

Build vs. Buy IT Management

15 questions to compare an in-house IT Management build with a purpose-built ITAM and ITSM platform

Use this checklist before approving an internal IT Management build, signing a software agreement, or presenting either option to leadership. It compares the evidence behind both paths, not the confidence of the people proposing them.

Outcome: a documented comparison of fit, operating responsibility, three-year cost, risk, and time to value.

How to use this checklist

Rate the internal build and the vendor option separately for every question. Use the same evidence standard for both.

Score	Evidence standard	What qualifies
0	Assumption	No named owner, budget, timeline, test, or verifiable proof.
1	Partially evidenced	The need is scoped, but ownership, cost, coverage, or testing remains incomplete.
2	Demonstrated	A working test or documented owner, budget, timeline, controls, and operating process exists.

Important: A vendor statement counts as evidence only when it can be demonstrated, documented, or contractually confirmed. An internal promise counts only when it has a named owner, capacity, and testable plan.

Complete the review in four passes

- Agree on the must-have workflows and the decision time horizon.
- Score both paths using evidence available today, not future intent.
- Record the three largest gaps for each option and price the work needed to close them.
- Compare evidence scores alongside three-year TCO, opportunity cost, risk, and expected outcomes.

Three-year TCO = implementation or build + platform and infrastructure + internal labor + integration upkeep + support and training + expected risk + opportunity cost

Scope and data foundation

1. Have we defined the full IT Management job, not just the asset list?

Why it matters: A useful IT Management platform must support decisions, ownership, lifecycle work, and service delivery across ITAM and ITSM. A list of fields is only the starting point.

Internal build evidence: Document hardware, software, users, locations, custody, lifecycle stages, roles, IT service workflows, exceptions, reports, and approval paths.

Vendor evidence: Map every must-have ITAM and ITSM workflow in a demo or trial. Record gaps, workarounds, configuration needs, and additional module costs.

BUILD 0 ☐ 1 ☐ 2 ☐ **BUY** 0 ☐ 1 ☐ 2 ☐ **Evidence or gap:** _____

2. How will every hardware and software asset enter and remain in the inventory?

Why it matters: Endpoint tools may cover enrolled hardware while missing peripherals, stockroom or remote equipment, network devices, installed software, SaaS applications, licenses, or purchases outside standard procurement.

Internal build evidence: Demonstrate installed agents across Windows, macOS, and Linux, including OS-specific upkeep, plus agentless network, MDM, installed-software, and SaaS discovery, imports, manual intake, and exceptions.

Vendor evidence: Verify cross-platform agents; agentless network and MDM discovery; barcode or QR intake; installed-software and SaaS discovery; normalization; sync frequency; and coverage limits.

BUILD 0 ☐ 1 ☐ 2 ☐ **BUY** 0 ☐ 1 ☐ 2 ☐ **Evidence or gap:** _____

3. How will conflicting records be reconciled?

Why it matters: Discovery produces multiple, changing records. Accuracy requires normalization, matching, deduplication, an authoritative source, and ongoing synchronization at defined intervals.

Internal build evidence: Provide matching rules, source precedence, defined sync intervals, failure alerts, exception queues, correction history, and ownership of unresolved records.

Vendor evidence: Test duplicates, conflicting fields, normalization, source precedence, sync intervals, stale-data or failed-sync alerts, and manual correction history.

BUILD 0 ☐ 1 ☐ 2 ☐ **BUY** 0 ☐ 1 ☐ 2 ☐ **Evidence or gap:** _____

4. Which integrations are required, and who maintains them?

Why it matters: An integration is a recurring operating responsibility. Authentication, schemas, APIs, and vendor limits can change after launch.

Internal build evidence: Name the owner, monitoring method, failure alert, repair process, test plan, API budget, and expected maintenance time for each connector.

Vendor evidence: Confirm whether each connection is native, supported, one-way or two-way, its sync interval, failed-sync handling, and any price impact.

BUILD 0 ☐ 1 ☐ 2 ☐ **BUY** 0 ☐ 1 ☐ 2 ☐ **Evidence or gap:** _____

5. Can the data model maintain context automatically and remain portable?

Why it matters: Context is useful only when relationships among users, assets, software, tickets, locations, services, and lifecycle events update automatically as data or infrastructure changes. Complete records must also remain exportable.

Internal build evidence: Demonstrate automatic relationship updates when users, assets, software, tickets, locations, or integrations change, plus complete exports of records, relationships, history, and attachments.

Vendor evidence: Verify automatically maintained relationships and infrastructure context, sync behavior, complete exports, documented APIs, configuration limits, and migration support.

BUILD 0 ☐ 1 ☐ 2 ☐ **BUY** 0 ☐ 1 ☐ 2 ☐ **Evidence or gap:** _____

Workflows and control

6. How will software entitlements, usage, and renewals be controlled?

Why it matters: Installation counts alone do not show entitlement, allocation, real usage, renewal exposure, or opportunities to reclaim licenses.

Internal build evidence: Demonstrate entitlement records, allocation rules, usage inputs, renewal alerts, reclamation approvals, and audit-ready exports.

Vendor evidence: Test license allocation, usage visibility, renewal workflows, reclamation, reporting, and any limits tied to supported vendors or modules.

BUILD 0 ☐ 1 ☐ 2 ☐ **BUY** 0 ☐ 1 ☐ 2 ☐ **Evidence or gap:** _____

7. Can we prove custody, changes, and audit history?

Why it matters: An audit requires more than current values. IT must show asset custody, software entitlement, what changed, when, and how each record was verified.

Internal build evidence: Show role-based access, timestamped history, custody and entitlement changes, return verification, physical and license audits, retention rules, and exports.

Vendor evidence: Run sample hardware and software audits. Inspect change history, permissions, custody, entitlement reconciliation, sign-offs, and export formats.

BUILD 0 ☐ 1 ☐ 2 ☐ **BUY** 0 ☐ 1 ☐ 2 ☐ **Evidence or gap:** _____

8. How will IT service workflows close every onboarding and offboarding task?

Why it matters: A device return alone does not complete an employee transition. Licenses, custody, access tasks, open work, and exceptions also need owners.

Internal build evidence: Map the service trigger, tickets, tasks, owners, reminders, escalations, exceptions, approvals, evidence, and final closure rule across HR, IT, and security.

Vendor evidence: Test one end-to-end onboarding and offboarding service workflow, including tickets, hardware, licenses, custody, access tasks, reminders, and unresolved exceptions.

BUILD 0 ☐ 1 ☐ 2 ☐ **BUY** 0 ☐ 1 ☐ 2 ☐ **Evidence or gap:** _____

9. Can automation handle routine work without hiding exceptions?

Why it matters: Automation should reduce repetitive effort while making failures visible. Silent errors can scale faster than manual ones.

Internal build evidence: Document triggers, conditions, approvals, error handling, retries, logs, ownership, and a safe manual route for unusual cases.

Vendor evidence: Build and test a representative asset or service workflow. Confirm trigger coverage, approvals, logs, failure alerts, exception handling, and configuration ownership.

BUILD 0 ☐ 1 ☐ 2 ☐ **BUY** 0 ☐ 1 ☐ 2 ☐ **Evidence or gap:** _____

10. Can automation or AI act on reliable, governed data?

Why it matters: AI can query and orchestrate IT Management data across ITAM and ITSM, but it cannot compensate for missing discovery, stale records, unclear permissions, or unreliable integrations.

Internal build evidence: Prove the underlying data pipeline, access controls, source citations, validation rules, human approvals, monitoring, and correction process.

Vendor evidence: Verify what the AI or automation actually accesses, which actions it can take, how results are checked, and where human approval remains required.

BUILD 0 ☐ 1 ☐ 2 ☐ **BUY** 0 ☐ 1 ☐ 2 ☐ **Evidence or gap:** _____

Ownership, economics, and proof

11. Who owns fixes, security, testing, upgrades, and the roadmap?

Why it matters: The build phase ends. Product ownership does not. Every release, vulnerability, dependency, and new requirement creates recurring work.

Internal build evidence: Assign named owners, capacity, budget, release practices, security reviews, QA, documentation, dependency updates, and roadmap governance.

Vendor evidence: Review security evidence, release practices, roadmap process, service reliability, support terms, product ownership, and upgrade responsibilities.

BUILD 0 ☐ 1 ☐ 2 ☐ **BUY** 0 ☐ 1 ☐ 2 ☐ **Evidence or gap:** _____

12. Who supports users and remains accountable when something breaks?

Why it matters: A functioning system needs a route for incidents, questions, training, documentation, escalations, and product defects.

Internal build evidence: Define support channels, coverage hours, triage, escalation, service targets, user training, documentation ownership, and incident communication.

Vendor evidence: Verify onboarding help, support routes, escalation ownership, response commitments, documentation, training, and treatment of known product gaps.

BUILD 0 ☐ 1 ☐ 2 ☐ **BUY** 0 ☐ 1 ☐ 2 ☐ **Evidence or gap:** _____

13. Will the system survive a change in people or providers?

Why it matters: A system understood by one developer is a dependency. A vendor without export and exit provisions creates a different dependency.

Internal build evidence: Require shared code ownership, architecture records, runbooks, credentials management, backup maintainers, recovery tests, and a handover plan.

Vendor evidence: Confirm exports of records, relationships, history, and attachments; contract exit terms; support continuity; implementation records; and a migration path.

BUILD 0 ☐ 1 ☐ 2 ☐ **BUY** 0 ☐ 1 ☐ 2 ☐ **Evidence or gap:** _____

14. Have we compared three-year total cost using the same assumptions?

Why it matters: A subscription is visible. Salaried time and internal risk are easier to omit. Both options also carry implementation and adoption costs.

Internal build evidence: Include discovery, design, development, infrastructure, integrations, QA, security, support, maintenance, rework, training, risk, and opportunity cost.

Vendor evidence: Include subscription, implementation, migration, configuration, integrations, modules, training, administration, vendor management, and platform risk.

BUILD 0 ☐ 1 ☐ 2 ☐ **BUY** 0 ☐ 1 ☐ 2 ☐ **Evidence or gap:** _____

15. What can we verify before committing, and how will value be measured?

Why it matters: A confident estimate is not the same as evidence. The decision needs testable proof and specific outcomes after implementation.

Internal build evidence: Use a working prototype with real data, integration tests, a delivery plan, costed maintenance, security review, and measurable 30-, 60-, or 90-day outcomes.

Vendor evidence: Use demos, trials, documentation, security materials, customer evidence, a scoped implementation plan, documented gaps, and the same outcome measures.

BUILD 0 ☐ 1 ☐ 2 ☐ **BUY** 0 ☐ 1 ☐ 2 ☐ **Evidence or gap:** _____

Decision scorecard

Transfer each score below. Add both columns, then examine the evidence gaps behind the difference.

#	Decision area	Build	Buy	Key evidence or unresolved gap
1	ITAM and ITSM operating scope	___/2	___/2	
2	Hardware and software discovery	___/2	___/2	
3	Reconciliation and scheduled sync	___/2	___/2	
4	Integration ownership and monitoring	___/2	___/2	
5	Automatic context and portability	___/2	___/2	
6	Software and license control	___/2	___/2	
7	Custody and audit evidence	___/2	___/2	
8	Service workflows and transitions	___/2	___/2	
9	Automation and exceptions	___/2	___/2	
10	AI and governed data	___/2	___/2	
11	Product ownership and upkeep	___/2	___/2	
12	Support and accountability	___/2	___/2	
13	Continuity and exit planning	___/2	___/2	
14	Three-year TCO	___/2	___/2	
15	Pre-commitment proof and ROI	___/2	___/2	

BUILD TOTAL: ____ / 30 **BUY TOTAL:** ____ / 30 **DIFFERENCE:** ____

How to read the result

- 24 to 30: strongly evidenced. The path is operationally well-defined, but still needs a competitive TCO and strategic case.
- 15 to 23: partially evidenced. Material questions remain and should be closed before approval.
- 0 to 14: largely assumption-based. Do not commit until ownership, coverage, cost, and proof improve.
- A difference of 0 to 4 points means the decision is close. Let three-year cost, time to value, and strategic focus decide it.
- A difference of 5 or more points signals a meaningful evidence advantage. Investigate the lower-scoring path's largest gaps before proceeding.

Do not use the score alone. A well-evidenced build can still be the wrong use of engineering time. A well-evidenced vendor can still be too costly or poorly aligned.

Record the decision

Decision field	Response
Preferred path	Build / Buy / More discovery required
Decision owner	_____
Decision date	_____
Three-year TCO: build	_____
Three-year TCO: buy	_____
Target date for reliable inventory and workflows	_____

The three assumptions that could change this decision

1. _____
2. _____
3. _____

Outcomes to measure after 30, 60, and 90 days

- ☐ Time to first reliable inventory: _____
- ☐ Manual reconciliation hours reduced: _____
- ☐ Assets or licenses recovered: _____
- ☐ Audit preparation time: _____
- ☐ Onboarding, offboarding, and IT service completion time: _____
- ☐ User adoption and unresolved exceptions: _____

Pressure-test your plan with a working IT Management platform.

[Start a Free Trial](#) | [Explore AssetSonar](#)

Test discovery, automatically maintained IT context, IT service workflows, automation, and offboarding against the needs identified in this checklist.