

Companies Are Building Their Own Planning Systems Now. Here Is What We See.

July 2026

More mid-sized companies are replacing vendor CPM licences with planning applications built by AI coding agents. What drives the shift, what it costs, and where teams fail when they try it alone.

Executive Summary

Clients who pay for expensive CPM platforms are rebuilding those systems as their own applications, with AI coding agents like Claude Code or Codex, and cancelling the recurring licences. Clients who are only now leaving Excel and looking at their first planning tool increasingly skip the vendor step entirely and ask us to build instead. A third group stays conservative. They keep the vendor product or want something off the shelf, and that is a legitimate position.

But the first two groups are growing quickly, and the reason is not that AI is fashionable. The reason is that the vendor model never really fit mid-sized companies in the first place.

What is driving it

- No two companies plan the same way. Vendor planning products are development platforms, not finished products, so they need heavy customisation.
- That customisation needs rare, expensive consultants. You end up with a complex application that only the vendor and those consultants can maintain.
- Internalising it afterwards is hard. Your team ends up renting its own planning logic.
- Licences are subscription-based, from roughly 15,000 USD per year at the entry level and up. Planning is a communication process, so you need licences for everybody who plans or budgets, and that is what makes the bill grow.
- A vendor route realistically takes six months from first internal alignment to a working system. An AI co-built route takes eight weeks.
- Over three years the co-built route now costs less than staying on the platform you already have.
- The documentation writes itself, which sounds like a footnote and turns out to matter more than almost anything else.

Where it goes wrong

Teams that try it entirely alone usually fail on four things: data integration, technical architecture, engineering discipline (version control, CI/CD, continuous testing), and translating business needs into instructions an AI can act on. Two years of using ChatGPT as “Google on steroids” does not prepare anybody for these. It is a different way of working, not a lookup.

Our position

We think this shift is right for most mid-sized companies. We also think doing it without experienced help, two months before the budget cycle, is a bad bet. We have done hundreds of planning projects on vendor software and now on custom AI applications, and the honest summary is: the application is only as good as the person who built it.

The Vendor Product Was Never Really a Product

Every planning system we have ever implemented was different. There are always commonalities: a cost centre hierarchy, a P&L structure, a headcount model, someone who insists on their own currency logic. But no two clients plan alike. Your business is organised the way it is for reasons that took thirty years to accumulate.

The vendors know this. That is why their products are, in practice, development platforms. You do not buy a budgeting system. You buy a toolkit and then pay somebody to build a budgeting system inside it.

This is worth sitting with for a moment, because engineering companies in this region understand it immediately. If you build special-purpose machines, you would never buy a generic machine and then pay the manufacturer an annual fee for permission to configure it into the machine you actually needed. You would build to specification. Yet in finance systems, that is exactly the deal that has been normal for twenty years.

And the people who can do that configuration well are rare. We know, because we employ some of them and we struggle to hire more. So you get a complex application, maintained jointly by a vendor and a handful of expensive specialists, which your own team cannot fully take over.

Then the invoice arrives every year regardless.

Why the Licence Model Hurts Mid-Sized Companies Specifically

Planning is a conversation. The cost centre owner in production, the regional sales manager, the plant controller, the HR person doing headcount: they all need to be in the system, or the process falls back to Excel attachments and the whole point is lost.

Every one of those people needs a licence. A typical budgeting process in a mid-sized company has around twenty participants. A platform fee of roughly 10,000 EUR plus twenty seats at 1,600 EUR each lands you at 42,000 EUR a year, and the seats are the larger half of that. Note which part scales: not the platform, the people. Add three plant controllers next year and the bill moves again.

That is all before anybody has configured anything.

Nobody at a family business enjoys explaining to the shareholder why the company rents its own budgeting process in perpetuity.

What Changed

Large language models and AI coding agents dropped the barrier to entry for building software. Not to zero, that claim is nonsense, but far enough that building a planning application is now sometimes easier than going through a tender, an integration project and a licence negotiation.

So the trend is not really about AI being clever. It is that the alternative finally became cheap enough to put next to a vendor quote and compare honestly.

The Numbers

Here is a model for a mid-sized manufacturer: around twenty planning participants, a Dynamics 365 or SAP source system, a normal cost centre and P&L planning scope with headcount and capex.

Every assumption is listed so you can argue with it. That is the point of showing it.

Cost element	Vendor CPM route	AI co-built application
Software, year 1	€42,000 (platform fee €10,000 + 20 × €1,600)	—
Hosting + AI agent seats, per year	—	€9,600
Implementation / build (one-off)	€98,000 (70 consultant days @ €1,400)	€49,000 (35 consultant days @ €1,400)
Internal effort during project	€47,500 (1.0 FTE × 6 months)	€23,750 (1.5 FTE × 2 months)
Year 1 total	€187,500	€82,350
Annual change & lifecycle support, from Y2	€21,000 (15 consultant days)	€14,000 (10 consultant days)
Internal ownership, from Y2	—	€14,250 (0.15 FTE)
Licence uplift	5% per year, compounding	none
3-year cumulative	€319,905	€158,050
6-year cumulative	€536,180	€271,600
Steady-state annual run rate	€67,305 and rising	€37,850 and flat

Internal FTE costed at €95,000 fully loaded. Consultant day rate €1,400 blended. Vendor implementations in this size class run 60 to 80 consultant days; 70 is the midpoint. We assume the AI build takes half of that, which is a deliberately pessimistic assumption on our side.

The €9,600 running cost is hosting at €600 a month plus two Claude Team Premium seats at USD 100 a month for your own people. Our agent licences are our cost, not yours, and they are not in this table.

Year one saving: about €105,000. Three-year: about €162,000, or 51%. Six-year: about €265,000, or 49%. Call it half.

The largest single line is not the implementation, it is the licence. Over three years you pay €132,000 in subscription fees before anybody has improved anything.

There is one more comparison worth running, and it surprised us the first time. Doing nothing at all, meaning you keep your current platform, pay the licence, absorb the usual

change requests and build no new system, costs about €174,000 over three years. The co-built application costs €158,000 and you own something at the end of it. Sitting still is no longer the cheap option.

That said, I want to be straight about the rest, because the market is full of people promising 90% savings. The cost advantage is real and worth having, but you still pay for expertise, and you should. Three other things change more than the money:

Speed. Eight weeks instead of six months. That is the difference between having a system for the coming budget round and waiting a year for the next one.

Ownership. The application, the data model and the source code are yours. No renewal conversation, no licence audit, no vendor roadmap deciding what you are allowed to plan next quarter.

Fit. It does what your planning process does, rather than what the platform permits.

The Six-Month Problem

The vendor timeline is what usually forces the decision, so it is worth laying out honestly.

One to two months of internal alignment, tender and procurement. Then a few weeks of functional and technical design. Then three to four months of honest implementation with a couple of UAT cycles. Best case, the system runs after six months, and you are aiming at the planning cycle after next.

Meanwhile the wake-up call is almost always the same: the budgeting cycle is around the corner, and in a couple of months the planning system needs to work, be connected to the local source systems, be tested, and be documented.

Those two timelines do not meet. That is the moment most of our conversations start.

The Documentation That Writes Itself

This is the least exciting advantage on the list and possibly the most valuable, so it gets its own section.

On a vendor project, documentation is a separate deliverable with its own line in the budget. It is the first thing cut when the timeline slips, and the last thing anybody volunteers for. Two years later nobody remembers why the allocation rule works the way it does, and the consultant who configured it has moved to another firm.

With an AI build, documentation is a by-product rather than a phase. The agent writes the data dictionary, the technical documentation, the functional description and the change log while it builds, and regenerates them when the logic changes. Nobody has to sit down on a Friday afternoon and write up what they did in March. It is simply there, and it is current.

The practical consequence is bigger than the time saved. The reason companies cannot take over their vendor-configured planning system is rarely the licence. It is that nobody ever wrote down how it works. Solve the documentation problem and internalisation stops being a wish and becomes a handover.

Where Teams Fail When They Do It Alone

We have now seen enough attempts to be specific. The failures cluster in four places, and none of them is about the AI:

- **Data integration.** Getting clean, reconciled actuals out of Dynamics 365, SAP HANA, or the local German ERP that nobody outside your industry has heard of. This is where most projects quietly die. It is unglamorous, it is where the domain knowledge lives, and an AI agent cannot infer your posting logic from the table names.
- **Technical architecture.** The wrong choice early is invisible for six weeks and then fatal. Applications built without an architecture work beautifully in the demo and fall apart around iteration fifteen.
- **Engineering discipline.** Version control, CI/CD, continuous testing. Skip these and you have not built an application, you have accumulated one. The first time you cannot reproduce last week's working version during the budget round, you will understand why this matters.
- **Explaining the business need to the AI.** This is the one people underestimate most. Prompting a coding agent to build a planning system forces you to state your own planning logic precisely: the drivers, the allocation rules, the exceptions, the approval flow. Most organisations have never written this down. The AI will not fill the gap, it will confidently build the wrong thing.

A team can absolutely learn all of this internally by building progressively harder applications. In our experience that takes six to twelve months of consistent effort. Which is fine, unless the budget cycle is in eight weeks.

And to be blunt about one thing: a couple of years of using ChatGPT as a better search engine does not transfer here. It is a mentality shift, not a skill you look up. It is also not something you pick up from a micro-course on vibe coding advertised on Facebook.

Our Honest Recommendation

Do it, but do it with someone who has done it before, and treat them as a co-author rather than a supplier.

What experienced consultants bring to an AI build is not typing speed. It is knowing how planning processes actually work, how to connect to your specific source systems, how to model data so it survives the third change of requirements, how to write functional and technical requirements that an agent can execute, and how to set up a development process that holds together after the tenth iteration.

That is a large body of knowledge, and the coded application is exactly as good as the person who built it. Which is the strongest argument both for doing this and for not doing it alone.

If you are on a vendor platform and the renewal is coming, or you are about to start a tender for your first planning tool, it is worth a conversation before you sign anything. We work with

companies like yours, and we will tell you if the vendor route is the better answer in your case. Sometimes it is.

Key Takeaways

- **Vendor planning products are platforms, not products.** You buy a toolkit and pay specialists to build your system inside it, then depend on them to maintain it.
 - **Per-participant licensing is the real cost driver.** A €10,000 platform fee plus twenty seats at €1,600 comes to €42,000 a year, or €132,000 over three years, before anybody configures anything. The seats are the part that grows.
 - **The cost saving is roughly half over three to six years** (51% and 49% in our model). Real and worth having, but not the 90% some people claim. Speed, ownership and fit matter more.
 - **Standing still is no longer the cheap option.** Keeping your current platform costs around €174,000 over three years against €158,000 to build and own a replacement.
 - **Time to value is the decisive factor.** Six months for a vendor route against eight weeks for a co-built application, with a budget cycle that will not wait.
 - **Documentation stops being a phase.** The agent writes and maintains it as it builds, which is what finally makes internalising the system realistic.
 - **The four failure modes are data integration, architecture, engineering discipline and requirements clarity.** None of them is solved by a better model.
 - **The application is only as good as the person who built it.** Which is why subject matter expertise, not AI access, is the scarce input.
-

Questions We Get Asked

Is it realistic to replace a CPM platform with an application built by an AI coding agent?

For planning scopes typical of mid-sized companies, yes, and we are doing it: cost centre and P&L planning, headcount, capex, sales planning, scenario comparison, approval workflow. Where it gets harder is heavy statutory consolidation, complex multi-GAAP requirements, or regulated disclosure processes, where the audit trail and the vendor's certification carry real weight. Those cases deserve a case-by-case answer rather than a trend.

How long does it take?

Around eight weeks to a first productive planning cycle, assuming the source systems are accessible and someone on the client side can decide things. The vendor equivalent is roughly six months including procurement. In consultant days that is about 35 against 60 to 80.

What does it cost compared with a vendor licence?

In the model above, roughly €82,000 in year one against €188,000, and about €158,000 against €320,000 over three years. The larger structural difference is the steady-state run rate: around €38,000 a year and flat, against €67,000 and rising with annual licence uplift. The €38,000 is hosting, two AI agent seats for your team, a small enhancement retainer and about 0.15 of an FTE internally.

Who owns the application afterwards?

You do. Code, data model and documentation. That is the point of the exercise. It also means you carry the ownership responsibility, which is why we build with version control, CI/CD and a test suite from day one rather than bolting them on later.

Does an AI-built system actually get documented properly?

Better than most vendor implementations, in our experience, because the agent produces the data dictionary, technical documentation and change log as it builds rather than as a separate work package at the end. That is also what makes handover to your own team realistic instead of theoretical.

Can our own team build this without external help?

Some can, and a few should. If you have a controller who codes, an architecture-minded BI lead, and twelve months before you need the system, then building internal capability by starting with smaller applications is a good strategy. If the budget cycle is two months away, that is not the moment to learn data integration, architecture and CI/CD at the same time.

What about our existing vendor investment?

Sunk cost is sunk. The only number that matters is what the next three years cost under each option, plus the switching effort. On our model that comparison is €174,000 to stay against €158,000 to build and own the replacement, which is closer than most people expect. Sometimes the answer is to keep the vendor platform for consolidation and build alongside it for planning. We have done that too.

Is this only for companies leaving Excel?

No, and the split is interesting. Companies moving off Excel for the first time now often skip the vendor step entirely. But some of the strongest cases we see are companies already paying for a CPM platform who know precisely what they need, because they have lived with the constraints for years.

Is there anything about this you would not put in a proposal?

One thing. This work is fun. Controllers who have spent fifteen years fighting Excel workarounds discover they can describe what they want and watch it appear an hour later. We have had to tell people to go home at the end of a build week. It is not a line item and I would not try to price it, but do not dismiss it either: a planning system the planners enjoyed building gets used differently from one that was delivered to them.

About the author

Ilya Fedorkov is Co-Founder and CTIO (Chief Technology & Information Officer) of Centida GmbH, a boutique consultancy focused on Planning & Reporting. He is responsible for Centida's technology and delivery architecture — from integrated planning and reporting platforms to AI-enhanced forecasting — and bridges the gap between finance logic and technical implementation. He supports clients across manufacturing, energy & utilities, telecoms and professional services.