Agentic Performance Management

Stop renting and start owning your performance management stack.

APM is Enterprise Performance Management evolved for the AI era

Agentic Performance Management (APM) is a performance management approach built on open platforms and universal languages as an alternative to specialized EPM software. You own the data, the models, and the code. AI agents support the work of building, operating, and interrogating the stack rather than sitting on top of it as a bolt-on.

This paper is written for FP&A practitioners and EPM experts exploring possibilities in the AI era. It shares what we have learned from building and deploying APM solutions for clients ranging from $30M to $30B in annual revenue. We cover both the benefits and challenges of APM.

APM stands on three core principles.

- **First-party ownership.** Your data sits in your warehouse, your models in your files, your logic in your repository. Nothing is locked in a proprietary vendor's system. You have full ability to build and customize with AI and optionality to adapt.
- **Open and universal.** APM runs on Excel, SQL, Python, and open APIs rather than vendor-specific formula syntax that has no value outside one product. This helps you involve more people in performance management and compound organizational knowledge instead of isolating it to a few specialized experts.
- **Agentic.** Because the stack is code-first and built on open standards, agents can not only access and analyze data from the solution, but they can also build it, maintain it, validate it, and answer questions on the platform itself. This provides more surface area for agents to operate and deliver value.

What sets APM apart from EPM software?

APM presents a number of benefits vs. specialized EPM software:

It meets people where they are. Organizations have long been conditioned to look at tools like spreadsheets as the enemy. Software vendors have fought for your attention. But many teams have continued to use common tools like spreadsheets and decks to communicate.

Why? Because they are universal and familiar. APM is built to embrace this reality. AI platforms are trending to be the first new universal interface for the enterprise, even more than mobile. Will traditional spreadsheets or AI platforms win out? APM is designed for both sides, as we expect both to co-exist.

This compatibility is what makes APM a superior solution under many circumstances. If people are going to gravitate to spreadsheets and AI platforms in any case, performance management should embrace them instead of fighting against them.

Ownership builds equity and reduces long-term costs. One APM client realized 90% software savings with APM, worth over $1.2M annually. AI is breaking down technical barriers, which opens doors for open platforms to disrupt specialized and low-code tools that do not give you full access to build. Building is now competitive with buying in many cases. While you should not try to build your ERP or HRIS system, performance management is much more replaceable.

With true disruption, you should get both better performance and lower costs for 10x results. Building and owning your stack realizes that value by eliminating escalating and recurring costs. AI follows the same logic: in a closed platform, AI is a premium feature you rent at the vendor's markup, and they control the economics. In APM, AI is a capability embedded in an asset you own, and you can choose how you deploy it.

The skills transfer. The same principles of EPM do not need to be reinvented for APM. Top-down, bottom-up, integrated planning, short-term, long-range, driver-based, zero-based, multi-dimensional: the concepts and judgment are the same. What APM does is take off the technical limiters that you inevitably run into with the best EPM tools. Those limiters where you are forced to find creative workarounds, stretch the system, or go back into spreadsheets or custom-built applications. With the power of full code opened by AI, there are virtually no technical limitations. And the skills involved are all transferable. Agentic coding, systems thinking, a product mindset: these are all skills that will retain long-term value in the AI era.

Why we designed the APM approach

We have spent years implementing Anaplan, Oracle Hyperion, Planful, and Workday Adaptive Planning. We know these tools well, and we know what they do and don't do well. When generative AI arrived, we researched the ecosystem for a next-generation EPM vendor that represented genuinely disruptive innovation. We did not find one.

Pigment is a clear upgrade to Anaplan and is much more compatible with AI through their MCP, but it's still a specialized EPM tool at the end of the day, one with a proprietary engine, proprietary syntax, a proprietary interface, and a long-term contract. An MCP lets AI reach into the data, but the engine and context still belong to the vendor.

Meanwhile the pain points our clients faced with traditional EPM tools kept intensifying:

- **Rapidly rising software costs**, driven by opaque platform fees plus per-seat and per-use-case licensing that escalates at every renewal. Baked-in AI costs will exacerbate this further as vendors take margin on top of escalating usage prices.
- **Product ceilings** on granularity, data volume, and customization that force workarounds on the most important planning use cases.
- **Dependence on specialized resources.** Only a select few certified model builders become the bottleneck and the key-person risk. Many avoid being pigeonholed as the 'EPM Tool Administrator'.
- **Limited executive adoption.** Executives rarely log into planning tools, if ever. Information still travels by Excel, PowerPoint, and live presentations. EPM tools are like drops in a sea of unused dashboards that may have valuable information but no path to discovery.

- **Reversion to Excel** the moment the tool cannot flex in the right way or fast enough, which recreates the ungoverned sprawl the tool was bought to fix.
- **Limited AI capability**, because a vendor's copilot can only see what the vendor exposes, on the vendor's timeline, and at the vendor's price.

APM is our answer to these challenges. To address them, it is intentionally not a product. It is an approach: a documented way of assembling widely used platforms into performance management capabilities that rival off-the-shelf options.

How AI is changing the technology landscape

APM is built around four structural shifts in how enterprise software is bought and used. Each one moves against the assumptions that cloud-era EPM tools were built on.

Code is becoming the most useful form of context. As AI lowers the barrier to reading and writing code, the most valuable representation of your business logic is code an agent can act on, not a configuration a person has to click through or proprietary low-code syntax. Logic increasingly belongs in repositories, where it is diffable, reviewable, and auditable, rather than in shared drives or vendor databases.

Software is going headless. Buyers want fewer interfaces and fewer logins, not more. Durable enterprise software value sits in the data and context layer. APM does not try to win executive attention with its own front end; it can feed whichever AI interface decision-makers already gravitate toward. This is the market shift behind the compatibility argument above: the interfaces that win are the universal ones.

The pendulum is swinging from renting back to owning. Buy versus build was always a misnomer in enterprise SaaS. You were not buying, you were renting, and your core planning process was subject to a landlord. AI adoption changes processes and roles, not just tooling, and the proper investment looks more like a capital investment than an operating expense.

Nobody knows where this market lands. Not in six months, and certainly not in two years. The rational response is optionality: move now on foundations that are safe under any outcome (your data, open languages, universal interfaces) without overcommitting to an AI winner that has not emerged or a specialized product that may be obsolete within the year. We've seen many examples of tools and harnesses that seemed compelling only to be completely disintermediated in a matter of weeks.

How APM is built for the AI era

We standardized on the Microsoft platform stack, with the AI layer deliberately left open, for several reasons.

- **Excel, not Excel-like:** Excel is still the universal language of Finance teams, and native integration with Excel is key.
- **One-stop for Data + Semantic Layers:** Fabric does a great job integrating the data and semantic layers and has broad adoption with Power BI.

- **Proven path for app-building:** Power Platform is a proven platform for apps, workflows, and now agents.
- **Agentic optionality:** Microsoft is taking a relatively open approach to AI, with paths to use other providers' models in Copilot or plug into other AI platforms.

The APM stack is organized in four layers. This vertical integration minimizes handoffs and maximizes speed from raw data to decision-making, with agents going the final mile.

Layer	Platform	What it adds
Data Layer	Microsoft Fabric	Consolidates and organizes data from key systems of record. OneLake for storage, pipelines for integration, a warehouse queryable in standard SQL.
Semantic Layer	Power BI, integrated in Fabric	Adds business context: measures, relationships, formatting, and row-level security. One governed layer drives both reports and agents.
Planning Layer	Microsoft 365, Fabric, Power Platform, or AI platform	Adds forward-looking projections, scenarios, and analysis. Excel for modeling, SharePoint for collaboration, Power Apps or custom-built apps for inputs and approvals. Excel survived the last two technology shifts. We do not expect this one to be different. Agents will be increasingly powerful for modeling on their own or as Excel harnesses.
Agentic Layer	Your choice of AI platform	Adds agents that deliver information in real time, build, and support solutions. Anthropic, OpenAI, Microsoft Copilot, or open-weight models, plugged into the context of the three preceding layers in addition to other systems of record.

APM is built to maximize context for agents. With the full stack of data in Fabric, an agent can see actuals, calculation logic, and forecast outputs together. That level of context produces better results than agents pointed directly at systems of record with no context, or vendor-specific agents that can only see what the vendor is willing to show. Fabric is incredibly open: you can download your entire Workspace in one zip file or sync your Workspace to GitHub or Azure DevOps where other AI context like skills will live.

Every organization is moving at its own speed with AI, depending on your natural technical aptitude and risk profile. APM can adapt to that speed. We are typically starting with simple reporting agents based on governed Semantic Models and the standard Power BI MCP. With this setup, agents can already reliably query and deliver information, displacing the need for pre-built Power BI dashboards. Even the much-maligned Copilot is now effective for these purposes.

Agent Type	Description	Examples
Serve	Answers business questions against a governed semantic layer, replacing pre-built reports and addressing ad-hoc questions in real-time. Reviews and monitors your solutions, providing design recommendations and assisting with support.	GTM Reporting Agent Fabric Monitoring Agent Model Validation Agent

Agent Type	Description	Examples
Build	Helps you to design and build forecasting models for business purposes and data architecture, pipelines, and scripts for technical purposes.	Fabric Architect Agent Pipeline Builder Agent Long Range Model Builder Agent
Operate	Helps you to proactively execute processes and perform work on your behalf.	BOD Deck Builder Agent External Research Agent Monthly Forecast Tracking Agent

From there, the sky is the limit in terms of agentic leverage: agents will not only serve you data but can increasingly build and operate on your behalf. Tech organizations are already moving away from hands-on coding, and this is possible for business teams as well.

The business case for owning vs. renting

Total cost of ownership is a major benefit of APM. Below is a cost stack for a scaled APM deployment:

Component	Annual cost	Note
Microsoft Fabric, two F64 reserved capacities	$120,064	Separate development and production. Usage-based; can scale up or down.
Power BI, 50 report builders	$14,400	Full report-building access.
Power BI, 250 viewers	Included	No viewer license required at F64 capacity and above.
Power Platform, 10 app builders	$2,400	Full app-building access.
Power Platform, 250 app users	$15,000	Per-app access.
Aimplan visual, 50 designers and 250 viewers	$45,000	Optional third-party input visual.
Total annual platform cost	**$196,864**	**Compare against typical packaged EPM spend of $750K and up at this scope.**

Microsoft list pricing before Enterprise Agreement discounting. Confirm current rates, SKUs, and region at the time of proposal.

Based on our experience, a typical EPM solution at this scale will easily run $750K+ annually. Costs scale up quickly with more users and use cases, stifling cross-functional adoption. When you account for the fact that Fabric not only replaces CPM tools but visualization, workflow, ETL, semantic model, and data management tools as well, the consolidation value is clear.

In terms of managing the solution long-term, APM resourcing looks much different. Modeling resourcing is moved back to the business. We recently plugged in a new revenue model for a hyper-growth client that their sales finance lead designed in Excel. This took a matter of hours. With Anaplan, this would have been a 6-8 week project to translate that Excel model. With this client it's not unlikely that they'll redesign this model within a year.

For a different client, we are ramping their internal technology team to manage their APM solution. The concepts of data warehouses, SQL, Python, and pipelines are naturally familiar to them, and we are able to ramp them in a matter of days. If they were trying to learn Anaplan from scratch, it would take months before they are capable, and they likely wouldn't even try in the first place.

The universality of APM means that the business and IT can collaborate and internal resources can be better utilized. You don't need to trade off institutional or domain knowledge for specialized tech knowledge, which is usually not a great tradeoff. If the business and IT are open to a healthy working relationship, APM provides common ground instead of another silo.

The commercial aspects of the software are also different. If you're a Microsoft shop considering APM, you're likely on M365 already. You're likely already on Fabric for Power BI. If not, Fabric has free trials. All of these tools can be procured on a month-to-month basis, self-serve, with no long-term contracts. This is a huge breath of fresh air compared to the traditional enterprise SaaS model.

What are the downsides of APM?

Just as you can think of the upside of APM in terms of owning a house, you can think of the downsides along similar lines. And like owning vs. renting, the answer and rationale are different for every organization.

With great power comes great responsibility: we uttered this phrase often with Anaplan, and APM is no different. APM provides maximum power, flexibility, and customizability. In some cases, this can be counterproductive. If you don't want the responsibility of maintaining your house and are willing to operate within constraints, APM is probably not the best fit.

Technical breadth: APM scope goes beyond FP&A into what would traditionally be considered BI or data engineering. This vertical integration is what unlocks more speed by reducing handoffs needed from raw data to delivering it to decision-makers. Some organizations have already pushed these boundaries and have ample technical aptitude within business teams. Some level of aptitude is important for APM.

IT buy-in: APM requires cooperation and collaboration with IT. Unfortunately, we see many cases where business teams go to SaaS solutions to avoid IT interactions. In these cases, APM is unlikely to pan out. Otherwise, APM is a great path to get business and IT teams back working together instead of in siloes.

Open-endedness: Fabric was not built for EPM specifically, so there's not a clear out-of-the-box path to making it work for EPM purposes. This is where OVG comes in: we provide you with the playbook, blueprints, and initial foundation, playing the role of architect, designer, and general contractor. Once you have a functional core in place, it's possible to take long-term ownership.

Who is the best fit for APM?

Our clients range from $30M to $30B in annual revenue. Where APM works depends more on organizational qualities than size. APM may be a great fit if:

- You are already using Microsoft at least for productivity.
- Your IT organization is open to building performance management on Azure and Fabric and APM won't be running counter to an existing BI strategy.
- Your team has previous experience with EPM tools and is wanting for more flexibility or capabilities.
- Your organization is pushing for AI adoption and has an AI platform that APM can plug into.

How to start your APM journey

Our clients' APM journeys are as varied and bespoke as the solutions themselves. With the virtually unlimited potential scope of APM, it's important to take a phased approach and realize real value within a matter of weeks before proceeding further. Below are a few common starting points:

- An organizational assessment to understand pain points and areas of improvement and a Fabric / Copilot proof of concept.
- A core FP&A deployment starting with headcount, operating expenses, 3-statement modeling, and reporting.
- A core analytics deployment to establish the data foundation, semantic, and agentic layer for a specific domain or use case.

If initial value is proven, the solution can then be expanded across use cases, functions, or both. In this way, APM is more of a journey and new way of working vs. a standalone software implementation. Each phase is a decision point rather than a commitment to the next one, and long-term you can run the solution yourself, co-manage it with us, or have us operate it under managed services. In all three cases the asset stays yours.