Closing the Day 1 Reporting Gap

A Power Query Consolidation Methodology for PE Carve-Outs

By Adams Doumbia

adamsdoumbia@hotmail.com | linkedin.com/in/adamsdoumbia | adamsdoumbia.github.io

1. Why this paper exists

This paper documents how a Power Query consolidation tool was built and deployed for a PE carve-out and how the same method can be applied elsewhere.

When a business is carved out, the acquirer needs consolidated financial reporting from day one of the transaction. Even buyers with their own EPM platform often face a gap before the carved-out entities can be properly integrated into their main consolidation environment. That gap can last anywhere from a few months to over a year depending on the complexity of the carve-out, the compatibility of the source ERPs and the buyer's integration roadmap. There are also situations where the buyer deliberately keeps the carved-out reporting separate, for example when the entities are short-lived, when audit or regulatory requirements demand a standalone view or when the integration cost simply does not justify the timeline.

The Power Query consolidation tool was built to address all of these scenarios. It runs entirely in Microsoft Power Query, requires no licence beyond standard Office 365 and can be stood up in a matter of weeks at a fraction of the cost of a full EPM implementation. The tool was designed to produce consolidation output functionally equivalent to OneStream for the perimeter in scope. For the EQT carve-out, the tool was operational from day one of the transaction and has been running monthly closes since.

The methodology was developed and deployed during the €2bn EQT carve-out from Adevinta (eBay Classifieds Group), part of the wider €14bn Blackstone and Permira take-private of the group. The Power Query approach was selected over both an EPM vendor proposal and an Alteryx-based alternative. It is shared here for practitioners facing similar situations.

2. The carve-out consolidation problem

PE-backed carve-outs are a particular kind of corporate transaction with a particular kind of finance problem. The buyer is acquiring a business unit that has spent years operating as part of a larger parent group, sharing systems, allocations and overhead. From legal close, the new entity needs to produce its own consolidated financials. From day one of trading, those financials need to be accurate enough for lender covenant reporting, sponsor value creation tracking and management decision-making.

The Day 1 reporting gap is the practical distance between this requirement and the operational reality. The parent's consolidation environment no longer covers the carved-out perimeter. The new entity does not yet have its own. Without a bridge, the finance team falls back on manual spreadsheet workarounds, which work for a month or two and then start to fail under the volume and complexity of monthly close.

For a PE sponsor, the gap is not just an operational nuisance. It threatens two things directly.

The first is the accuracy and timeliness of reporting itself. Late or inaccurate consolidation has consequences across the function, from covenant compliance to value creation plan delivery to board confidence. The finance team has no slack on these timelines.

The second is TSA exit. Each month the buyer remains dependent on the seller for management reporting, payables or other financial functions is a month of fees paid to the seller and operational leverage retained. Independent reporting infrastructure is the precondition for cutting these dependencies.

These are the conditions that frame the rest of this paper.

3. Build-vs-buy decision framework

Most acquirers procure an enterprise EPM to consolidate post-deal. OneStream, Oracle or Tagetik are the dominant choices for groups with sustained complexity. They are the right answer for a steady-state group with multi-year horizons, dedicated finance systems support and budget headroom for licence and implementation fees.

In a carve-out context the timeline does not allow it. The buyer typically has four to six months from TSA signing to the first independent close. EPM procurement alone (vendor selection, contracting, internal approvals) runs two to three months. The implementation work after procurement runs another six to twelve months. Total time from project start to go-live is eight to fifteen months. The arithmetic does not work.

The alternatives that get evaluated tend to fall into two categories. Mid-market CPM tools (Vena, Planful, Workday Adaptive) shorten the timeline but still require procurement, licensing and onboarding. Workflow automation tools (Alteryx, Tableau Prep) handle the transformation layer well but require additional licences, separate environments and a learning curve for finance teams trained on Excel. A Power Query approach has none of these dependencies. The deployment can begin from the moment an SAP extract or general ledger dump is available, without procurement, without IT involvement and without a new vendor relationship.

The cost economics are direct. The build was delivered at a fraction of the cost of a OneStream alternative (£415k or more in year one for implementation, first-year licence and partner fees and around £237k annually thereafter for licence and maintenance). The methodology integrates with a finance team already familiar with Excel and SAP. These are publicly available cost benchmarks at similar perimeter scale.

The argument is not that Power Query replaces an enterprise EPM in the steady state. It does not. It is fit-for-purpose for the carve-out window, where speed, cost and flexibility matter more than long-term platform investment. For PE sponsors, that flexibility translates directly into faster TSA exit, which PwC analysis suggests can deliver 5-7% business value uplift on top of the avoided EPM implementation cost.

4. Architecture overview

The methodology is built on a three-layer pipeline pattern. Each layer has a distinct purpose and sits behind a clean interface from the layers above and below. This layering is what allows the model to be operated, audited and extended without rewiring the whole structure.

Layer 1 (Ingestion) receives raw data from the source ERPs and from the journal adjustments controllers post separately. The methodology works with any source ERP. At Adevinta, the entities on SAP feed the pipeline through an extract from the Journal Entry Analyzer in S/4HANA. The entities on Navision feed it through a trial balance the controllers convert into the same internal template. Adding a third source ERP would not require any new design. The new entities' source data is converted into the same internal template, the same way Navision is handled today.

Layer 2 (Mapping and consolidation) translates the source data into the group reporting structure and applies the consolidation logic. The chart of accounts is mapped, dimensions are translated, currency translation is applied where needed, intercompany elimination runs (Trade and Investment), top-side adjustments are posted and carry-forward processing handles year-end balance roll-over. The consolidation logic is configured through mapping tables alongside the queries, not embedded in code, which is what makes the system portable between deployments.

Layer 3 (Output) produces the consolidated dataset that downstream reports consume. Historical actuals from prior years are retained in the same dataset, providing trend, comparative and audit continuity. A dedicated publish step writes the output to a flat file in a separate folder. Reports never read the live model. They read the published flat file, which is refreshed on a defined cadence and frozen between publishes.

The model handles multiple scenarios in parallel. Actuals, Budget and Forecast all run through the same pipeline. Additional scenarios (rolling forecasts, sensitivity cases, restated history) can be added without architectural change. Comparative reporting across any combination of scenarios and time periods works from the same model with no separate planning environment required.

The deployed perimeter started at seven entities and was reduced to four following the divestment of three entities to Scout24 ahead of the EQT transaction. The system absorbed this without redesign. The three divested entities were excluded from both live consolidation and historical actuals through configuration changes to the Perimeter table and rule overrides.

The full architecture is shown in Diagram 1.

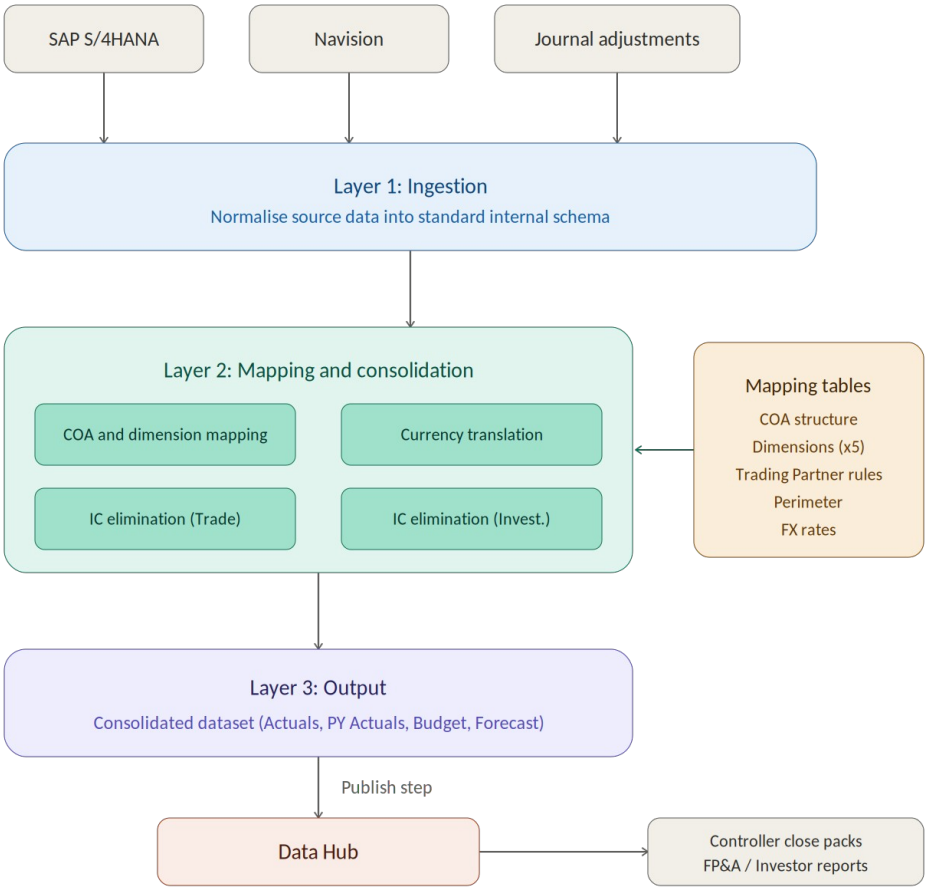

Diagram 1: Logical architecture

5. The operating model

The system sits inside a single shared drive structured into three folders, each with role-based access controls.

The **Operator folder** is restricted access. Only the operator can write to it. It contains the live model, the mapping tables, the dropbox folders for source ERP extracts and journal adjustments, the period repository for backups, the year-end frozen archive and the historical templates from prior years. Everything that the model needs to read or that the operator needs to maintain lives here, behind a single access boundary.

The **Data Hub folder** is viewer access for everyone in the function. It contains a single file: the published consolidation output. The data lands here through a publish step run by the operator after each refresh of the live model. No Power Query logic lives in the Data Hub. No source dependencies. No connection to the live model. It is a pure data container, refreshed only when the operator runs the publish step and frozen between publishes. Controllers and FP&A connect their reports to this file rather than to the live model. This pattern delivers four things: report refresh performance (consumer reports read static data in seconds rather than re-running the full pipeline in minutes), concurrent access for the team without file-locking conflicts, stability of consumer reports between publishes and a clean audit trail of when each version of the consolidated data was signed off.

The **Reporting folder** is contributor access. It contains separate subfolders for the different report types (controller close packs, FP&A management reporting, investor reporting). Reports in here read from the Data Hub. Anyone with access can build, modify or refresh their own reports without any risk of touching the underlying model.

The close cycle runs on a two-pass pattern. Both passes produce a complete consolidated result. The first pass shows the consolidation based on SAP source data with automatic intercompany elimination. Controllers post entity-level adjustments through the journal adjustments template. The consolidation team posts group-level entries (for example PPA, IFRS adjustments and fair value entries) through the same template. The second pass adds all journal adjustments and republishes the result.

The full operating model is shown in Diagram 2.

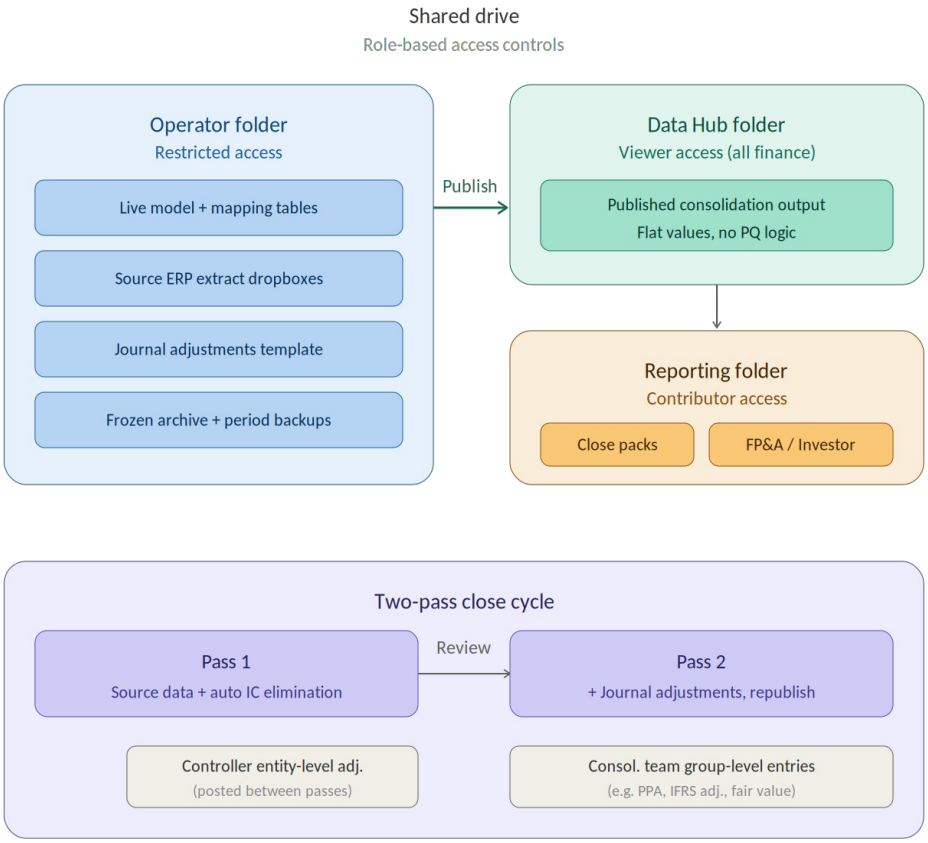

Diagram 2: Operating model

6. Trade intercompany elimination

Trade intercompany elimination is the standard pattern: when two entities in the consolidation perimeter trade with each other, the receivable on one side and the payable on the other have to be cancelled out, along with the matching revenue and cost. Without elimination, the consolidated accounts double-count the same economic event.

The methodology handles this in two passes. The first pass is the automatic elimination, the second is the residual plug.

In the first pass, the model identifies every transaction tagged as intercompany in the source data and pairs the sender entity with the receiver. Both the sender and the receiver have to be active in the consolidation perimeter for the period being processed. For matched pairs, the elimination entries are generated in a dedicated ledger that carries the reversed amounts.

The second pass handles the residuals. Even with disciplined intercompany booking, real intercompany positions never net perfectly to zero. Foreign exchange differences, cut-off timing on month-end transactions and rounding all create small mismatches between the two sides of an intercompany pair. The model groups every intercompany pair, sums the original amounts to find the net mismatch and posts a plug entry to a separate ledger.

The two-ledger separation is a deliberate design choice. Controllers reviewing the consolidated output can see clearly which eliminations are clean reversals and which represent residual mismatches that have been plugged.

The methodology pairs this with intercompany reports built for controllers. Each financial controller has a report that pulls from the Data Hub, surfaces every intercompany pair their entity is part of, shows both sides of each booking and flags any mismatch above a configurable tolerance threshold. Controllers can adjust the tolerance to suit their materiality threshold. The report highlights the rows that need attention before the operator runs the second pass of the close.

7. Investment elimination

Investment elimination is what removes the parent's investment in subsidiaries from the consolidated balance sheet, along with the matching equity on the subsidiary side, plus goodwill, plus dividends paid up the chain. Without it, the consolidated accounts double-count the equity of the group.

The EQT perimeter consists of fully owned entities, so the Full consolidation method applies throughout. The methodology eliminates three categories at investment level: equity (share capital, share premium, reserves), goodwill and dividends paid intra-group. Each is identified through the Group reporting structure of the COA rather than through hard-coded references, which keeps the methodology portable to other clients with different account ranges.

The elimination logic is perimeter-aware and hierarchy-aware. The Perimeter table carries a Hierarchie attribute for each entity, either Holding-Co or Subsidiary. The methodology applies three rules.

For a Holding Co sender posting equity, goodwill or dividends: eliminate if a partner exists, regardless of whether that partner is also active in the current perimeter. This handles cases where the Holding Co is recording its investment in a subsidiary that may have entered or left the perimeter at a different point in the year.

For a Subsidiary sender posting equity, goodwill or dividends: eliminate always, regardless of partner status. A subsidiary's own equity must be eliminated when it is consolidated, whatever its own partner activity.

For any other investment-tagged account (intercompany loans, deferred consideration, other capital movements): eliminate only if both sender and partner are active in the perimeter. This is the same symmetric rule that Trade IC uses.

The perimeter check is dynamic per row, so entities that join or leave mid-year are handled correctly without bespoke rules.

The elimination entries are posted to a dedicated ledger, separate from the Trade IC elimination ledgers, so the three classes of elimination are visible and auditable as distinct activities in the consolidated output.

8. The Trading Partner rule engine

OneStream handles conditional remapping through its rule expression syntax, applied during consolidation. The methodology described here implements the same pattern in Power Query. The remit at Adevinta was to deliver consolidation output functionally equivalent to what OneStream would produce on the same data and the rule engine is what closes the gap between raw SAP data and the structured output that consolidation requires.

The same rule engine is used for all five mapping dimensions. Trading Partner is the example below because it carries the most operational weight, but the pattern applies equally to the other dimensions (Verticals, Revenue Stream, Distribution Channel, Customer Group, Function).

A standard SAP extract carries a Trading Partner field with gaps and exceptions. Some accounts need their trading partner forced to a specific entity regardless of what the source field says. Some entity-account combinations need a different partner than the SAP default. Some accounts need their partner cleared entirely. Some need a partner derived from the account itself.

Each row in the Trading Partner mapping table carries a rule expression and a target value. The rule expression uses a compact syntax with three segment types:

> **E#[entity-list]**: matches if the row's entity is in the listed set
>
> **A#[account-list]**: matches if the row's account is in the listed set
>
> **I#[partner-pattern]**: matches if the row's intercompany partner matches the pattern (specific code, wildcard or empty)

Segments are separated by colons and evaluated as AND-clauses. A rule like E#[1003;1005]:A#[E11001;E12001]:I#[*] fires when the entity is in {1003, 1005} AND the account is in {E11001, E12001} AND the partner is any non-empty value. When the rule fires, the row's Trading Partner field is overwritten with the target value defined for that rule.

The engine evaluates rules in declared order. The first matching rule wins. Rows that match no rule pass through with their original Trading Partner unchanged. New exceptions are added at the top of the table, the existing logic continues to work and the rule order itself documents priority.

The same engine handles trading partner reassignment for Trade IC and equity counterparty mapping for Investment Elimination.

Embedding a small domain-specific language (DSL) inside Power Query is what makes the methodology genuinely replicable. Rules sit in a maintainable table, not hard-coded into the model. Adding new exceptions, modifying priority or porting a rule set from another deployment is a configuration change, not a code change. Where the methodology is being deployed alongside or in place of an existing OneStream environment, the rule sets can be ported into the mapping table with limited rework.

9. Multi-currency translation

Multi-currency translation in the methodology rests on five components.

The first is an FX rates table held alongside the other mapping tables. It carries one row per period, currency and rate type, with the rate value applied.

The second is rate type assignment by account range, governed through the COA mapping. Standard IAS 21 treatment applies: P&L accounts at Average rate, balance sheet accounts at Closing rate, equity accounts at Historical rate at the date of original transaction. Retained Earnings carries the prior closing balance forward at the prior closing rate, with current year P&L movement added at Average.

The third is the translation step itself, a transformation between Layer 1 and Layer 2 that looks up each row's source currency and rate, multiplies the local amount by the rate and outputs the group-currency translated value. The methodology translates from company currency to group currency at consolidation time, relying on the source ERP to have correctly converted transaction currency to company currency at posting time. This is the standard pattern for EPM-class consolidation tools and is appropriate where the source ERP enforces consistent currency handling at posting, as S/4HANA does. The model retains both the local amount and the group amount columns, so any consolidated value can be drilled back to its source. Adding a new currency requires only an additional set of rows in the FX rates table. The same translation logic handles any currency.

The fourth is the Cumulative Translation Adjustment. Translating P&L at Average and balance sheet items at Closing creates a mathematical imbalance: the balance sheet does not balance after translation. The plug is posted to a CTA account in equity. The methodology calculates the CTA per entity per period and accumulates it across periods, so the consolidated balance sheet balances after translation and the cumulative impact of currency movements is visible as a single equity line.

The fifth is historical rate management for equity. Share capital, share premium and pre-acquisition reserves are held at the rates on the dates they were originally booked, not retranslated each period. For new entities formed within the consolidation perimeter this is straightforward. For entities with pre-existing equity history, the historical rates are loaded once as static reference data.

Together these five components support multi-currency translation across any perimeter.

10. Defensive engineering

A consolidation tool that works on a clean dataset in a development environment is not the same thing as one that works in production over months and years. The difference is in the failure modes that emerge once real users, real data drift and real perimeter changes hit the model.

Two design choices protect the deployed system beyond what the operating model already covers.

The first is **the Frozen Archive for prior-year data**. Closed years are written out as static values into a separate archive workbook, removed from the live pipeline. Without this, the live model would reprocess every period of every closed year on each refresh, with refresh time growing linearly as historical data accumulated. The Frozen Archive keeps the live processing volume bounded, so refresh performance stays stable as the methodology runs over multiple years. Year-end snapshots also serve a compliance purpose: a permanent, immutable record of the consolidated position at each reporting date.

The second is **the error log**. It scans the data after dimension mapping has been applied and identifies rows where source values exist but the target group reporting value is missing. Each row in the error log identifies which mapping table needs updating.

The error log serves two purposes:

> **For the operator**: it catches new GL accounts, dimensions or other source values that have not yet been mapped, before they can drop out of the consolidation silently.
>
> **For the finance team during month-end close**: it acts as a feed of data drift, surfacing new GL accounts and new dimension values that have appeared in the source data during the period and which may affect controller reporting.

These two choices, alongside the publish gate built into the operating model, are what allow the methodology to be deployed without a dedicated systems support function behind it.

11. Implementation roadmap

A deployment of this methodology runs around 15 weeks from initial scoping to go-live. The phasing reflects the layered architecture: foundation first, then consolidation logic, then reporting integration, then validation and readiness. The architectural work itself does not repeat between deployments. The methodology, the operating model, the elimination patterns, the rule engine and the defensive engineering are reused. What each new deployment does is configure the system for the client's chart of accounts, perimeter, dimensions and reporting needs, with additional Layer 1 work where the source ERP requires it.

Weeks 1 to 4 are **foundation and data architecture**. Knowledge transfer from the existing finance team and the predecessor reporting environment. Source ERP analysis, including confirmation that the source carries intercompany identification at transaction level (Trading Partner field in SAP or equivalent in other ERPs). Where intercompany identification is not native to the source, a derivation layer is built into Layer 1 to fill the gap. Chart of accounts and dimension mapping configured for the new context. Initial build of Layer 1 (ingestion) and Layer 2 (mapping and consolidation logic) for the client perimeter. By the end of Week 4, the model is producing consolidated output from source data.

Weeks 5 to 8 are **core consolidation logic**. Trade IC elimination configured for the perimeter's intercompany structure. Investment elimination configured for perimeters that require it. Top-side adjustment posting. Scenario handling for Actuals, Budget and Forecast. Carry-forward processing for year-end. By the end of Week 8, the model produces a complete consolidation including all elimination patterns and supports comparative reporting across scenarios and periods.

Weeks 9 to 11 are **reporting integration**. Data Hub publish step. Controller close-pack reports, FP&A management reporting templates and any bespoke investor reporting required by the deal context. By the end of Week 11, every report consumed by the function is connected to the Data Hub and refreshes correctly.

Weeks 12 to 15 are **validation and readiness**. Reconciliation against the predecessor environment to confirm consolidation output is functionally equivalent. Sign-off cycles with controllers and FP&A. Governance and documentation finalised. Training delivered. Final readiness review. Go-live.

Hypercare follows go-live as a client-scoped support phase. Typical PE-backed deployments cover one to three closes of hypercare, during which the operator is available on-demand to controllers, fixes are applied directly to mapping tables and the model and lessons are captured in the documentation. Hypercare duration is a business decision rather than a methodology requirement, scoped to the client's confidence level and the perimeter's complexity.

The cost economics are direct. A fraction of the cost of an EPM vendor alternative (£415k or more in year one, £237k or more annually thereafter). The differential compounds with every subsequent year of operation and accelerates further when the methodology is reused across a PE portfolio of carve-outs.

12. Replicability and PE portfolio implications

A methodology becomes valuable to a PE house only when it can be redeployed across the disposal pipeline rather than rebuilt for each transaction. The methodology described here is designed for that.

Three things make it genuinely replicable rather than nominally replicable.

The first is **structural separation between architecture and configuration**. The architecture (the layered pipeline, the operating model, the elimination logic, the rule engine, the defensive engineering) is a fixed asset that does not change between deployments. The configuration (chart of accounts mapping, dimensions, perimeter, trading partner rules, FX rates) is what each new deployment populates with client-specific values. This is the same separation that mainstream EPMs maintain between their platform and their configuration. It is what makes a 15-week subsequent deployment realistic.

The second is **proven adaptability under perimeter change**. The Adevinta perimeter reduced from seven entities to four during the engagement, following the divestment of three entities to Scout24 ahead of the EQT transaction. The system absorbed this through changes to the Perimeter table and rule overrides, with no redesign of the model itself. Both the live consolidation and the historical actuals were correctly adjusted to exclude the divested entities. A methodology that handles perimeter change in production is meaningfully different from one that handles only the configuration it was originally built for.

The third is **the methodology as a prototype for the eventual EPM**. PE-backed groups commonly carry the question: do we eventually move the carved-out entity to a permanent EPM or keep it on the lighter infrastructure? Either path is supported. The methodology's configuration (the chart of accounts mapping, dimension structure, elimination rules, perimeter logic) can be lifted into a future OneStream or Oracle implementation. The carve-out window is then not just bridge consolidation, it is also a low-risk environment in which to validate the consolidation logic before it is committed to a permanent platform.

For a PE house with three or four disposals planned over a 24-month horizon, the methodology becomes portfolio-level infrastructure. Each deployment captures methodology value at the same defined effort, against the same EPM cost benchmark. The cumulative saving over a multi-disposal pipeline is substantial.

13. Closing

This methodology was developed and deployed during the €2bn EQT carve-out from Adevinta (eBay Classifieds Group), part of the wider €14bn take-private of the group by Blackstone and Permira. The deployed model has been running monthly closes since the EQT transaction completed, supporting the post-sale Spanish consolidation perimeter across two source ERPs, three scenarios and historical actuals from the two years preceding the transaction. The Power Query approach was selected over both a OneStream-based EPM vendor proposal and an Alteryx-based workflow alternative, principally on grounds of speed, cost and configurability through the carve-out window.

The methodology is the author's own. The architecture, operating model, elimination logic and rule engine described here are designed for replication beyond the original engagement. They are shared in this paper for practitioners and PE operating teams facing the same Day 1 reporting gap.

The author is a senior finance consultant who worked with Adevinta alongside Protiviti as Adevinta's long-standing finance transformation partner. Available for senior consulting and interim engagements across the UK and Europe in finance transformation, Group FP&A, consolidation and PE carve-out work.

Contact: adamsdoumbia@hotmail.com | linkedin.com/in/adamsdoumbia | adamsdoumbia.github.io