

Replacing vendor planning tools with AI-built applications

Centida | Data. Intelligence. Decisions.

July 2026

Agenda

1. What we see in the market
2. Why the vendor model does not fit
3. What it costs and how long it takes
4. Where AI builds fail
5. How we would run it with you

Executive summary

What we are seeing across our client base, and what it means for your next planning cycle

The shift	The economics	The risk
Clients are cancelling CPM licences and rebuilding their planning systems with AI coding agents. Two groups are growing quickly: companies replacing an expensive platform they already own, and companies leaving Excel who now skip the vendor step entirely. A third group stays with vendor products, and for some of them that is still the right answer.	A co-built application costs about half as much over three years and runs eight weeks after kick-off, not six months. EUR 158,000 against EUR 320,000 for a 20-participant scope, and a run rate of EUR 37,850 a year instead of EUR 67,305 and rising. It also costs less than staying on the platform you already have, which is the comparison most people never run.	The application is only as good as the person who built it, and four things are where teams fail. Data integration, technical architecture, engineering discipline, and translating business needs into instructions an AI can act on. None of these is solved by a better model. Building the capability internally takes six to twelve months, which is more than a budget cycle gives you.

Three groups of clients, and two of them are moving

What we observe across our planning and controlling engagements

Replacing a CPM platform

- Already own an expensive vendor platform and know exactly where it constrains them.
- Rebuild the planning logic as their own application, then cancel the recurring licence.
- Strongest business case, because they can compare the new system against a live one.
- Typical trigger: a renewal quote, or a change request the vendor priced at six figures.

Leaving Excel

- First real planning system, migrating from workbooks that only two people understand.
- Increasingly skip the vendor evaluation entirely and go straight to a custom application.
- No legacy to unwind, so time to value is shortest here.
- Typical trigger: the budget cycle is two months out and Excel will not survive another round.

Staying with a vendor

- Prefer a supported product with a name on the contract, and that is a legitimate position.
- Often driven by group IT policy, audit requirements, or statutory consolidation needs.
- Also the right choice where the planning process is genuinely standard.
- This group is not shrinking, but it is no longer the default.

Why the vendor model stopped fitting mid-sized companies

One answer, three reasons it holds

For most mid-sized companies, a co-built AI planning application now beats a vendor CPM platform on time, cost and fit.

Vendor planning products are development platforms, so you pay twice: for the licence, and for the rare specialists who configure it.

Per-participant licensing scales with your headcount rather than with the value the planning system delivers.

AI coding agents removed the build barrier, so the alternative is finally cheap enough to compare honestly.

Heavy customisation is mandatory, not optional

Maintained by the vendor, not internalised

Planning is a conversation: everyone needs a seat

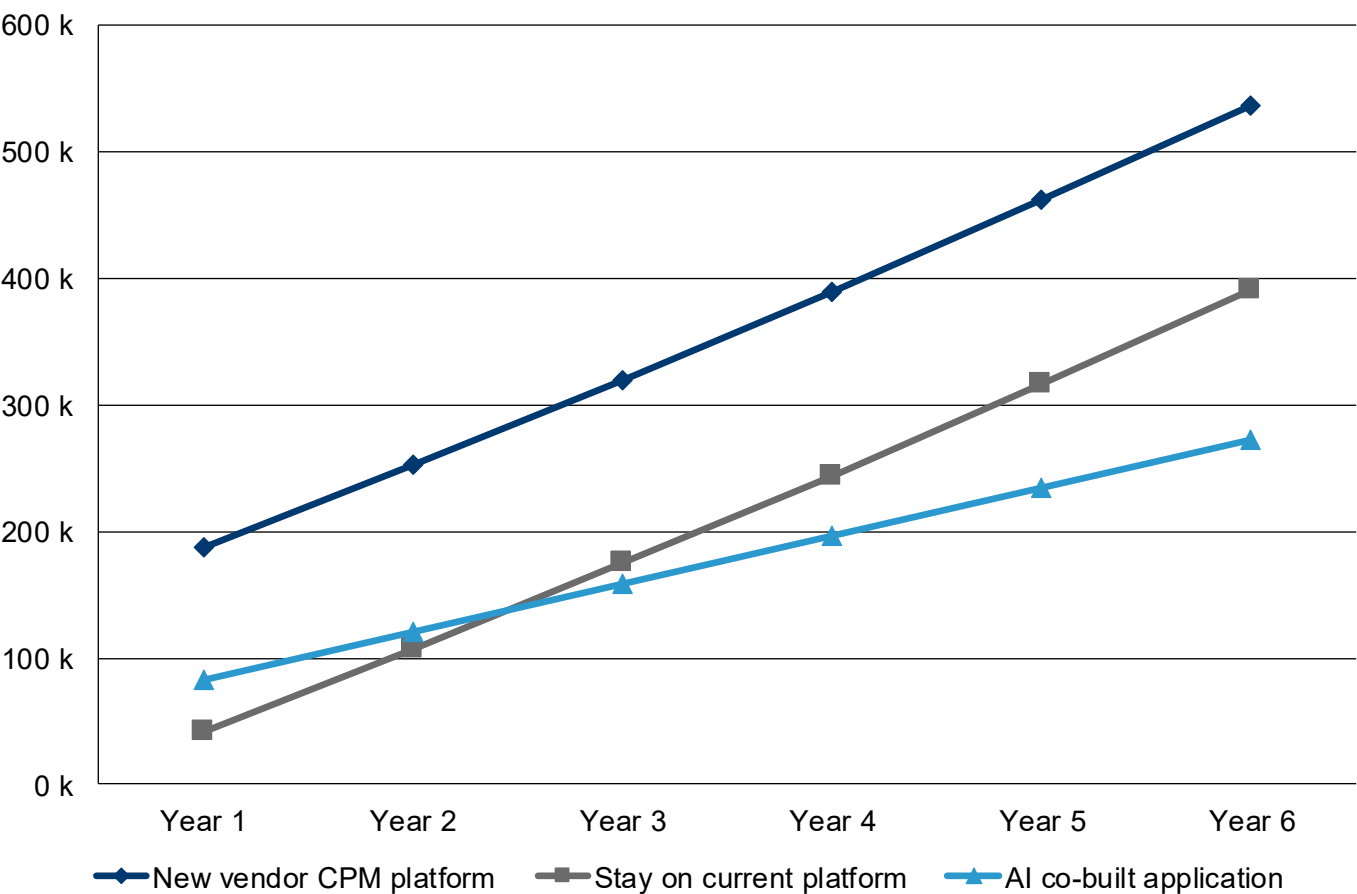
From USD 15,000 a year at entry level, and up

Eight weeks to a working system, not six months

The agent writes the documentation as it builds

Cumulative cost of ownership over six years

Mid-sized manufacturer, 20 planning participants, cost centre and P&L planning scope



Key takeaway

Building your own is cheaper over three years than staying on the platform you already have. Sitting still is no longer the low-cost option.

- Three years: EUR 158,050 to build and own it, EUR 174,405 to stay put, EUR 319,905 for a new vendor platform.
- The licence alone is EUR 132,000 over three years, before anybody has improved anything.
- Run rate: EUR 67,305 a year rising at 5%, against EUR 37,850 a year and flat.

Implication: the do-nothing case is not free. It just spreads the cost where nobody has to approve it.

Source: Centida cost model, July 2026 — 20 participants, EUR 1,400 consultant day rate, EUR 95,000 internal FTE, 5% licence uplift. Licence = EUR 10,000 platform + 20 x EUR 1,600. AI run cost = EUR 600/mo hosting + 2 agent seats; Centida's own agent licences excluded.

Three routes, scored against the decisions that actually matter

Typical mid-sized manufacturer with 20 planning participants and a budget cycle in Q4

Criteria	Stay on current platform	New vendor CPM platform	AI co-built application
Live for the next budget cycle			
Three-year cost below EUR 250,000			
Fits our planning logic without workarounds			
Documentation current and handover feasible			
Low delivery risk with no external partner			

Harvey balls: empty = does not meet · quarter = partially meets · half = meets · three-quarter = exceeds · full = best in class. The co-built route scores worst on delivery risk without a partner — which is precisely the gap an experienced co-author closes.

The six numbers a CFO asks for first

Co-built application against a vendor CPM route, same scope, same day rates

Time to first planning cycle

8 weeks

Against six months on the vendor route, including procurement

Year-one total cost

EUR 82k

Against EUR 188k for a vendor implementation

Three-year saving

EUR 162k

51% lower over three years, and EUR 16k below doing nothing

Steady-state run rate

EUR 37.9k

Flat, against EUR 67.3k rising at 5% a year

Licence paid over 3 years

EUR 0

Against EUR 132k of subscription fees on the vendor route

Consultant days to build

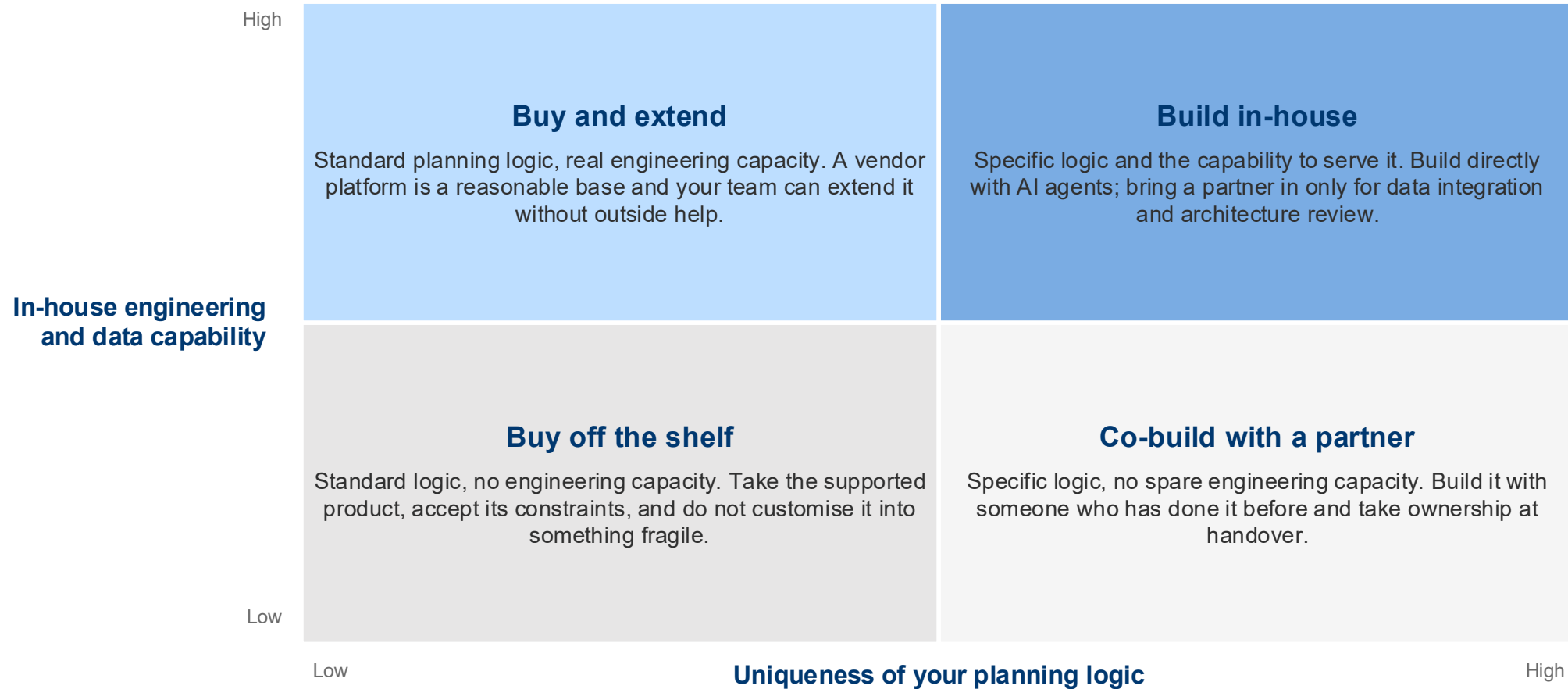
35 days

Half of a 60-80 day vendor implementation

Source: Centida cost model, July 2026. 20 planning participants; EUR 1,400 consultant day rate; EUR 95,000 internal FTE; 5% annual licence uplift. Illustrative — validate against your own scope.

Where you sit decides whether to buy, build, or co-build

Two dimensions that determine the right route — and neither of them is budget



Takeaway: most Black Forest manufacturers sit bottom-right — highly specific planning logic, no spare software engineering capacity. Co-building is the only route that closes both gaps inside one budget cycle.

Where AI planning builds fail, and none of it is the AI

The four risk areas we see when teams attempt this without help

1. Data integration

Getting clean, reconciled actuals out of Dynamics 365, SAP HANA, or the local German ERP nobody outside your industry has heard of. This is where most attempts quietly die. An agent cannot infer your posting logic from the table names, and this is exactly where the domain knowledge sits.

2. Technical architecture

The wrong choice early is invisible for six weeks and then fatal. Applications built without an architecture demo beautifully and fall apart around iteration fifteen, usually in the middle of the budget round when nobody has time to unpick them.

3. 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 a planning cycle, the cost of skipping them becomes obvious.

4. Requirements and prompting

Prompting an agent to build a planning system forces you to state your own logic precisely: drivers, allocation rules, exceptions, approval flow. Most organisations have never written this down. The AI will not fill the gap, it will confidently build the wrong thing.

Eight weeks from kick-off to a tested planning system

How we run a co-build, with your team owning the result at handover

Phase 1: Frame	Phase 2: Build	Phase 3: Harden and hand over
Weeks 1 – 2	Weeks 3 – 6	Weeks 7 – 8
Objective Planning logic, data model and architecture agreed before a single line of code is written. Key activities ■ Map the real planning process, including the exceptions ■ Fix the data model and the source-system integration design ■ Set up the repository, CI/CD pipeline and test harness Milestone: signed-off functional and technical design, week 2	Objective A working planning application connected to live actuals, built in weekly increments with your team in the room. Key activities ■ Build input forms, driver logic, workflow and reporting ■ Connect actuals from the ERP and reconcile against the close ■ Weekly demo to the planning participants, with changes taken the same week Milestone: first full planning cycle on real data, week 6	Objective The system is tested, documented and owned by your team rather than by us. Key activities ■ User acceptance testing with all planning participants ■ Performance, security and access review ■ Handover: repository, agent-generated documentation, runbook, two working sessions Milestone: production go-live and handover complete, week 8

Recommendation

Based on the cost, timing and risk analysis on the preceding slides

Co-build your planning application with Centida in an eight-week engagement, target the coming budget cycle, and keep the code, data model and documentation in-house.

Rationale	Expected impact	Risks and mitigations
Your planning logic needs heavy customisation on any vendor platform, and that customisation is the expensive part of the vendor route. A vendor route cannot be live before the coming budget cycle. Eight weeks can. Staying put costs EUR 174,000 over three years against EUR 158,000 to build and own the replacement, so doing nothing is not the cheap case either. You have the subject matter expertise. What you lack is spare engineering capacity, and that is the part we bring.	EUR 162,000 lower total cost over three years, EUR 265,000 over six, and none of the EUR 132,000 of licence fees a vendor route incurs in the same period. Planning system live in 8 weeks instead of 26, so the coming cycle runs on it rather than the one after. Documentation, data dictionary and change log generated by the agent as it builds, so no separate documentation work package and no stale handover pack.	Risk: source-system integration is harder than expected. Mitigation: integration design is completed and tested in phase 1, before any application build starts. Risk: the application degrades after we leave. Mitigation: version control, CI/CD and a test suite from week 1, plus a documented handover and two working sessions. Risk: your planning logic is not written down. Mitigation: phase 1 exists to write it down, and that artefact is valuable whichever route you choose.

Centida

Data. Intelligence. Decisions.

Centida GmbH | Hauptstrasse 52, 77704 Oberkirch, Germany | +49 7802 704 2332 | info@centida.com | Registration number: HRB 712503 | VAT ID: DE298062157 | <https://centida.com>