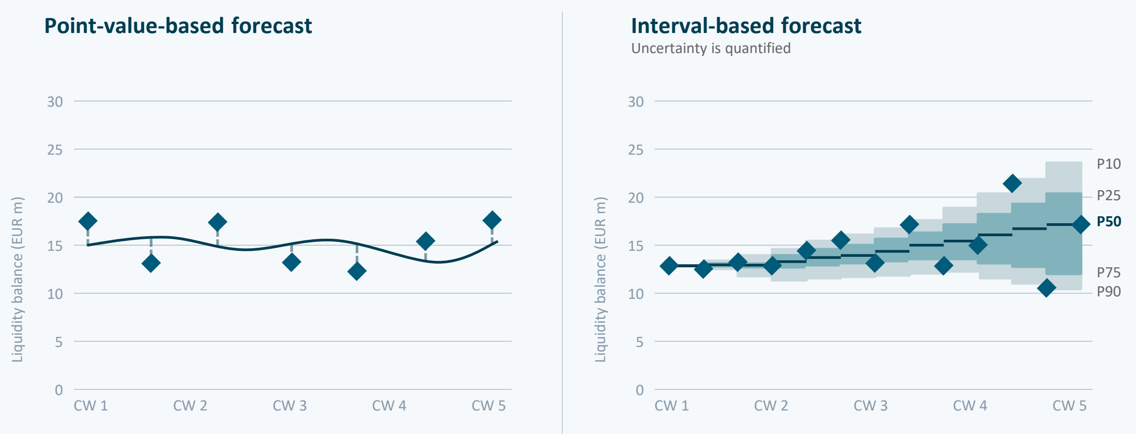


Fig. 5: Point-value-based versus interval-based liquidity forecast. Diamonds show actual liquidity. The forecast interval makes planning uncertainty quantifiable and enables risk-adjusted decisions. P90 denotes the value reached with 90 percent probability. (Own illustration, illustrative values)

The ML model converts part of this dispersion into explained variation by tracing it back to concrete drivers (e.g., invoice amount, seasonality, process variant). The remaining, unexplainable residual variance is smaller. This makes the forecast intervals narrower without losing reliability.[6] The operational core of the approach is therefore not to ignore the uncertainty, but to systematically reduce the unexplained share.

The weekly corridors are not created by mere addition of expected payment dates, but by simulation-based aggregation of the individual distributions. Using Monte Carlo simulation, common influencing factors such as seasonality, industry conditions, and payment-run patterns are accounted for as correlated disturbance variables. The result is a corridor that realistically reflects portfolio effects, instead of assuming independence of the individual positions. In this way, confidence scenarios for operational planning and credit-line management can be derived robustly.

How the forecast improves over time

Explainability as a precondition for acceptance. A forecast model that no one understands is of no use to anyone. When the model forecasts a significantly later payment for a customer, the planning owner must be able to justify this assessment to the CFO, sales, or credit management. So-called feature-importance analyses and SHAP values (Shapley Additive Explanations)[5] make this possible. They decompose each individual forecast into its drivers and make it traceable (e.g., "Main driver: invoice amount above EUR 500,000 + customer paid 15 days slower over the last 3 months + Q1 effect"). This allows the planner to explain any deviation from the previous plan figure with concrete, data-driven arguments. This transparency is decisive for the model to survive in day-to-day operations. Equally important is explainability in the plan-vs.-actual comparison: if a forecast payment fails to arrive or comes in significantly later, the feature contributions reveal which assumption did not materialize (e.g., "Based on the historical Q4 pattern, the model expected an accelerated payment, but the customer changed its payment run"). In this way, the forecast error is not only quantified but traced back to its most important forecast drivers. This is what distinguishes a learning system from a "black box".

The learning loop. The key idea is continuous feedback: every actual incoming payment is reconciled with the corresponding forecast and feeds into the next model update. In a conventional system, forecast errors are at best analyzed retrospectively and lead to manual adjustments. In a learning system, this adjustment takes place in a structured, data-based way. Payment behavior changes over time, whether through cyclical shifts, changed payment terms, or one's own measures such as adjusted dunning.[7] These shifts can be systematically detected in ongoing forecast monitoring and accounted for in the periodic model update.

The most direct way to capture policy effects is to encode them explicitly as a feature: a change in dunning strategy, an adjustment of payment terms, or a change in approval processes is stored as a structural variable in the model. This way, the model does not first have to learn the change from the data, but can account for it from the moment of introduction. In addition, the model captures gradual changes through two data-driven mechanisms: first, through recency weighting, which gives more recent data points a higher weight and thus automatically captures creeping shifts. Second, through periodic retraining on updated data, which captures structural breaks.

Governance and operating model. In a financial environment, a forecast model must not "learn automatically" without control. The model update takes place periodically (e.g., monthly) and is subject to a defined approval process. Key components are monitoring of forecast accuracy (forecast vs. actual, e.g., as the mean absolute percentage forecast error on the weekly incoming payments), defined thresholds for recalibration, auditability of the model results, and a clear escalation process in the event of significant deviations. Likewise, before going live it must be clarified who operates the model on an ongoing basis and how structural breaks (e.g., M&A, system migrations, major changes in the customer portfolio) are handled.

06

Implementation

What the data must deliver

The success of a process-mining- and AI-supported liquidity forecast depends decisively on the quality of the available data. The following prerequisites should be checked before the project starts:

Historical data depth. For robust seasonal patterns, at least 24 to 36 months of historical transaction data are needed.

Master-data quality. Customer master and vendor master must be kept current and consistent. In particular, the assignment of payment terms, customer segments, and regional attributes must be reliable.

Payment assignment. Incoming payments must be unambiguously assigned to the corresponding invoices. In practice, this is not always the case: lump-sum payments without a reference, offsets, "on account" postings, and incorrect assignments in bank clearing lead to the link between payment and invoice being incomplete. If a significant share of payments is cleared manually and the assignment quality is low, the model learns on faulty data.

Therefore, before the actual analysis, a data-cleansing phase is not uncommon. This preliminary stage should not be underestimated, but neither should it be made an obstacle. Even an analysis of the highest-revenue customers with well-maintained data can deliver meaningful results.

Technology stack at a glance

The approach described is not tied to a specific tool. The architecture consists of three functional layers (see Figure 4):

Data and process-mining layer. Event logs are extracted from the ERP system and the process analysis is carried out. Established process-mining platforms such as Celonis, SAP Signavio, UiPath Process Mining, or Noreja offer pre-built connectors for SAP and other ERP systems. In S/4HANA environments, extraction via CDS views and ACDOCA-based data models is recommended. Alternatively, event-log extraction can be implemented with custom ETL processes (Extract, Transform, Load).

Analysis and ML layer. The payment-behavior analysis and the training of the forecast models are carried out via common data-science stacks (Python with scikit-learn, XGBoost, LightGBM) or integrated analytics platforms such as Databricks. What matters is less the choice of framework than the quality of feature engineering and the ability to integrate into the existing IT landscape.

Planning and reporting layer. The forecast results feed into the existing planning environment. Integration typically takes place via an automated data interface. The forecast model writes the updated expected payment date and the confidence information into an interface table, which the planning tool reads automatically.

The integration of the three layers can be carried out incrementally. A pragmatic first step is to run the process-mining analysis as a standalone preliminary study, whose results are initially carried over manually into the planning parameters.

Four phases to production

An effective implementation can take place in four phases that can be carried out in parallel with the ongoing planning cycle. The implementation path described assumes that the company operates at least at maturity Level 2 (structured planning process with access to historical ERP transaction data) and has a sufficient historical data base. The actual duration depends decisively on data quality, the ERP system used, IT security requirements, and the complexity of the integration.

Phase 1: Data readiness and initial analysis. Data quality and availability are checked, event logs for the OtC process are extracted, and an initial process-mining analysis is carried out on the highest-revenue customers. Result: first payment profiles, identified data-quality issues, and, for the first time, empirical visibility into actual payment behavior.

Phase 2: Model development and validation. ML forecast models are trained on historical data and tested against actual values (e.g., forecast the incoming payments of the last six months, then reconcile with the actual values). Result: a validated forecast model whose improvement over the status quo can be quantified.

Phase 3: Integration and parallel run. Old and new methodologies run in parallel over several planning cycles, and the planning owners are trained. Result: proof that the improvement holds under real conditions.

Phase 4: Production operation and rollout. Monitoring and governance structures are established, and the methodology is gradually extended to further customer segments, the payables side, and additional planning horizons. By this point at the latest, operating responsibility must be clarified. Who operates the model after the project ends, and how are licenses and infrastructure funded on an ongoing basis?

Quick wins can already be realized in Phase 1. The existing processes and systems initially remain untouched and are enriched step by step. This reduces implementation risk and the change pressure on the organization.

07

Business Case

The economic benefit does not arise in the analysis itself. It arises where more robust forecast corridors lead to better decisions about liquidity reserves, line utilization, and responsiveness. This benefit can be described on two levels. First, through a more robust forecast that increases steering quality. Second, through operational insights that act directly on working capital.

Improved forecast quality. The effects of a more robust liquidity forecast cannot always be expressed directly in euros. The advantages are nonetheless evident: treasury steers on the basis of more reliable figures, the reconciliation effort between business functions and financial planning declines, and the reaction time to deviations is shortened. Teams that today spend several days per month on the manual preparation and correction of forecast data can redirect this time to value-adding analysis and steering. Moreover, a more reliable forecast enables a forward-looking management of credit lines, lowers the necessary liquidity buffer, and creates room for FX hedging.

Quantifiable effects through working-capital optimization. Besides forecast patterns, the process-mining analysis also uncovers operational improvement potential. On the receivables side, delayed invoicing, slow clarification processes, and conspicuous payment behavior of individual customers become visible - customers who systematically delay incoming payments and where targeted receivables management can be applied.

On the payables side, it becomes visible where cash-discount windows lapse unused. Whether using them is economically sensible depends on the individual case, since earlier payment ties up short-term liquidity. These insights are not the subject of the forecasting approach described here, but they can trigger standalone optimization initiatives in working-capital management.

Even at mid-sized companies, these effects add up noticeably. Tied-up working capital is freed, the financing requirement falls, and downstream cost blocks such as line fees, overdraft interest, and process costs in receivables management are reduced. Depending on revenue volume and customer structure, the annual savings move into the six-figure range. The investment for an initial implementation (data extraction, model development, pilot integration) can often pay back within one to two years, provided the data quality is robust and operating responsibility is clearly defined.

08

Getting started

The accuracy of short-term liquidity planning stands or falls, in many cases, with the assumptions about actual payment behavior. Process mining makes this behavior visible. Machine-learning/AI methods translate these insights into predictive models that improve over defined governance processes on a periodic basis.

The approach does not replace the existing planning infrastructure, but improves the validity of the forecast assumptions and extends the decision logic. It can be implemented incrementally, is compatible with existing planning tools and system landscapes, and delivers measurable and sustainable results as early as the initial project phases. The data for this is available at most companies, and the technology is at hand. What is missing is the systematic evaluation of the transaction data that the ERP system has long been capturing. Companies that take this step change not only their forecast quality. They replace the estimate of a single exact figure with a quantified range and steer liquidity along defined confidence levels rather than on the basis of a single point estimate.

The decisive first step therefore begins not with new software, but with two questions:

"How far does actual payment behavior deviate from the agreed payment terms, and how large is the dispersion?"

Anyone who has quantified the dispersion once will no longer accept average values as a steering basis.

Sources

[1] van der Aalst, W. M. P. (2016). Process Mining: Data Science in Action (2nd ed.). Springer. ISBN 978-3-662-49850-7. DOI: 10.1007/978-3-662-49851-4

[2] van der Aalst, W. M. P. et al. (2012). Process Mining Manifesto. In BPM 2011, LNBIP, Vol. 99 (pp. 169–194). Springer. DOI: 10.1007/978-3-642-28108-2_19

[3] Vovk, V., Gammerman, A. & Shafer, G. (2005). Algorithmic Learning in a Random World. Springer. ISBN 978-0-387-00152-4. DOI: 10.1007/b106715

[4] Romano, Y., Patterson, E. & Candès, E. J. (2019). Conformalized Quantile Regression. NeurIPS 2019, 32, 3538–3548. arXiv: 1905.03222

[5] Lundberg, S. M. & Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. NeurIPS 2017, 30, 4766–4777. arXiv: 1705.07874

[6] Gneiting, T. & Raftery, A. E. (2007). Strictly Proper Scoring Rules, Prediction, and Estimation. JASA, 102(477), 359–378. DOI: 10.1198/016214506000001437

[7] Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M. & Bouchachia, A. (2014). A Survey on Concept Drift Adaptation. ACM Comp. Surveys, 46(4), Art. 44, 1–37. DOI: 10.1145/2523813

[8] Appel, A. P. et al. (2019). Optimize Cash Collection: Use Machine Learning to Predicting Invoice Payment. arXiv: 1912.10828

[9] Gleich, R., Kappes, M., Maron, C., Oehler, K. & Tschandl, M. (eds.) (2024). Planung und Forecasting neu gedacht. Haufe. ISBN 978-3-648-18082-2

[10] REL Consulting / The Hackett Group (n.d.). Working Capital Metrics (WAT, WADTC, WADTP). thehackettgroup.com/working-capital-management

About the authors

Axel Meythaler

Managing Director

Axel.Meythaler@Fortlane.com

+49-151-17141018

Pirmin Mutter

Partner

Pirmin.Mutter@Fortlane.com

+49-151-18236029

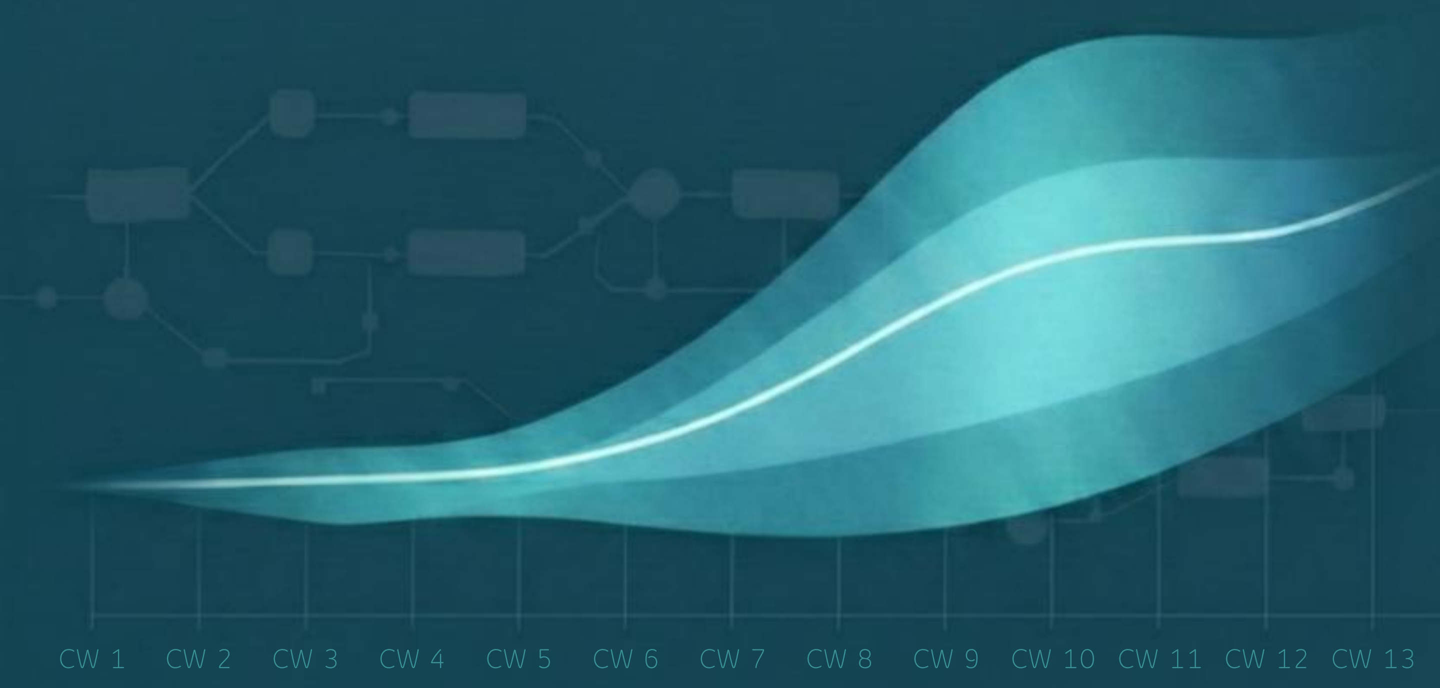
Pascal Krawczyk

Senior Consultant

Pascal.Krawczyk@Fortlane.com

+49-151-18236025

FORTLANE PARTNERS
STRATEGY · M&A · TRANSFORMATION



About Fortlane Partners

Fortlane Partners is a leading European consultancy with a focus on strategy, M&A, and transformation. With an integrated advisory approach, Fortlane Partners combines management-consulting and corporate-finance expertise to support companies in successfully shaping their future.

Our work combines deep industry knowledge with analytical rigor and pragmatic implementation capability. We work collaboratively and results-oriented - from the first concept to measurable impact.

In the Transformation and Performance area, Fortlane Partners supports companies, among other things, in optimizing working-capital processes, implementing data-driven planning approaches, and applying process mining and AI in operational steering.