

Organization design · Finance systems

Decide who owns finance's technology work

Decide who owns one finance capability, what each person may change and how the work will get done. Use the example to agree responsibilities across finance, IT and the teams providing the data.

What you finish with: An ownership decision covering responsibilities, authority, available time, handoffs and the conditions that would make you revisit it.

[Read this guide online](#)

OWNERSHIP / FIVE LAYERS

Assign the work before the reporting line.

Result	The number and its accuracy	ONE OWNER
Configuration	Rules, models and definitions	ONE OWNER
Data feeds	Complete, checked extracts and loads	ONE OWNER
Platform	Contract, access, security and run cost	ONE OWNER
Local use	Business-unit requirements and use	ONE OWNER

For every layer: authority to act, time to do the work and an explicit handoff.

01 Start from one capability and the result it serves, not the organization chart

The question "should finance own its technology?" has no answer at the level of the organization chart, because a planning application, a reconciliation rule, a reporting layer and an assistant need different work from different people. Ask it about one capability at a time: the planning model, the close checklist in the ERP, the invoice matching rule. Write the result that capability serves in the words the finance leader uses, with its measure, and name the person who owns that result. That person is a business owner in finance; they are not, by that fact, the owner of the software.

Then list what has gone wrong in the last year, with dates: the change that waited on a supplier ticket, the feed nobody noticed had failed, the local spreadsheet that grew because the model did not reflect a unit's drivers. Those are the facts the decision has to fix.

02 Separate the layers of the work, and give each one owner

Separate the work into five layers for this decision. One person may cover more than one layer; make each responsibility clear. The result: the number the business acts on and its accuracy. The configuration: the model logic, the rules, the driver definitions, the report layouts, the things that change when the business changes. The data feeds: the extracts and loads that carry actuals, pipeline or master data into the capability, and the check that they arrived complete. The platform: the contract, the security, the access provisioning, the upgrades and the run cost.

Local use: what each business unit does with the capability and the requirements it can legitimately ask for. Each layer has one owner, named as a role that exists today, and the owner of a layer holds two things: the authority to change it and the capacity to do the work. Confirm both the time available and the authority to act before assigning the work. Where a layer sits with enterprise IT, a shared service or a supplier, the owner is the accountable person in the organization, not the supplier.

03 Test each owner for authority, capacity and the handoff

For each layer, ask three questions and write the answers down. Authority: what can this owner change on their own authority, and where does that authority stop? A planning analyst who may change driver logic but not the integration or the security model has a limit worth writing in one line. Where an automation or an agent acts inside a layer, name the account it acts under and what it may change without a person; that limit belongs to the layer's owner and is written beside theirs. Capacity: how much of this person's time does the layer need, measured from last year's changes and incidents, and is that time available or already committed?

Handoff: when this owner needs the next layer, who do they ask, how long does it take, and what happens if the two disagree? Name the roles that resolve a disagreement and the deadline. Use an existing meeting when its date allows the decision to be made in time. Local teams get the same

treatment: a unit's finance manager can propose a driver that reflects the unit's business and can ask for a report; record whether the business needs one shared model or justified local variations.

Where the platform is run by enterprise IT under a shared-service arrangement or an outsourced contract, the test does not change: the accountable person is inside the organization, and confirm the internal time needed to oversee the work and the service capacity available under the agreement. Record response and completion commitments separately; a response time is not a completion promise.

04 Ask for one decision and set the trigger that reopens it

State the decision in one sentence: this capability, these owners by layer, these limits, this escalation route, and the work that stops because the ownership is now clear. Say what each owner needs to take the layer on: access, training, time released from something else, a contract change. Say what does not change: the platform's contract, security and run cost stay where they are unless the decision moves them, and moving them is a separate decision with its own cost. Then set the trigger: a measured condition that shows the ownership is not working, who watches it, and at which existing meeting it is reported.

Name the responsibilities before choosing a reporting structure. Consider a new role or team when the work, authority or capacity cannot be covered effectively by existing owners.

Example decision

The driver-based planning model

A manufacturing group has one planning application and three business units. Group FP&A owns the forecast; enterprise IT holds the platform contract and administrator rights. Model changes pass through IT to the supplier. Two units also keep spreadsheet forecasts because the model does not reflect their business drivers.

The result:

The FP&A lead owns the monthly group forecast, its agreed accuracy measures and acceptance of each unit's forecast. Last year's record shows 22 model-change requests, averaging 16 business days elapsed. Two units rebuilt forecasts in spreadsheets in nine of twelve months. The CRM feed failed in March and June without anyone raising the failure before the review.

Model configuration:

The planning analyst will maintain approved drivers, model logic and report layouts. The FP&A lead approves the requested change and checks its effect before the next forecast. The analyst records each change and its reason. Enterprise IT provisions narrowly scoped permissions and audits their use. Integrations, security and user access remain outside the analyst's authority.

Available capacity:

The initial estimate is 66 days a year: 22 changes at three days of effort each. That is 30 percent of a 220-day working year. The proposed allocation is 40 percent, or 88 days a year, to allow for additional requests. These are effort estimates; the ticket log records elapsed time. Actual effort and demand will be measured from the first month.

Where the time comes from:

Moving three units' variance narratives releases an estimated six analyst days a month. Stopping consolidation of two spreadsheet forecasts releases a further three, after the FP&A lead accepts those units' forecasts from the application. Full release is nine days a month against an average requirement of 7.3. Until consolidation stops, only six are released.

If that lasts a full quarter, the estimated gap is four analyst days, before any training or transition effort. The FP&A lead must reduce or defer approved work, or secure named cover, before accepting delivery dates that require the missing capacity. A queue records that gap; it does not remove it.

Receiving teams:

Each unit's finance manager must confirm time for the transferred narrative work. Existing spreadsheet work may be replaced for two units once their forecasts are accepted; no offset has been established for the third. The FP&A lead also estimates one additional day a month for checks and acceptance, and sales operations estimates half a day for the CRM checks.

Each owner confirms the time available and what will move before the transfer begins.

Data feeds:

The enterprise IT integration lead owns the ERP feed, its daily completeness check and escalation of a failed load. In this example, the existing service agreement provides a next-business-day incident resolution target, confirmed with that lead. The sales operations lead owns the CRM extract, its load and a same-day completeness check. The FP&A lead reconciles it against CRM as a second check.

Feed owners raise failures when detected; they do not wait for the performance review.

The platform and local teams:

The head of enterprise applications retains the contract, security, access, upgrades and license cost. Before assigning the analyst's permissions, confirm that the platform can restrict them to the agreed scope and whether the role costs extra. If it cannot, enterprise IT retains the administrator role; its lead must confirm capacity and delivery dates for the analyst's approved requests.

Each unit's finance manager proposes drivers and reports through the analyst. This example retains one shared model.

Delivery dates and disagreements:

The analyst confirms each completion date after scope, approval, inputs and available capacity are known. At a steady 40 percent allocation, a three-day change requires 7.5 business days of capacity before waiting time or interruptions; that calculation does not establish a ten-day service promise. For example, two such changes need six effort days, while ten business days provide four at that allocation.

The FP&A lead prioritizes competing requests. For IT requests, distinguish the contract's ten-day response from an agreed completion date. Scope disagreements go first to the FP&A lead and head of enterprise applications, then to the finance leader and head of enterprise applications if unresolved within ten business days.

Use the existing technology review when it falls before the affected decision deadline; otherwise the same approvers resolve it earlier.

What must be in place:

Confirm scoped access, the supplier's 3,500-dollar training price, training time, any additional license cost and each owner's capacity. The training payment uses the existing budget. If access or capacity cannot be confirmed, retain the current operating arrangement until an agreed alternative is ready. Where automation is involved, record its account, permitted actions and human approval limits with the responsible owner.

What stops:

Supplier tickets stop for model changes within the analyst's confirmed scope. A unit's spreadsheet forecast stops only after the FP&A lead accepts its application forecast through one monthly cycle, recording the unit, month and drivers. Spreadsheet consolidation stops with the accepted replacements. Variance narratives transfer only when the receiving managers' capacity is confirmed.

Decision and limits:

At the October technology review, the finance leader and head of enterprise applications approve the responsibilities and conditions above. The FP&A lead confirms those conditions before the transfer starts. The platform budget stays with enterprise IT. A new team or role requires evidence of work that existing owners cannot cover; this decision does not establish that need.

When to revisit:

The FP&A lead brings missed agreed completion dates, effort above available capacity, continued spreadsheets after acceptance, and feed failures first detected by a secondary check to the monthly technology review. An issue that threatens the next forecast is escalated when found. The approvers adjust capacity, authority or ownership in response to the evidence.

Your working record

TECHNOLOGY OWNERSHIP DECISION

Field	Your record
Capability and the result it serves (measure, result owner):	
What went wrong last year (dated, from the record):	
Configuration owner (authority, available time, change approval and checks):	
Data feed owners (each feed, the check, who raises a failure):	
Platform owner (contract, security, access, upgrades, run cost):	
Local use (what units may propose, who approves, one model or several):	
Handoffs (response, completion date, dependencies and available capacity):	
Escalation (roles, deadline and meeting if timely):	
What each owner needs (access, automation limits, training, time, contract change):	
Work stopped or transferred (acceptance and receiving-team capacity):	
Not decided here (budget moves, new roles):	
Decision requested (who approves together):	
Trigger (measured conditions, who reports, at which meeting):	

[Open the guide to copy this record into your team's document](#)

Related reading

[Turn a system change into work your team can do](#)

When the owners and their capacity are agreed, give the receiving team the change as a brief: one practice task, the time made available, and the first checkpoint.