WEVO GL ANALYTICS | WHITE PAPER

Wevo Creates the Data Your AI Needs

A clean, time-stamped history of the ledger figures that matter - exact, structured and ready for any analytics or AI tool

Professional Software Ltd

1. Executive summary

A General Ledger is built to answer one question well: where did each balance end the period? Management reporting has been organised around that question for decades. But the questions finance leaders increasingly ask are different. How did we get here? Is this line behaving the way it normally does? Where will it land if nothing changes? Those are questions about the path a figure takes, not the total it reaches, and they are exactly the questions analytics and artificial intelligence are good at answering - provided the right data exists.

In most organisations it does not. Ledgers overwrite balances as postings arrive, reporting tools refresh overnight, and nobody keeps the intermediate positions. An AI initiative in finance usually starts by trying to rebuild a history that was never recorded.

Wevo GL Analytics records it. This paper explains how, and what becomes possible:

- **The building blocks are already in Wevo.** Your organisational and account structures, the pre-calculated figure at every point where they meet, and a watchlist that captures a time-stamped history of the figures you choose.
- **Capture is automatic.** An administrator marks any organisation and account node pair. From then on, every engine run that updates the pair - in practice every 15 or 30 minutes - writes a snapshot of its opening position and all 16 periods, for movements and balances, in every defined currency.
- **Wevo creates the data; your analysis runs elsewhere.** Snapshots land in an ordinary table and are archived, typically monthly, to a separate analytics database on its own server - ready for SQL, your BI platform, Python or R, or your own AI platform. Wevo reuses the storage, AI workloads never compete with reporting, and nothing leaves your estate.
- **The uses are practical.** Rolling forecasts, early warning of budget overruns, detection of unusual patterns such as balances that build before period end and reverse after, intercompany drift, covenant headroom and more - thirteen are described in section 6.
- **Wevo is AI-ready, not AI-powered.** It supplies structured, exact, time-stamped data. You choose the models and the questions, and finance stays in control of the conclusions.

> *The ledger tells you where a balance ended. The watchlist tells you how it got there - and that is the part analytics and AI can learn from.*

2. Why period totals are not enough

2.1 A ledger answers where, not how

A period-end balance is a single number. Two cost centres can close the month on identical figures and have behaved completely differently getting there: one spending steadily, the other flat for three weeks and then posting a large accrual on the last afternoon. A month-end report cannot tell them apart. Neither can a comparison of this month's total with last month's.

Copyright 2026 Professional Software Ltd. Distribute freely.

The information that separates them - the timing and shape of the postings - exists only while the period is open. Once the ledger rolls forward, it is gone from every report finance normally sees.

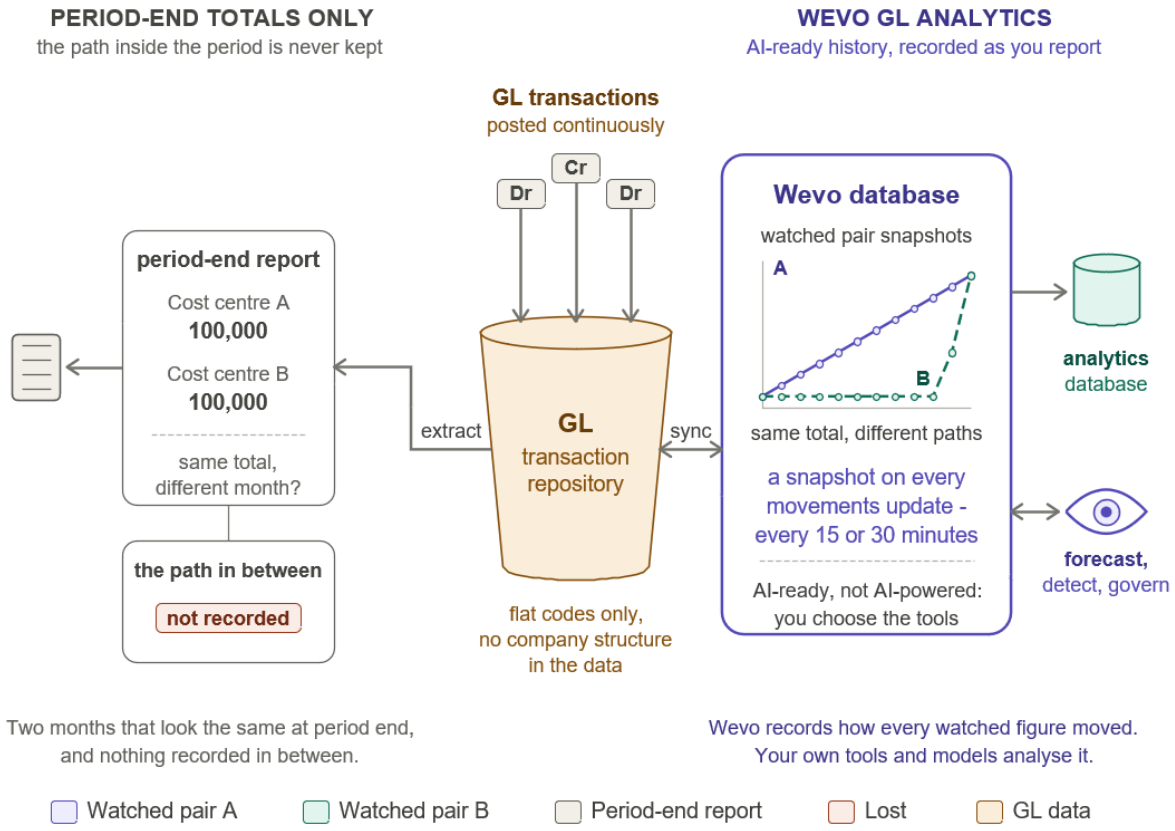


Figure 1 - Two cost centres close the period on the same total; only the snapshot history shows the different paths they took

2.2 What finance teams miss as a result

- **Build-up and reversal.** Balances that rise sharply just before period end and fall away just after it.
- **Run-rate drift.** A cost line running ahead of its usual pace in the first half of the month, visible only when it has already overrun.
- **Timing changes.** Accruals, recharges or intercompany postings that start arriving later, earlier or not at all.
- **Late adjustments.** Changes to a closed period's figures that arrive weeks after the close.
- **Silence.** A recurring posting that simply stops. A zero is information, but only if something recorded when it should not have been zero.

2.3 Why AI projects in finance stall

Analytics and machine learning need history: a consistent, regularly spaced record of how each figure has behaved, labelled in the way the business understands it. In practice, most finance teams find that assembling that history takes far more effort than the analysis itself. Transactions have to be extracted, mapped to the reporting structure, reconciled, re-aggregated at the level managers act

Copyright 2026 Professional Software Ltd. Distribute freely.

on, and stored somewhere safe - and even then the intermediate positions inside each period are missing, because no system kept them.

The result is familiar: promising pilots built on a one-off extract, which cannot be refreshed, cannot be trusted by the controllers who own the numbers, and quietly stop.

3. What AI can realistically do for corporate reporting

It is worth being precise about what analytics and AI bring to financial reporting, because the value is real but specific. The techniques fall into five broad families:

Capability	What it means for finance	What it needs from the data
Forecasting	Projecting where a balance or run-rate will land, with a realistic range	A long, regularly spaced history of the same figure
Anomaly detection	Flagging figures that depart from their own normal behaviour	Enough history to establish what normal looks like
Pattern and change detection	Spotting seasonality, and the point at which behaviour changes	Unbroken series, with the timing of each change preserved
Comparison and clustering	Grouping cost centres or divisions that behave alike, and finding the odd one out	Comparable series across many figures, on the same structure
Narrative generation	Drafting variance commentary and answering questions in plain English	Structured, labelled, trustworthy figures and their movements

Just as important is what these techniques cannot do. They do not replace judgement: a model can flag that a suspense account is behaving unusually, but only a person can decide whether it matters. They do not prove wrongdoing: pattern detection finds things worth investigating, not conclusions. And they cannot fix poor data - which is why the quality and structure of the underlying history matter more than the choice of model.

> *The deciding factor in finance AI is rarely the model. It is whether a clean, structured, trustworthy history exists for it to learn from.*

4. The building blocks already in Wevo

4.1 Structure: figures at the level managers act on

Wevo holds your General Ledger at leaf level and rolls it up through two independent structures: an organisational hierarchy (group, division, department, cost centre) and an account hierarchy (the chart of accounts and its groupings). Every point where the two meet is a **node pair** - for example the Legal and Governance Division against Advisory Service Fees - and its figure is calculated in advance and kept current to within minutes of the ledger.

This matters for AI. Raw ledger transactions are noisy and numerous; the figures managers are accountable for are aggregates on the reporting structure. A node pair is already at that level,

Copyright 2026 Professional Software Ltd. Distribute freely.

already labelled in the business's own language, and already reconciled to the ledger. Analysing node pairs means analysing the numbers people actually discuss.

4.2 Choosing what to watch

Any node pair can be placed on a **watchlist** for a cube. An administrator does this during ordinary Drilldown: with the chosen organisation node and account node in focus, one click on the watchlist button adds the pair. From that point, two pairs of gold glasses - one on each hierarchy panel - show that the pair in focus is being watched, so everyone drilling through the figures can see at a glance which intersections are accumulating history. Standard users see the indicator; only administrators can add or remove pairs.

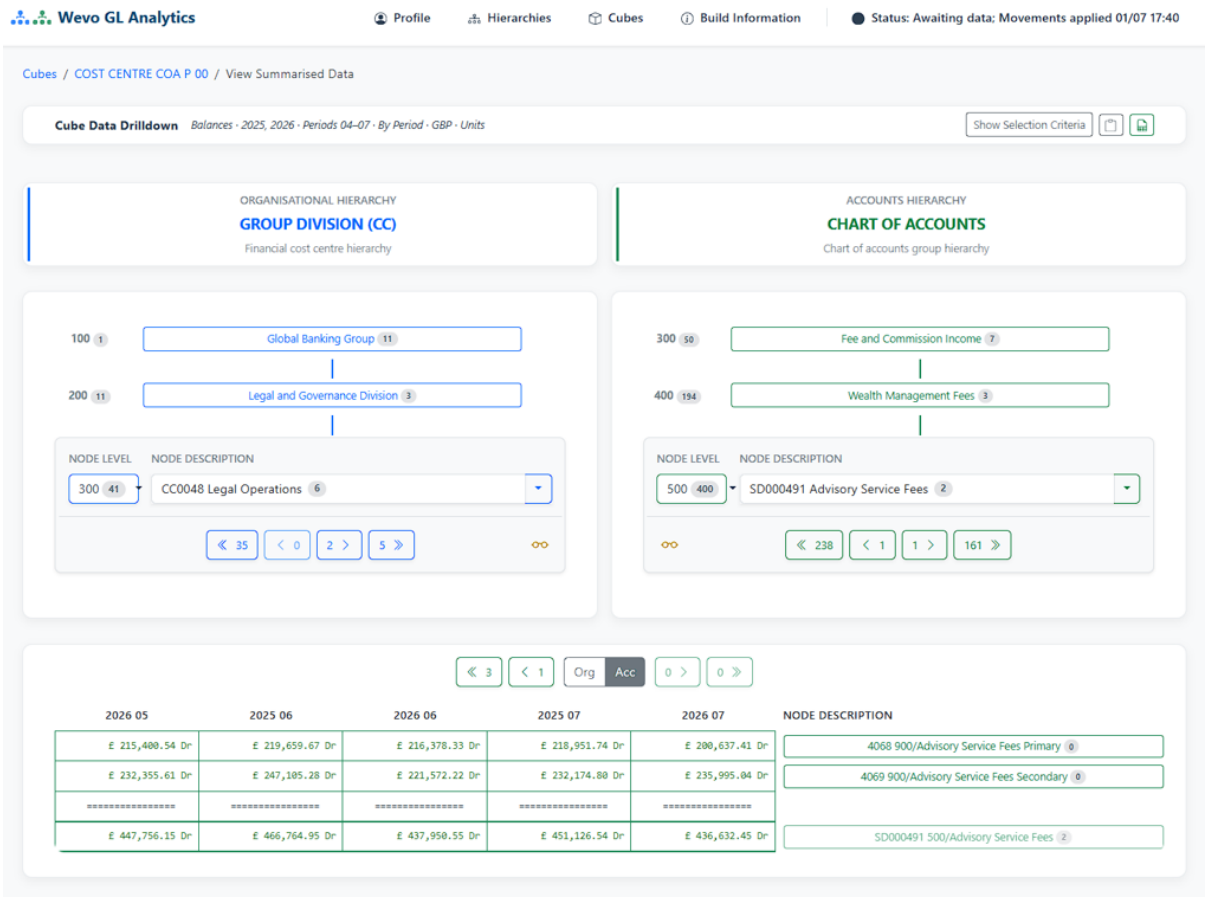


Figure 2 - The Drilldown transaction. The gold glasses on both hierarchy panels show that the node pair in focus is on the watchlist

The watchlist is self-maintaining. Entries are stored by node name rather than internal index, so they survive hierarchy rebuilds; a pair whose organisation or account node no longer exists in the cube after a restructure is retired automatically, and its snapshots, which no longer describe anything in the reporting structure, are cleared down.

4.3 The snapshot

On every incremental movements update in which a watched pair is calculated, Wevo writes a **snapshot**: one row per watched pair, per cube year, per currency, per engine run. With movements files typically arriving every 15 or 30 minutes through the working day, each watched figure accumulates dozens of time-stamped observations a day.

Copyright 2026 Professional Software Ltd. Distribute freely.

Each row carries the complete picture for the year as it stood at that moment: the opening movement and opening balance, then the movement and balance for each of the 16 periods - 34 values in all. Amounts are held as exact whole numbers in minor units (pence, cents), so there is no floating-point rounding anywhere in the history. Periods not yet reached hold zero; zeros in earlier periods are deliberately kept, because a zero where a posting was expected is itself a signal.

Column	Holds
SNAP_Snapshot_idx	Unique identifier of the snapshot row
SNAP_Watchlist_idx	The watchlist entry, and through it the organisation and account node names
SNAP_Cube_Year_idx	The cube year the figures belong to
SNAP_Currency_code	The currency of the values in the row
SNAP_Run_timestamp	When the engine run wrote the snapshot, to the millisecond
SNAP_Mstart_amt, SNAP_Bstart_amt	Opening movement and opening balance
SNAP_MPD01_amt ... SNAP_MPD16_amt	Movement for each of the 16 periods
SNAP_BPD01_amt ... SNAP_BPD16_amt	Balance for each of the 16 periods

The companion watchlist table, Wevo_Monitoring_watchlist, records which pairs are monitored for each cube, by organisation and account node name, who added them and when, and whether an entry has been retired. Snapshots deliberately carry no internal node identifiers: those are reassigned when a hierarchy is rebuilt, whereas the node names on the watchlist entry are stable. The snapshot table uses page compression, so the many zero values cost almost nothing to store.

Copyright 2026 Professional Software Ltd. Distribute freely.

4.4 From watched pair to insight

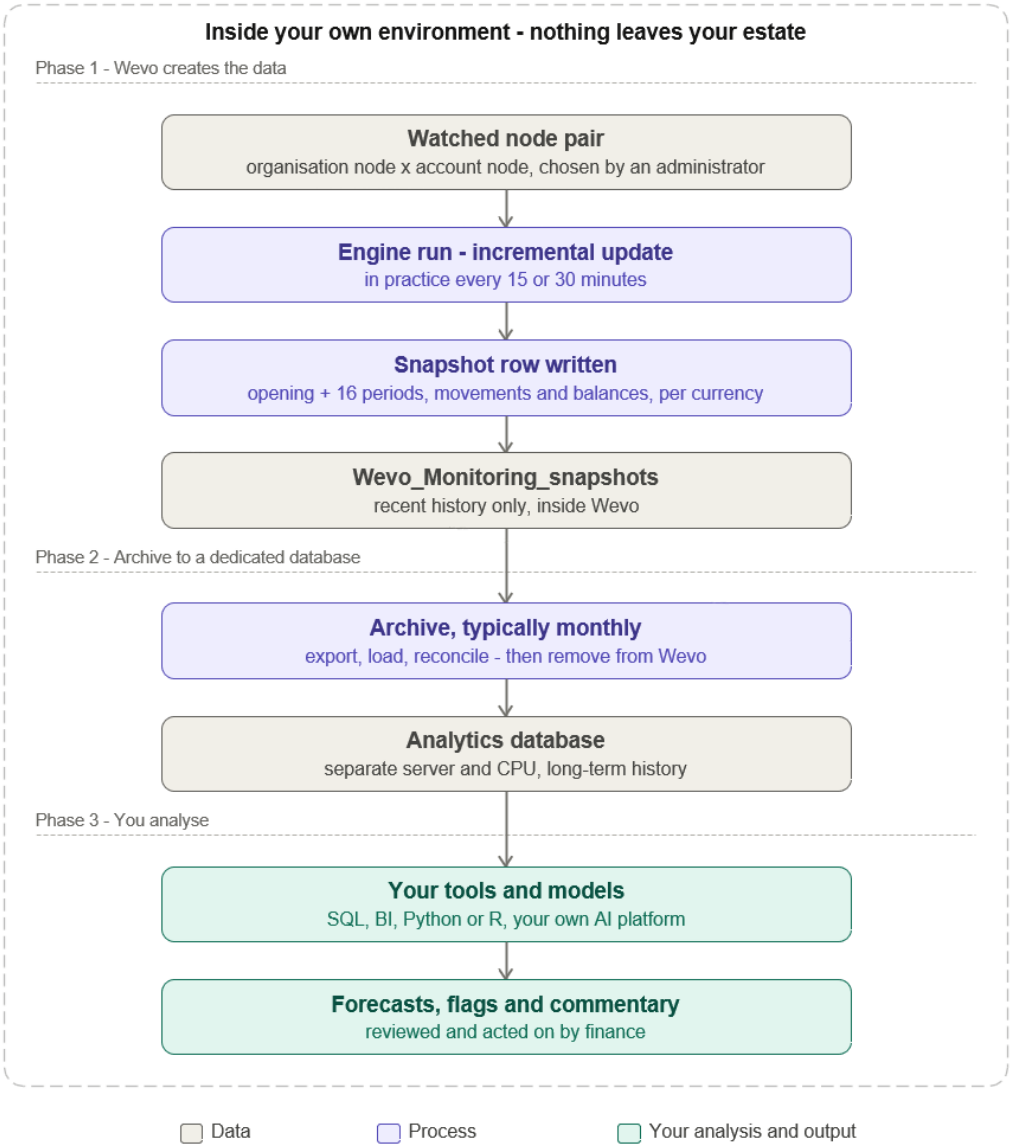


Figure 3 - Wevo creates the history; it is archived to a dedicated analytics database, where your own tools and models analyse it

4.5 Why this shape of data suits analytics

- **Time-stamped and regular.** Observations every 15 or 30 minutes, captured by the engine rather than by anyone remembering to run an extract.
- **Business-shaped.** Every series is a figure someone is accountable for, on the structures the business reports through.
- **Exact.** Integer minor units, reconciled to the ledger by construction.
- **Complete.** All 16 periods and the opening position in every row, so late changes to earlier periods are captured too.
- **Multi-currency and multi-year.** One series per currency defined in the system, and per cube year.

Copyright 2026 Professional Software Ltd. Distribute freely.

- **Durable.** Watchlist entries survive reorganisations, so a series is not broken by a restructure.
- **Open.** A standard relational table, archived to whichever database platform your data team already runs, and readable by any tool.

5. Reading the history: two clocks

The feature that makes the snapshot history unusually powerful is that every value carries two times. The **period** says which part of the financial year the figure belongs to. The **run timestamp** says when that figure was known. Most finance data has only the first.

Holding both opens up three kinds of question that are otherwise very hard to answer:

- **What moved, and when.** The difference between two consecutive snapshots is exactly what reached the ledger for that pair in between - the flow of postings through the period, not just its end point.
- **What did we know at the time?** Any past position can be reconstructed exactly as it stood on a given day and hour. That is valuable for audit and review, and it lets forecasting models be tested honestly, using only the information that was available at the time rather than figures that arrived later.
- **What changed after the close?** Because each row carries every period, a change to a closed period's balance shows up the moment it happens, with the time it arrived.

None of this needs specialist tooling. The query below, written in standard SQL, lists every change to period 7 balances for the watched pairs in one cube year, with the amount that arrived between each run and the next:

```
SELECT w.WTCH_Org_node_name,
       w.WTCH_Acc_node_name,
       s.SNAP_Currency_code,
       s.SNAP_Run_timestamp,
       s.SNAP_BPD07_amt / 100.0 AS balance_period_07,
       (s.SNAP_BPD07_amt - LAG(s.SNAP_BPD07_amt) OVER (
            PARTITION BY s.SNAP_Watchlist_idx, s.SNAP_Currency_code
            ORDER BY s.SNAP_Run_timestamp)) / 100.0 AS change_since_last_run
FROM   Wevo_Monitoring_snapshots s
JOIN   Wevo_Monitoring_watchlist w
       ON w.WTCH_Watchlist_idx = s.SNAP_Watchlist_idx
WHERE  s.SNAP_Cube_Year_idx = :cube_year
ORDER BY w.WTCH_Org_node_name, w.WTCH_Acc_node_name, s.SNAP_Run_timestamp;
```

Dividing by 100 converts minor units to pounds or other two-decimal currencies; currencies with a different number of decimal places use the matching divisor. The same pattern, pointed at any period column, feeds a BI chart, a spreadsheet, a Python notebook or a model training set.

The query is written against the Wevo tables. In the analytics database described in section 8, the node names are already on every archived row, so the join is not needed. In either place, group a series by its organisation and account node names rather than by the watchlist entry number: if a

Copyright 2026 Professional Software Ltd. Distribute freely.

pair is removed from the watchlist and later added back, it receives a new entry, and the names are what join its history into one continuous series.

6. What analytics and AI can do with it

The snapshot history supports a wide range of uses. They are grouped below into three families. Each organisation will choose the handful that matter most to it; the data to support all of them is collected in the same way.

6.1 Forecast

Use	The signal in the snapshots	Typical technique	What finance gains
Rolling forecasts	Period-end balances and intra-period run-rates, updated with every snapshot	Exponential smoothing, ARIMA-style models, gradient-boosted regression	A forecast that refreshes itself through the month, not once a quarter
Cash and liquidity projection	The pace of movements on cash, receivables and payables pairs within each period	Run-rate projection with seasonal adjustment	An earlier, evidence-based view of where cash will land
Seasonality	The same periods across several years, at the same point in each period	Seasonal decomposition	Budgets and targets set against how the line really behaves
FX exposure trends	Balances captured in every defined currency	Trend and volatility analysis by currency	Visibility of exposure building up before it is hedged or reported

Copyright 2026 Professional Software Ltd. Distribute freely.

6.2 Detect

Use	The signal in the snapshots	Typical technique	What finance gains
Unusual postings	Spikes, round sums and late close-period adjustments in the change between snapshots	Statistical baselines, isolation forests, rules on size and timing	A short list of movements worth a second look, instead of a full-ledger trawl
Window-dressing	Balances that build before period end and reverse after it	Pattern matching around period boundaries	Period-end behaviour made visible, pair by pair
Suspense and clearing build-up	Clearing accounts whose balance grows instead of returning to zero	Trend and ageing thresholds	Unreconciled items caught while they are small
Intercompany drift	Matching pairs in two entities that stop mirroring each other	Paired-series comparison	Consolidation differences found before the close, not during it
Risk-based audit sampling	Which pairs behave least like their own history	Anomaly scoring and ranking	Audit effort directed by behaviour rather than by size alone

Detection techniques flag patterns worth investigating. They do not establish error or wrongdoing, and every flag should be reviewed by someone who understands the business behind the figure.

6.3 Govern

Use	The signal in the snapshots	Typical technique	What finance gains
Early budget-overrun warning	Cumulative movement against its usual path at the same point in previous periods	Percentile bands, projection to period end	An overrun raised mid-month, while it can still be managed
Margin erosion	Revenue and cost pairs for the same division, captured at the same moments	Ratio trend and change-point detection	Margin pressure seen as it starts, division by division
Covenant and ratio headroom	The balances that make up each ratio, captured continuously	Ratio projection against limits	Continuous visibility of headroom, not a quarterly surprise
Post-acquisition synergy tracking	Cost lines of acquired units before and after integration	Before-and-after comparison, trend analysis	Savings measured as they land, against the plan

6.4 Commentary and plain-English questions

Large language models are well suited to one of the most time-consuming parts of the reporting cycle: writing variance commentary. Given the structured change between snapshots for a set of watched pairs - what moved, by how much, when, and against what usual path - a language model

Copyright 2026 Professional Software Ltd. Distribute freely.

can draft a first version of the narrative for a finance manager to review and correct. The same data lets people ask questions in plain English, such as which divisions' advisory fees are running ahead of their usual pace this month.

Two principles keep this safe. The model works only from the structured figures it is given, so its statements can be checked against them. And because the data sits in your own estate, the model can be one you run privately, so that ledger data never leaves your control.

7. Three worked examples

The examples below use illustrative data to show the kind of result each technique produces. In practice, each would run across every watched pair automatically, surfacing only those that need attention.

7.1 The build-up before period end

A clearing balance sits flat for most of the month, builds steadily over the last three working days, and falls back within two days of the period end. The month-end report shows a single figure - the peak. The snapshot history shows the build-up, the reversal and the time of every step, and a simple rule on behaviour around the period boundary flags it for review.

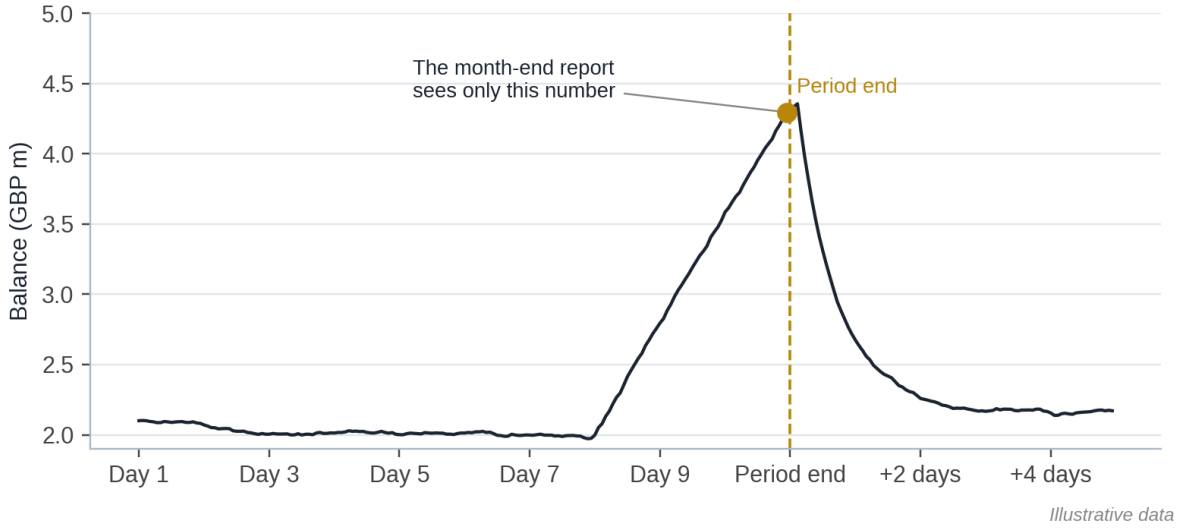
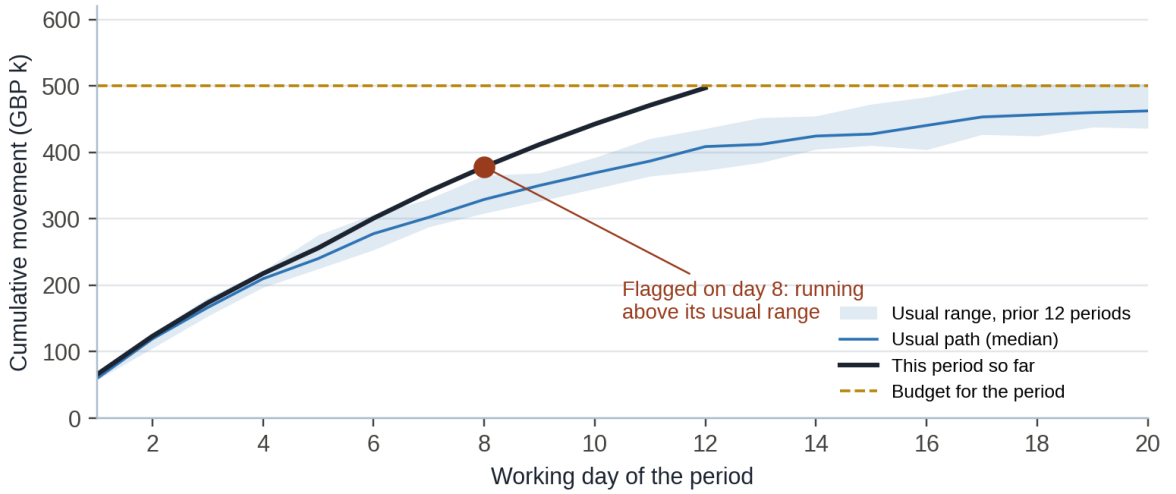


Figure 4 - A balance that builds before period end and reverses after it, as captured by half-hourly snapshots

7.2 A cost line running ahead of its usual path

Each period, a divisional cost line accumulates along a recognisable curve. Using the previous twelve periods, the usual range at each working day can be calculated directly from the snapshots. This period, the line leaves that range on day 8 and is on course to reach its budget for the whole period by day 12. The flag arrives with most of the month still to run.

Copyright 2026 Professional Software Ltd. Distribute freely.

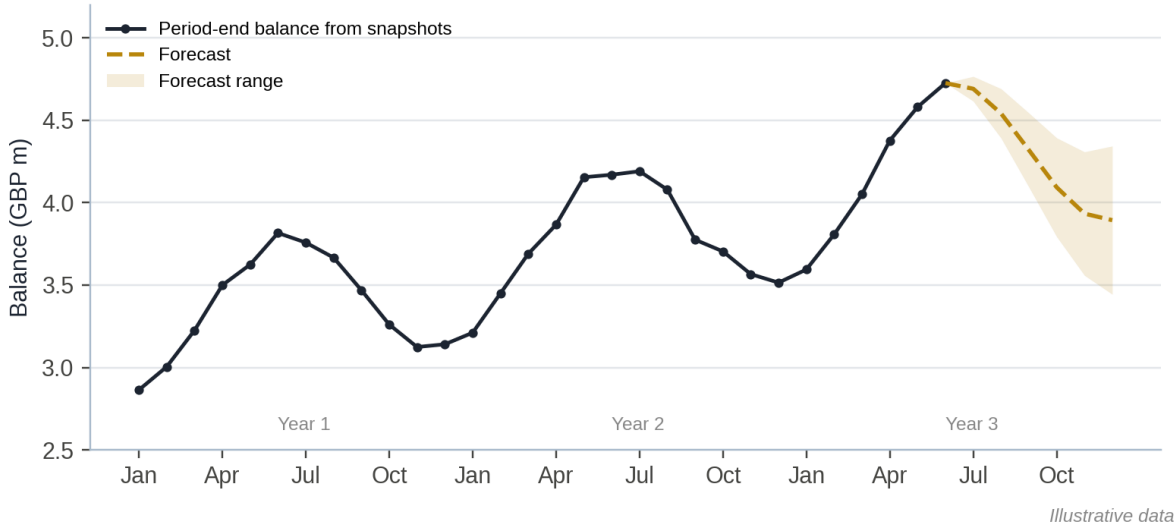


Illustrative data

Figure 5 - Early warning: this period's cumulative movement against its usual range from previous periods

7.3 A rolling forecast from snapshot history

Taking the period-end balance for a watched pair from each period's final snapshot gives a clean monthly series with a clear seasonal pattern. A standard seasonal forecasting model projects the next six periods with a realistic range, and refreshes automatically as each new snapshot arrives - including the partial position of the period in progress.



Illustrative data

Figure 6 - A seasonal forecast built from the period-end balances in the snapshot history

8. A dedicated analytics database

8.1 Wevo creates the data; analysis runs elsewhere

Wevo's job is to keep management reporting current to within minutes of the ledger, and to create the snapshot history as it does so. Analysing that history is a different kind of work. Training a model, scoring every watched pair or querying years of snapshots uses a great deal of processing and storage, and none of it should compete with the engine or with finance users drilling through the figures.

Copyright 2026 Professional Software Ltd. Distribute freely.

The recommended pattern is therefore to keep the two apart. On a regular cycle - typically monthly - completed snapshots are copied out of Wevo into a separate analytics database on its own server, and then removed from Wevo. Wevo remains the creator of the data and keeps only recent history. The analytics database holds the long-term history, on processing and storage sized for analytical work.

8.2 The monthly cycle

1. **Set a cut-off.** Typically the end of the previous month: every snapshot written before that moment is archived.
2. **Export the rows.** Each archived row carries the cube, the year, the organisation and account node names, the currency, the run timestamp and all 34 values, so the archive is self-describing and never depends on Wevo's internal identifiers.
3. **Load them into the analytics database.** Keyed on node names, cube, year, currency and run timestamp, so a load that is repeated after a failure cannot create duplicates.
4. **Reconcile.** Confirm that the row counts and value totals loaded match those exported.
5. **Remove the archived rows from Wevo.** Only once the reconciliation passes does the archive job delete the archived rows from the snapshot table. Wevo does not remove snapshot history itself, other than for pairs whose node no longer exists, so this step belongs to the archive job. If any step fails, nothing is removed and the next run picks the rows up again.

Archived field	Holds
Cube and year	Which cube and financial year the figures belong to
Organisation node name, account node name	The node pair, by the stable names used on the watchlist
Currency	The currency of the values
Run timestamp	When the engine run wrote the snapshot
Opening movement and balance	The opening position for the year
Movement and balance, periods 1 to 16	The complete year as it stood at the run timestamp

8.3 Why this works well

- **Wevo stays lean.** The snapshot table holds only recent history, so its storage is reused month after month and writes stay fast.
- **Analytical work gets its own resources.** Model training and large queries run on separate processing and storage, sized for the job, and can never slow reporting.
- **Any platform will do.** The analytics database can be another SQL Server, PostgreSQL, or the data warehouse or lakehouse your data team already runs.
- **History is kept as long as policy requires.** Retention in the analytics database is independent of Wevo, so years of history can build up without Wevo growing.

Copyright 2026 Professional Software Ltd. Distribute freely.

- **History outlives restructures.** When a node disappears from a cube, Wevo clears down that pair's snapshots as they no longer mean anything in its structures. Anything already archived stays in the analytics database, under the names it had at the time.
- **Closed monthly batches suit model governance.** Each archive run is a fixed, reconciled set of data, which makes model retraining and review cycles easy to evidence.

Monthly is the usual cycle for trend analysis and forecasting. Where a use depends on the month in progress - the in-month early warning in section 7.2, for example - the same cycle can simply run more often, weekly or daily, or the open month can be read directly from Wevo through a read-only database login restricted to the monitoring tables.

9. Putting it to work in your environment

9.1 Architecture

Both databases sit inside your own environment: Wevo's application database in your cloud account, a hybrid arrangement or entirely on premise, and the analytics database wherever your data team runs its platforms. Analytics tools connect to the analytics database in the ordinary way. The archive job itself needs only a read-only login on the two monitoring tables and permission to delete archived rows from the snapshot table. Nothing in this approach calls out to Wevo, to a vendor-hosted service or to the internet.

This is a significant difference from many finance AI offerings, which require ledger data to be sent to an external platform for analysis. With Wevo the data stays where it already is, and you decide which tools and models - if any - are allowed to see it.

9.2 Access and security

Access to the snapshot history - in Wevo and in the analytics database - is governed by database permissions, not by Wevo's reporting profiles, which control what people see in the Drilldown and email reports. Treat read access to the history with the same care as access to the ledger itself: grant it to named analytical services and people, and include both databases in your regular access reviews.

9.3 Retention

Wevo needs to keep only the snapshots not yet archived - typically the current month and part of the previous one. Long-term retention is set by your own policy in the analytics database. The longer the history, the more a model can learn: a few weeks are enough for patterns inside each period, while seasonality and year-on-year comparison need at least two to three years.

9.4 Model governance

- **Start with transparent methods.** Percentile bands, run-rate projections and clear rules are easy to explain to controllers and auditors, and often deliver most of the value.
- **Keep a human in the loop.** Flags and draft commentary are inputs to a review, not conclusions.

Copyright 2026 Professional Software Ltd. Distribute freely.

- **Record what the models do.** Document each model's purpose, inputs, thresholds and owner, in line with the controls you already apply to financial reporting.
- **Test honestly.** Use the run timestamps to evaluate models only on information that was available at the time.

10. Getting started

The watchlist is part of the standard product, at every tier and in every deployment model. Getting value from it is a matter of choosing well and letting the history build.

1. **Choose 10 to 30 node pairs.** Good first candidates are the lines people already worry about: major cost lines by division, revenue by business unit, suspense and clearing accounts, intercompany balances, cash and working capital, and the balances behind any covenant.
2. **Let the history accumulate.** Collection starts the moment a pair is watched and costs no effort. The dataset becomes more valuable every week it runs.
3. **Set up the analytics database and the archive cycle.** Have both in place before the first month closes, so Wevo holds only recent history from the start.
4. **Begin with simple baselines.** Usual-range bands and run-rate projections, reviewed each week, build confidence and show where more advanced models will pay off.
5. **Add detection, then forecasting.** Extend to anomaly scoring across all watched pairs, then to rolling forecasts once a year or more of history is in place.
6. **Feed back what you learn.** Add pairs where questions keep arising, and retire those that never do.

A ninety-day proof of concept against your own GL extracts is long enough to put the first pairs on the watchlist, see the history build, and try the first baselines on real data.

11. Conclusion

Analytics and AI can transform how finance understands its own figures - but only when there is a clean, structured, time-stamped history for them to learn from. For most organisations, building that history is the hardest part of any finance AI initiative, and the part most likely to fail.

Wevo GL Analytics removes that obstacle. The structures are already defined, every figure is already reconciled to the ledger, and the watchlist records how the figures that matter evolve, every 15 or 30 minutes. Wevo creates that history; a dedicated analytics database keeps it, on resources of its own, inside your own estate.

> *Wevo collects the data. You choose the questions.*

To see the watchlist working against a real GL dataset, or to discuss which node pairs would give your organisation the most insight, contact **info@wevo.co.uk** or visit **wevo.co.uk**.

Copyright 2026 Professional Software Ltd. Distribute freely.