

# The roadmap to the AI enabled finance function

Article 3 of 6 · aicfopartner.ai · PDF issued 8 September 2026

## EXECUTIVE SUMMARY

- AI implementation is usually best managed through four connected workstreams: people, data, governance, and process and technology. Progress should depend on meeting agreed conditions, rather than dates alone. Costs and benefits should be measured throughout the programme.
- Start with a current state baseline and the minimum governance framework, so finance knows where the opportunity sits, what AI use is permitted and how ROI will be measured.
- Use a practice phase of roughly six weeks on lower consequence tasks to build the team's AI fluency and create bandwidth for the process redesign and data work that follows.
- With some capacity created, the team can begin the process and data work needed for more permanent automation. Build the data foundation use case by use case: agree authoritative data sources and definitions, map records between systems, reconcile the numbers, connect reliable feeds and assign ownership.
- Prioritise processes for automation, beginning with those that offer the greatest ROI, can scale and have fewer barriers to implementation.
- For each process to be automated, review how far the existing ERP and wider technology stack can already automate or improve it. Use what is already owned where it is fit for purpose, then buy or assemble to close genuine gaps and build only exceptionally.
- Prove each selected process automation in a bounded pilot before live use. Only work that meets its accuracy, control and economic thresholds should scale, with benefits and controls kept under review.

## 01 — THE ROADMAP AT A GLANCE

The strategy article sets the destination for the finance function's AI programme, names its owner and defines how return will be measured. This article turns those decisions into a practical delivery plan. It covers how AI fluency, data, sourcing and governance should develop, and the evidence needed before each stage moves on.

A programme of this kind is usually better managed through gates than a fixed calendar. A stage is complete when its exit conditions are met. The six week practice phase is the exception because a defined end date keeps the learning period focused. Data work, sourcing decisions and live deployment will otherwise vary with the organisation's systems, data quality and available capacity. Gartner reported in May 2026, based on a March survey

of 204 finance leaders, that 63 per cent of finance organisations said AI implementation had been slower than expected in 2025. Slower than expected is not failure, but it supports an evidence based approach to each gate.

The organisational change often starts before the technology does. The team needs enough capacity and confidence to change the work before more is asked of it. Mid sized functions can often move faster than enterprises because they have fewer systems to connect and shorter decision chains.

Subsequent delivery depends on four connected workstreams: people and fluency, data, process and technology, and governance and value. They will usually develop at different speeds, and progress in one can expose work needed in another. Each workstream moves through establish, prove, live and scale. The grid below summarises what each stage is intended to achieve. The rest of the article then develops those outcomes in practical terms.

| WORKSTREAM             | ESTABLISH                            | PROVE                                | LIVE                                | SCALE                       |
|------------------------|--------------------------------------|--------------------------------------|-------------------------------------|-----------------------------|
| People and fluency     | Approved tool and first tasks        | Six week practice phase              | Named owners and repeatable methods | Wider team capability       |
| Data                   | Sources, owners and definitions      | Priority data cleaned and reconciled | Reliable feeds and approved access  | Governed shared foundation  |
| Process and technology | Processes and pain points identified | Way of working and pilot             | Integrated live process             | Reusable components         |
| Governance and value   | Governance floor and baseline        | Test evidence and thresholds         | Go live controls                    | Benefits and control review |

## 02 — ESTABLISH THE BASELINE AND THE GOVERNANCE FLOOR

A sensible starting point is a short current state review, not months of process mapping. The review should show where manual effort, delays and recurring control problems sit across the main finance processes. Useful measures include process hours, close duration, forecast accuracy, overdue debtors, late supplier payments, recurring errors and rework. It should also record the spreadsheets and handoffs between systems, existing software and licences, material data gaps, current AI use and the owner of each process. Teams often find more unapproved AI than leadership expected, and a process with no end to end owner is usually the first thing to fix.

The governance floor should also be put in place during this initial stage. The first task is to identify current AI use and approve the tools that may handle finance data. The initial policy should also define what confidential information approved tools may receive, what must stay outside them and which approved business or enterprise environment may be used. At the outset, read only access should normally be the default at the general ledger level. Anything that can post should keep a person in the approval route, and every automated task should have a named owner. The governance floor is not a brake. It gives the team safe and clear boundaries within which useful work can start.

The same current state review should establish how benefits from AI and automation will be measured. A simple benefits ledger should record the baseline, expected benefit, running cost and actual result for every process in scope. The baseline should cover hours, error rate, cost and any working capital effect before the process changes. Later claims of value can then be tested against a fixed starting point rather than a shifting memory. Without the

baseline, the second project often never gets funded. By the end of this stage, the function should have a short list of processes and pain points worth testing, together with baseline measures, named process owners and an approved AI tool.

---

### 03 — RECOVER CAPACITY AND BUILD FLUENCY

Capacity is often the next practical constraint once the baseline and governance floor are in place. The fluency article sets out how to create it through [a controlled six week practice phase](#). The purpose at this point is not to automate the function. It is to let the team use an approved AI tool on familiar, low consequence, admin heavy work to learn what reliable output looks like and recover time for the process and data work that follows.

An approved AI tool such as Claude, ChatGPT, Gemini or Copilot should normally sit on a business or enterprise plan. Suitable first tasks are usually simple to hand over, stable month to month, worth keeping and low consequence, with read only work preferred where practical. Routine FP&A work such as variance commentary or a first pass over reporting often fits this profile because the underlying records are not changed. [The fluency article](#) gives a wider set of suitable tasks and explains how to choose them. During the six week practice phase, hours, errors and spend should be measured against the baseline. Time saved counts only after review and correction time are included. Proven tasks can then be saved as Skills within the AI tool, reusable instruction sets that turn one person's technique into a capability the team can share.

Two disciplines are particularly useful during the practice phase. Recovered hours should be reassigned deliberately to data improvement, process redesign or other agreed work rather than disappearing into the day job. Error rates should also be measured through reviewer corrections and other observable evidence, not impressions.

The practice phase itself does not need a separate pass or fail gate. It ends after the fixed six week learning period. The useful question at that point is whether the team has enough fluency and recovered capacity to support the data and process work that follows. If that capacity has not yet been created, the team can continue supervised use on tasks that still need work, redesign poor candidates and stop those that do not justify further effort. The data work can still progress in parallel. By the end of the practice phase, the approved tool should be in regular use, with several tasks running reliably. Hours, errors and spend should be measured against the baseline, successful methods should be repeatable by more than one person, and genuine time savings should have been reassigned.

---

### 04 — BUILD THE DATA FOUNDATION USE CASE BY USE CASE

The data workstream should develop in parallel rather than wait for the AI fluency work to finish. For each use case, the data foundation is the set of records, definitions and documents the process relies on. A collections use case, for example, may need customer master data, open invoices, cash receipts and the rules used to match and chase them. Those inputs need an agreed source, consistent definitions, controlled links between systems, reliable refreshes and current supporting documents. The aim is not to clean every dataset in the company before starting.

Two failure patterns are common. Some teams wait for perfect data and never start. Others deploy onto unchecked data and lose confidence as soon as a number proves unreliable. The practical starting point is narrower. For the use case being developed, identify the data it needs, agree which systems are authoritative, fix material gaps and reconcile the financial totals to the general ledger or another control total. Data outside that use case can wait until another process needs it.

That distinction explains why imperfect data is not the same as unreliable data. An LLM can read an untidy document and cope with inconsistent formats. It cannot make a broken customer master or a mismapped ledger reliable. The practical bar is therefore imperfect but workable data. Problems that make the current use case unreliable should be fixed first. The data work then covers two related areas: structured data held in systems, and document context such as contracts, reports, policies and process instructions.

The structured data work usually starts by mapping the sources. For each priority use case, the team should record where each input originates and which system is authoritative for it. A collections process, for example, may use customer details from the CRM, invoices and balances from the ERP, and cash receipts from bank feeds. Agreeing the authoritative source matters when systems disagree. The ERP remains the accounting system of record. Other tools, workflows and data platforms should ultimately reconcile back to it rather than create a parallel ledger.

The records then need to join consistently across systems. Common dimensions, the fields used to group and join records, usually include entity, account, customer, supplier, product, project and cost centre. One identifier across every system is rarely realistic. A governed mapping handles that by keeping a controlled cross reference between the codes used in different systems. If the CRM calls a customer C102 and the ERP calls it 4587, the mapping records that both codes refer to the same customer. The workflow or reporting layer reads that mapping on each run. An AI tool can use the same reference, but the relationship should sit in the controlled data layer rather than in model memory.

The same discipline applies to management measures. Revenue, bookings, margin and headcount should use agreed definitions across finance, sales and operations. Any permitted variant, such as headcount against full time equivalent, should be named explicitly rather than introduced inside a process. Consistent definitions reduce the risk of an AI assisted process creating another version of an existing number.

Data quality work should focus on defects that could make the selected use case wrong or incomplete. These may include duplicates, missing critical fields, inconsistent classifications or broken mappings. At least initially, the aim is to fix the issues that matter to the current use case rather than start a general clean up. The relevant data should reconcile to the authoritative source, and recurring exceptions should have a named owner.

AI tools can assist with this work. An LLM can compare field names, descriptions and nearby values across ERP, CRM and HR data and flag likely matches. It might identify two differently named customer fields as candidates for the same definition. The relevant business or data owner should confirm the relationship. Once approved, the mapping or definition should be stored in the controlled reference used by the process. The model should not decide or remember the relationship on its own.

Recurring manual extracts can usually move towards reliable automated feeds once the underlying data is usable. In practical terms, a scheduled or system connection refreshes the data rather than somebody exporting and reloading the same file each time.

At mid market scale, the ERP and its reporting layer can themselves provide a governed data foundation. A separate warehouse or data platform is not automatically required. It usually becomes useful when reporting or AI workloads start to slow the ERP, when finance needs to combine substantial non finance data, when historical snapshots must be preserved, or when the ledger does not contain enough detail.

Incoming information should also be captured digitally as early as practical. Where supplier invoices arrive as PDFs, document extraction can capture supplier details, amounts, dates and references into structured fields, reducing later rekeying. Any posting still follows the agreed finance controls.

Ownership should be explicit at three levels. Each critical dataset should have a named business owner responsible for its meaning and quality, together with a refresh cadence, access rules, a quality check and an escalation point. Finance should own the financial definitions, control totals and reconciliation. IT should own the technical environment. Where a central data function exists, it should maintain common enterprise standards and coordinate the shared data model across functions so finance uses the same customer, supplier and organisational definitions as the rest of the business.

Structured records are only one part of the foundation. AI assistants also need the documents that explain how finance work should be done. Contracts, reports, policies, process instructions and review checklists should sit in controlled folders with clear names, current versions and owners. An assistant reviewing a contract, for example, needs the current agreement, the relevant policy and the checklist that defines what should be checked. Without those supporting documents, the assistant may read the contract correctly but still apply the wrong rules.

The same data preparation supports later work in two different ways. A vendor trial needs recent real data that has been reconciled and is representative of the process, so product fit can be tested on the organisation's actual records. [The Buy build article covers how that evidence is used in product selection and trials](#). FP&A uses the data foundation directly. Forecasting and commentary become more useful as financial data is connected with agreed operational drivers such as volumes, headcount and pipeline.

Together, this work provides a practical test of whether the data is ready for a particular use case:

- the source data systems are known
- the authoritative data source is agreed
- critical data fields and dimensions are consistently defined and traceable
- the data reconciles to the relevant control total, such as the general ledger balance, bank balance or another agreed total that confirms the records are complete
- the data refresh and review cadence suits the use case
- material exceptions are visible and owned
- data access has been approved

---

## 05 — PRIORITISE THE PROCESSES AND CHOOSE THE ROUTE

The practice phase is mainly about learning to use an AI assistant and recovering capacity. It will often reveal recurring finance processes where a more permanent automated or AI enabled process could remove more work. The next decision is which of those processes is worth redesigning and automating for live use. That usually means connecting the process to source systems, defining the way of working, adding the necessary controls and assigning ongoing ownership. Close, collections, payables and forecasting are common candidates. This is a different decision from choosing [the low consequence practice tasks covered in the fluency article](#).

The current state review creates a list of processes and pain points worth improving. At this stage, that list should be ranked by what each candidate could deliver and what it would take to implement safely. Expected value, repeatability and volume indicate the economic opportunity. Rules versus judgement and data readiness indicate how feasible the process is to automate or support with AI. Financial or control consequence, time to value and reversibility indicate the risk and commitment involved. A change that can be unwound quickly is a smaller commitment than one embedded deeply in a core process. A simple technical feasibility check should also confirm

that the necessary systems can connect safely and the required access is available. The Buy build article covers [the detailed assessment of system connections, product fit and implementation options](#).

Once the priorities are clear, the next question is what technology each process actually needs. [The context article sets out that decision logic](#). The default is to automate where the rules are clear and the route is known. Machine learning is brought in where the task is prediction or classification, and an LLM where the work needs interpretation and judgement. An agent becomes relevant where the next action depends on what the previous action found, so the route cannot be fixed reliably in advance.

An invoice under five thousand pounds from an approved supplier matched to a purchase order is a rule. An invoice from a new supplier with no purchase order is a decision. The first belongs in automation. The second may justify an LLM. Using an LLM for routine rule work will usually add cost without improving the outcome.

Once the technology need is clear, finance should decide where that capability should come from. [The Buy build article develops that decision in detail](#). The practical order is to check what is already available before adding anything new. Finance should review the ERP, Microsoft or Google estate, other existing licences and any outsourced provider. If those do not provide enough capability, an established specialist product is usually the next option to assess. Assembly becomes relevant where no product fits the organisation specific requirement or the economics do not work. Build is the fourth option and should remain unusual because the organisation then owns the software and its maintenance for the long term.

The Buy build article sets out how to compare those choices, including sourcing economics, product trials and the decision between workflow led and agent led assembly. Cash management is often one of the first specialist purchases worth assessing. It can work from bank feeds and the general ledger with relatively modest data preparation, and at mid market scale the pricing can often be proportionate to the value delivered.

Finance should usually take only a small number of the best supported processes into the prove stage at any one time, rather than progress every candidate at once. By then, each process should have an agreed way of working, sourcing route, named owner and expected return. Those decisions give the pilot something concrete to test and keep effort focused on the cases most likely to justify live use.

---

## 06 — PROVE IT, THEN TAKE IT LIVE

A use case chosen for live use should normally be tested in a bounded pilot first. The current performance should be baselined and the target stated before testing begins. A first pilot can use one supplier group, account set or business unit rather than the full population. Copied or prior period data is usually preferable where that keeps the live ledger outside the test.

The sample should include routine cases and awkward exceptions, because the exceptions are often where the effort sits. The pilot should follow the process as people actually run it, including the small corrections that may be missing from the written procedure. Error types and pass thresholds should be agreed before the results are known. A person should review the outputs, and total process time, errors, cost and value should be measured rather than drafting time alone. Inputs, outputs, reviewer corrections and results should be retained together as the pilot's evidence record.

The context article [distinguishes a successful experiment from a production process](#). The go live gate applies that distinction to each candidate. Every requirement below should be met before the process is relied on live.

- Ownership: a named business owner and a named maintenance owner.
- Data and integration: defined source data, reconciliation, stable connections to the systems it touches, and approved permissions.
- Testing: thresholds agreed before testing, evidence that they were met, and coverage of the awkward exceptions.
- Control: agreed review and decision rights, a current operating procedure, trained reviewers, an evidence trail, and an exception and failure route.
- Economics: the measured case still stands once the full running cost is in.

One line of the gate matters most. A pilot that misses its agreed accuracy, control or economic threshold does not go live. It is redesigned, given different resources or stopped. Stopping at the gate costs little. Stopping after go live costs far more, and usually part of the team's credibility.

The gate should continue to apply after go live. A material change to the model, its instructions, source data or a system connection should send the process back through the relevant tests before reliance resumes.

Processes with more autonomy will usually need tighter controls. A standing agent starts work on a schedule or trigger rather than at a person's request. The governance article sets out the additional requirements, including a separate identity, an agent control record and a harness, [the technical layer coded round the LLM](#).

---

## 07 — SCALE BY REPETITION

The final stage is not a grand rollout. It is repetition. Each new use case can reuse parts of what has already been proved, including data feeds, governed mappings, permissions, workflow components, testing methods, control patterns, Skills and vendor connections. That reuse should allow later cycles to move faster without lowering the bar.

Each cycle should also remove the work the new process replaces. Spreadsheets, duplicate reports and manual workarounds should be retired rather than left running alongside the new process.

The benefits ledger should continue after go live. Actual results should replace pilot assumptions, with full running cost, including model and platform spend, compared with the realised benefit. A process that stops earning its place should be redesigned or stopped rather than protected by its original business case.

The live review should also keep the approved thresholds in force. Error rates, exceptions and control failures should be tracked alongside value, and a process that moves outside an approved threshold should return to review. Usage based AI charges can rise with activity rather than headcount, so each agent or tool should have a named owner, a spending cap and a place in the benefits ledger.

The team will also tend to change as three capabilities become more prominent. Process owners and technical accounting specialists should continue to own finance outcomes and supervise automated work. Data and automation specialists should configure and maintain data, workflows and agents. Across both groups, finance professionals should apply accounting, control and business judgement to the design and review of that work. These capabilities still need conventional management. Senior finance leaders should set priorities, resolve tradeoffs, decide where human review remains necessary and make sure the different roles operate as one finance function.

Human accountability remains the constant. Whatever runs automatically, a named person remains accountable for the number. The federated approach described in the strategy article, with finance owning its domain within shared IT and data standards, can then become part of normal operations rather than a temporary programme structure.

Some organisations may eventually revisit the core finance platform as the foundation matures. The options can include an AI native finance ERP or a continuous close platform, which spreads close work through the period. The starting question should remain what the existing ERP can do before a replacement is considered. [The buy build article owns that decision.](#)

The destination described in the strategy article, largely automated transactions, a close in a day or two, current profit and cash, rolling forecasts and one trusted set of numbers, is unlikely to come from one systems programme. It is more often the cumulative result of repeated improvements, with the data foundation becoming stronger each time and the same evidence based gates applied throughout.

---

#### THE AI IN FINANCE SERIES

- 01 AI in finance: what it is and how it works [aicfopartner.ai/ai-in-finance](https://aicfopartner.ai/ai-in-finance)
- 02 AI strategy for finance: the six choices [aicfopartner.ai/ai-strategy-for-finance](https://aicfopartner.ai/ai-strategy-for-finance)
- 03 The roadmap to the AI enabled finance function [aicfopartner.ai/ai-finance-roadmap](https://aicfopartner.ai/ai-finance-roadmap)
- 04 Developing an AI enabled finance team [aicfopartner.ai/ai-finance-team-development](https://aicfopartner.ai/ai-finance-team-development)
- 05 Sourcing AI capability: buy, build or assemble [aicfopartner.ai/ai-buy-build-or-assemble](https://aicfopartner.ai/ai-buy-build-or-assemble)
- 06 Governance that speeds AI adoption [aicfopartner.ai/ai-governance-for-finance](https://aicfopartner.ai/ai-governance-for-finance)

Questions or comments: [tim@aicfopartner.ai](mailto:tim@aicfopartner.ai)