

Before You Write the RFP: Five Decisions to Determine Platform Fit

See also the companion workbook: Kahua-Class Capital Program RFP Requirements

Most owners begin writing the RFP after key structural decisions are already taking shape. Those decisions often develop informally as teams borrow ideas from vendor demos, peer templates, and language from the last procurement. By the time the RFP is drafted, assumptions about program structure, data ownership, and how vendors will prove their claims may already be built into the requirements. This guide will show how to reverse this order of operations.

It walks through five decisions to make before you write requirements, so the RFP you release selects for fit instead of marketing claims. For each decision, the guide outlines what to define internally, questions for your team and example language to use in the RFP.

A companion requirements workbook turns the five decisions into a comprehensive set of requirements ready to use as an RFP attachment, evaluation scorecard or both.

You will learn how to:

- Make five decisions that determine whether your future platform fits your organization, or forces your organization to fit the software.
- Align internal stakeholders before requirements harden.
- Write outcome-based requirements that resist a checkbox answer.
- Build demo scenarios that test fit instead of showmanship.
- Choose a procurement path early enough that it widens the field instead of narrowing it.

The result is an RFP that selects for organizational fit and supports consistent vendor evaluation.

How to use this guide

The body is built around the five decisions below. Each follows the same structure: Why It Matters, What to Define, Questions to Ask Internally, and Example RFP Language. Work through them with the people who own each area: program management, IT and security, finance, and operations. When you finish, the Pre-RFP Checklist near the end confirms you are ready to write.

This guide cites Kahua in several places as practical evidence of what a decision looks like in practice for owners running capital programs. The decisions themselves are written for the owner and apply to any platform you evaluate.

Decision 1: Define your portfolio operating model

A capital program has multiple levels: portfolios contain programs, and programs contain projects with their own contracts, budgets, and documents. The first decision is who controls that structure and who can change it. If every project team invents its own cost codes, phase names, and approval steps, nothing rolls up. If the structure is locked centrally with no room for local variation, teams route around it and keep their real numbers in spreadsheets.

What to define

- ✓ The portfolio hierarchy, and the level at which standards are enforced versus delegated.
- ✓ Who owns configuration of templates, document types, and approval workflows, and through what change process.
- ✓ Whether your own administrators can make those changes, or whether each one requires a vendor services engagement.
- ✓ How a new project inherits standards on day one instead of being rebuilt from scratch.

Questions to ask internally

- Who owns our standards, and who is allowed to change a workflow?
- Can our administrators configure the system without buying vendor professional services each time?
- When we start a new project, does it inherit our templates, budgets, and permissions automatically?

Most products claim to be configurable. Specify what that means in practice: your administrators can set and update standards through a defined governance process and manage and report across project, program, and portfolio levels.

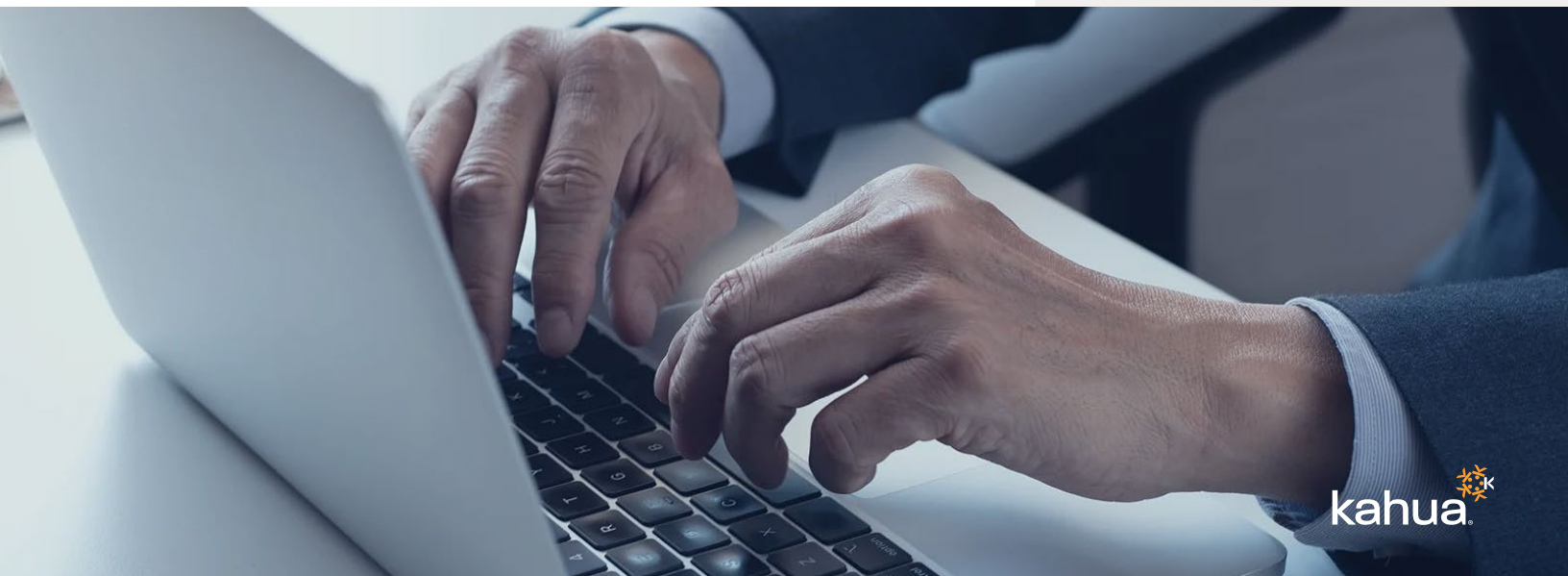
From the field

“We can easily build custom applications that conform to our business processes, rather than tailoring our processes to the system.”

Jeff Wilson,
Senior Engineer Project
Manager Pennsylvania
Turnpike Commission

Example RFP language

Owner administrators shall be able to create and modify process templates, document types, custom fields, and approval workflows without vendor professional services. The system shall support a configurable hierarchy spanning enterprise, program, and project levels, with data roll-up at any node and role-based access cascading through the hierarchy.



Decision 2: Define your platform strategy

Most capital program software is sold as a finished product: a fixed set of workflows and data structures you are expected to conform to. That approach works only until the program changes. A new funding rule or approval step can expose the system's limits, leaving teams to work around the system with spreadsheets, side tools, or paid customizations that complicate future upgrades.

A well-designed platform keeps security, permissions, audit history, and integrations stable while allowing the owner to configure forms, workflows, and applications. The foundation governs security, permissions, audit history, integrations, and data quality.

Kahua uses this architecture: the platform layer carries governance and security, while kBuilder lets owner administrators build and adapt applications without waiting for a vendor release or support ticket.

What to define

- ✓ Which of your processes change often enough that you need to change the system yourself.
- ✓ Who should be able to add a field, change a workflow, or stand up a new application.
- ✓ Whether configuration changes survive the next release.
- ✓ How adaptability stays governed, so it does not turn into sprawl across the portfolio.

Questions to ask internally

- How often do our processes change, and who should manage those changes?
- What happens to our configuration when the vendor ships an upgrade?
- Can we build a new application ourselves, or do we file a ticket and wait?

From the field

"Software being able to conform to your business is essential. It is about how easy we can make it for our users to adopt it."

**Monica Rizzo, Project
Administration
Administrator, BPGS**

Example RFP language

The platform shall allow owner administrators to modify workflows, forms, and business processes, and to build and deploy new applications natively, without software redevelopment or vendor source-code access. Configurations shall persist across vendor upgrades and remain governed by enterprise security and approval policy.

The image shows a laptop screen with a financial management application. The application has a top navigation bar with tabs for 'Capital Plan', 'Amount', 'Remaining', and 'Released'. Below the navigation bar, there's a search bar and a 'By Function' dropdown. The main content area displays a table of projects. The table has columns for 'Number', 'Name', 'Original Budget', 'Start Date', 'End Date', 'Amount', 'Variance', '2025', 'January 2026', and 'February 2026'. The table lists several projects, including 'AZ DOA', 'Danie & Fish', 'Corrections', 'Public Safety', 'Parks & Trails', 'ASPT 100% Building Renewal Formula Funding', 'ASPT Statewide Campground Improvements', 'ASPT Statewide Shade Structure Construction', and 'ASPT Statewide Water Conservation'. The bottom of the screen shows a status bar with the 'ELG' logo, 'Plan Start' and 'Plan End' dates, and a user login information.

Capital Plan	Amount	Remaining	Released
FIN2 Capital Plan	\$90,500,000	\$10,208,000	\$77,794,000
			\$3,022,000

Number	Name	Original Budget	Start Date	End Date	Amount	Variance	2025	January 2026	February 2026
0000	AZ DOA		1/1/2025	12/31/2028	\$77,794,000	\$0	\$27,860,598.98	\$3,436,542.50	\$3,070,402.50
0001	Danie & Fish	\$14,195,000	01/01/2025	12/31/2027	\$14,195,000	\$0	\$1,832,648.58	\$109,750	\$709,350
0002	Corrections	\$14,370,000	01/01/2025	12/31/2028	\$14,370,000	\$0	\$8,765,700	\$1,077,750	\$1,077,750
0003	Public Safety	\$38,482,000	01/01/2025	12/31/2028	\$38,482,000	\$0	\$9,795,070	\$1,202,062.50	\$1,202,062.50
0004	Parks & Trails		1/1/2025	12/31/2028	\$10,694,000	\$0	\$10,367,780	\$446,480	\$80,200
0001	ASPT 100% Building Renewal Formula Funding	\$3,692,000	03/10/2025	08/12/2025	\$3,692,000	\$0	\$3,692,000	\$0	\$0
0002	ASPT Statewide Campground Improvements	\$4,300,000	05/05/2025	10/12/2025	\$4,300,000	\$0	\$4,300,000	\$0	\$0
0003	ASPT Statewide Shade Structure Construction	\$2,233,000	08/13/2025	01/24/2026	\$2,233,000	\$0	\$2,233,000	\$0	\$0
0004	ASPT Statewide Water Conservation	\$1,258,000	06/18/2025	03/25/2026	\$1,258,000	\$0	\$1,258,000	\$80,340	\$0

Save Get Current

Plan Start 1/1/2025

Plan End 12/31/2028

ELG

DM001 | 1:35:45 | Last Login Friday, July 11, 2025

Decision 3: Define security and compliance requirements

Security requirements usually arrive as a list of certifications pasted into a late section of the RFP. That produces a yes-or-no answer and tells you almost nothing about how the system behaves. Capital program data carries real exposure: budgets, contracts, drawings, personnel records, and sometimes controlled or regulated information.

The RFP should also test how the architecture controls access and produces audit evidence, along with how it meets hosting requirements.

What to define

- ✓ The data classifications the system must hold, and the compliance regime that applies to each. Confirm whether a specific authorization level is genuinely required for the data in scope; over-scoping this is a common and costly error.
- ✓ How access is controlled at the portfolio, project, and record level, including externally federated access for partners and contractors.
- ✓ What the audit trail logs, how immutable it is, and how an auditor retrieves it.
- ✓ Hosting and data-residency requirements, stated up front rather than discovered in negotiation.

Questions to ask internally

- What data will actually live in the system, and what security requirements govern it?
- Who needs access, including outside organizations, and at what level?
- What audit evidence will we need to produce, and to whom?

Kahua serves government programs through the Kahua Government Network, a dedicated environment that is FedRAMP Moderate Authorized, and commercial programs through the Kahua Commercial Network, which carries SOC 2 Type II reporting and ISO/IEC 27001 certification.

Use these as architectural constraints: Ask each vendor to demonstrate how the system meets them and provide the relevant certifications.

Example RFP language

The system shall enforce role-based access at the portfolio, project, and record level and produce a tamper-evident, non-deletable audit log retrievable by authorized auditors. The hosting environment serving in-scope data shall hold FedRAMP Moderate authorization where required by the data classification, maintain SOC 2 Type II reporting, encrypt data at rest and in transit, and keep customer data within the continental United States.



Decision 4: Define your integration strategy

No capital program software lives in isolation. It exchanges data with an ERP or financial system, document management, scheduling, reporting, and often GIS or asset management. Integration failures can leave the same data in two systems without a good way to reconcile it.

What to define

- ✓ The named systems that must connect, and which system is authoritative for each type of information.
- ✓ Whether each integration uses a documented, vendor-supported interface or requires custom development, and who is responsible for maintaining it when either system is updated.
- ✓ The data latency the program can tolerate: real-time, scheduled, or manual.
- ✓ The ownership and ongoing cost of each connector after go-live.

Questions to ask internally

- Which system owns the financial record, and which systems must exchange data with it?
- After go-live, who maintains each integration, our team or the vendor?
- Can we build and change integrations ourselves, or are we dependent on the vendor for each one?

An integration only the vendor can explain or change creates an ongoing dependency.

From the field

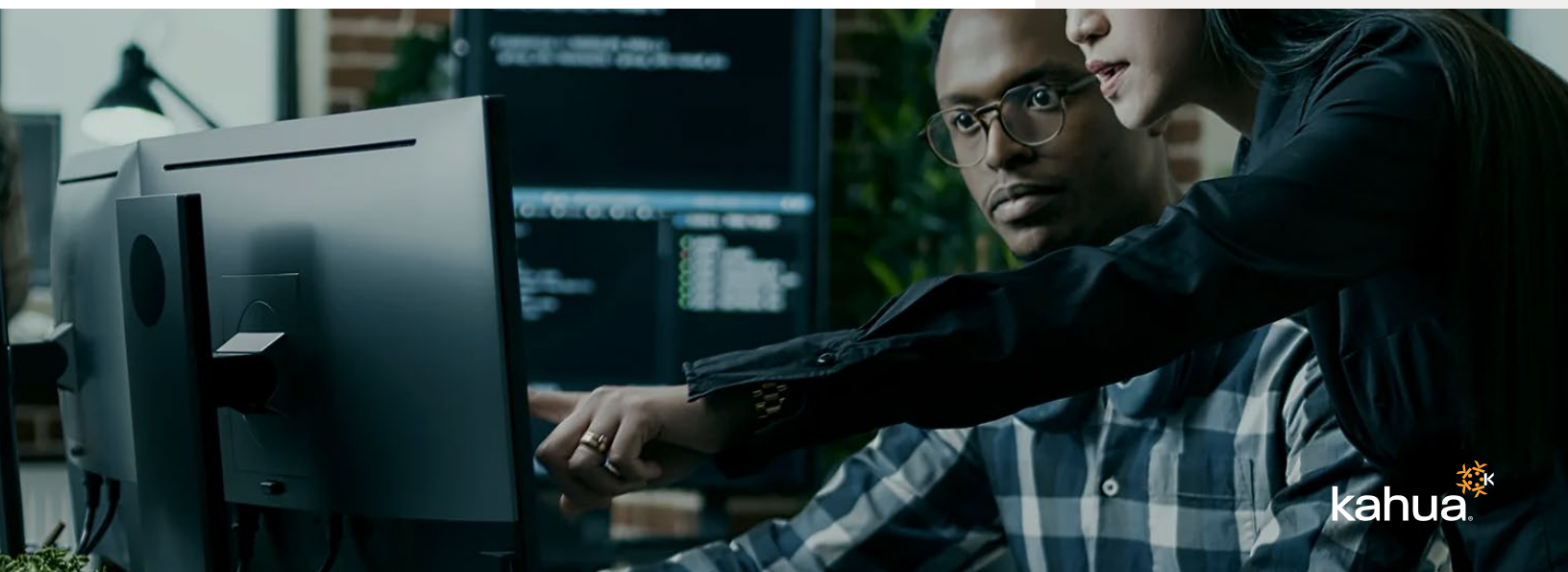
“Even out of the box, the platform gave us more ability to configure for our needs, and the analytics were a big part of the decision.”

Jeff Morrow,
VP, Chris-Tel Construction

Example RFP language

The system shall provide a documented, supported REST/open API enabling bidirectional integration that is maintained across upgrades to either system, and shall provide bidirectional integration with the owner's ERP or financial system.

The owner shall be able to define the direction of data flow and the authoritative source for each shared object, and to build and modify integrations without depending on the vendor for each change.



Decision 5: Define reporting and success metrics

Reporting can fail in two ways. First, the system may capture data well but not roll it up across the portfolio in the format executives need. Second, the evaluation may focus on features without examining adoption. A vendor can prove a feature exists in a demo; adoption evidence helps show whether teams will use the system after the consultants leave.

What to define

- ✓ The portfolio-level views the program must produce on demand, including cost and schedule rollups, exception reports, executive summaries and any required public or oversight reports.
- ✓ That reporting draws from the live system of record, not a separately maintained extract.
- ✓ Proof of adoption: active-user counts, modules in regular use and multi-year retention, supported by references from owners you can contact directly.

Questions to ask internally

- What reports do our executives and oversight bodies actually need, and how often?
- How will we distinguish between widespread adoption and completed installation?
- What evidence from comparable owners would convince us the platform is used day to day?

Where reporting must serve the public (taxpayers, boards, and oversight committees), the same principle holds: the dashboard should read from the system of record, not a separate environment maintained by hand.

From the field

“Before, this took half a day of calling people and wrangling numbers. Now the report is automatic. About 30 minutes, and it is all in one spot.”

Aaron Randall,
Financial Analyst, Walters

Example RFP language

The system shall report across multiple projects and the full portfolio, aggregating cost, schedule, and status with drill-down to line-item detail. The system shall generate exception reports and executive summaries using current data from the system of record, without relying on a separate, manually maintained data extract. The vendor shall provide adoption evidence from owners of comparable scale, including active-user counts, modules in daily use, and references the owner may contact.



Turn these decisions into an RFP

Decisions matter only if they survive contact with the procurement document. The following steps turn them into testable requirements.

Step 1: Complete the five decisions.

Work through Decisions 1 - 5 with the owners of each area, and record the answers. The companion requirements workbook gives you a structured way to document this.

Step 2: Write requirements as outcomes you control, not features the vendor names.

Outcome questions are hard to answer with a checkbox. It's easier to verify them in a live demo.

Instead of	Write
The system shall be configurable.	Owner administrators shall modify workflows, forms, and business processes without vendor professional services.
The system shall be secure.	The system shall enforce role-based access at the portfolio, project, and record level and produce a tamper-evident audit log an auditor can retrieve.
The system shall have good reporting.	The system shall roll cost and schedule up across the portfolio from the live system of record, with drill-down to line-item detail.

Step 3: Script the demo to your needs and priorities, not the vendor's.

Hand each vendor a short set of scenarios drawn from your own work, and watch them perform the same tasks:

1. Stand up a new project under portfolio standards.
2. Change an approval workflow as an administrator, without vendor help.
3. Roll a cost overrun up to an executive view.
4. Grant a contractor-governed access to one project without exposing the rest of the portfolio.
5. Exchange project data with a named external system.

A demo that the owner scripts tests fit. A demo that the vendor scripts tests showmanship.

Pre-RFP preparation checklist

Before the requirements document goes out, the program office should be able to mark each step complete:

- Portfolio hierarchy and governance model defined; configuration authority assigned ☐
- Authoritative system named; data ownership and export terms confirmed ☐
- Data classifications matched to the compliance regimes that actually apply ☐
- Access and audit architecture specified at the behavior level, with certifications named where required ☐
- Systems that must integrate named, with an owner for each connector after go-live ☐
- Portfolio and oversight reports described as required outputs ☐
- Adoption evidence and reference owners required, not a feature demo ☐
- Procurement vehicle identified and its current availability confirmed ☐

An owner who can mark these complete is ready to write an RFP that selects for fit. Otherwise, vendors will define the frame.

Choose your procurement path early

The procurement route affects both the timeline and the field of bidders, so choose it before requirements are final. The steps below are described at a planning level; confirm exact contract identifiers and current availability against live sources before any external use.

- **Direct purchase from the publisher:** The simplest commercial path, suited to an owner with no mandate to use a specific vehicle.
- **GSA Multiple Award Schedule:** For federal owners, a pre-competed path that reduces lead time, provided the relevant schedule and item number are current.
- **Reseller schedules and government-wide vehicles (for example, NASA SEWP):** Available through authorized resellers for in-scope products. A publisher's schedule and a reseller's schedule are distinct instruments; one does not cover the other.
- **Cooperative purchasing agreements:** Frequently available to state, local and education buyers, with pricing and terms pre-competed.
- **Small-business or socioeconomic partners:** Where set-aside goals apply, a qualified partner can satisfy both the technical need and the socioeconomic requirement, as long as the partner is genuinely authorized to deliver and support the product.

Choose the route before requirements are final because it sets the competitive field and realistic award timeline.

Why Kahua aligns with these decisions

Owners select Kahua for owner-controlled configuration, multi-project portfolio management and financial governance. They also use it for integration flexibility, as well as security and compliance requirements that must stand up under audit. The platform adapts to an owner's existing processes, and the organizations cited throughout this guide run their daily program management in Kahua.

The companion requirements workbook makes those capabilities concrete as requirements you can test. Use this guide as the front half of a procurement: Translate the five decisions into requirements with the workbook, build a demo script from your portfolio and confirm the buying route before finalizing the RFP.

When the program office defines these decisions early, the RFP is more likely to reflect the program's actual needs.

[Learn more about Kahua](#)

Kahua® is an AI enterprise construction platform built to connect the people, processes, and data behind complex capital programs. Trusted globally across government, healthcare, education, commercial, and energy sectors, Kahua serves as a system of record for owners and delivery teams, bringing stakeholders into a single governed environment for portfolio-level visibility and collaboration across the full asset lifecycle. With intelligent, purpose-built solutions for key industries and a highly configurable platform, Kahua adapts to how teams work today and tomorrow.