Operations Management Software Comparison

- Why operations teams compare software before procurement, not after
- Scope: what operations management software includes and excludes
- Method preview: categories, criteria, use cases, decision workflow
- Promise: a repeatable, evidence-based process that reduces risk
- Goal: accelerate time-to-value and defend the decision

Why Software Selection Fails Before Implementation

- 70% of implementations fail; root causes start in selection
- Opening vendor talks before closing internal ones is costly
- Software cannot fix a broken approval workflow
- Vendor choice is 30% of success; adoption is 70%
- The 12–16 week evaluation discipline prevents demo-driven decisions

Root causes start here.

THE RISK SIGNS

Ignoring these signs in selection almost always leads to failure in implementation.

The Operations Software Landscape: Categories That Decide

- ERP owns financial record; MES owns production execution; OMS owns order orchestration; WMS owns warehouse tasks
- CMMS owns assets and maintenance; workflow automation owns cross-system process
- Each category is the system of record for exactly one thing
- Overlap is expected, but ownership must be explicit
- Constraint test: multichannel complexity → OMS, warehouse throughput → WMS, financial complexity → ERP
- Vendors expand beyond their core category, creating comparison risk

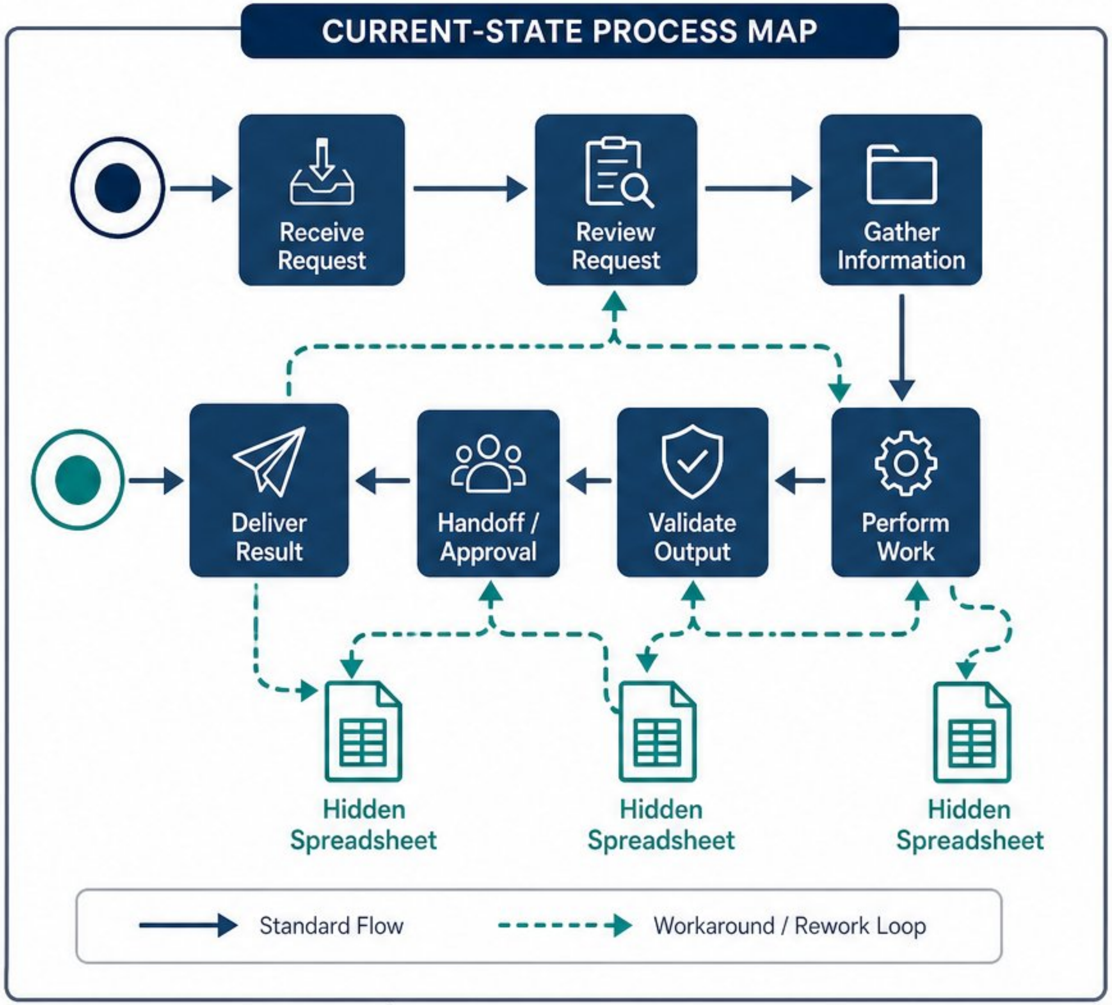
Comparison Criteria That Matter for Operations Teams

WEIGHTED SCALE

Prioritize What Drives Operational Success

- Functional fit leads: 35–45% of total evaluation weight
- Also weigh integration, data flexibility, and user adoption
- Operational criteria: SLA support, exceptions, audit trails, change management
- Avoid feature-count traps; weight criteria against real operating workflows
- Gate requirements: must-have, should-have, nice-to-have before any demo
- Score vendor viability separately from product fit

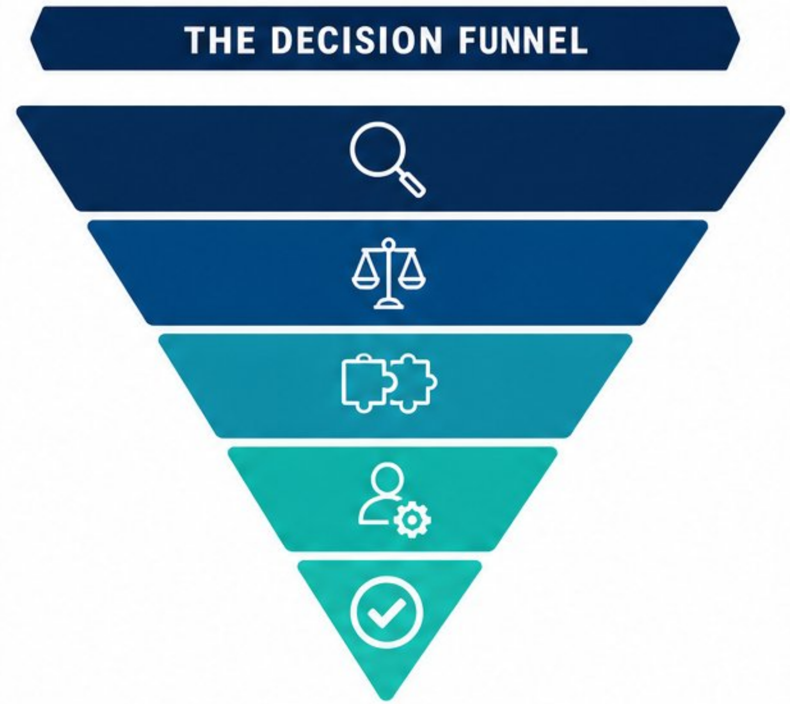
Mapping Process Fit Before Comparing Features

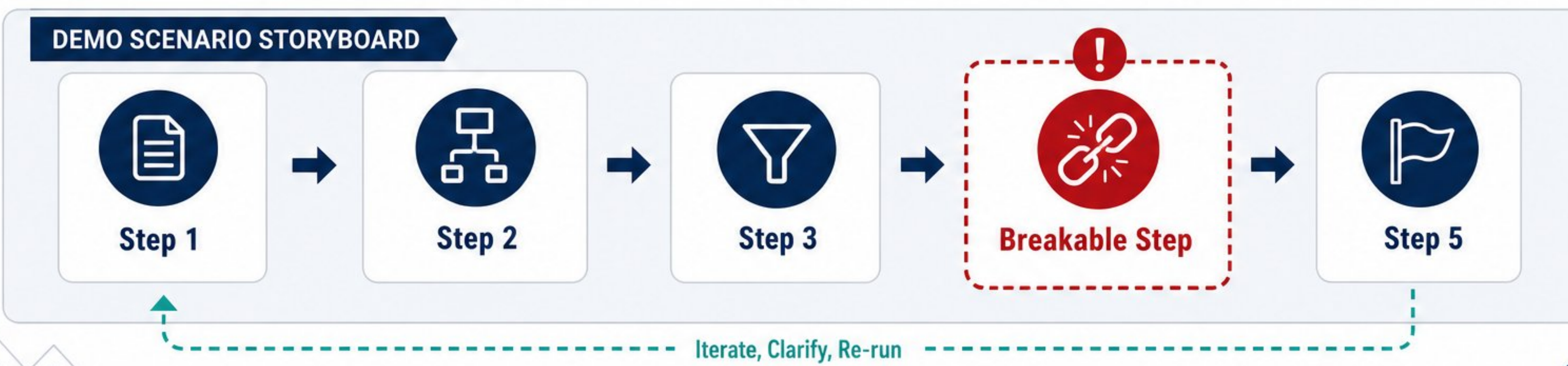


- **Map** current-state processes before comparing software capabilities
- Separate software problems from process problems honestly
- Inventory manual workarounds and hidden spreadsheets as requirements
- Test configurability vs. custom development for edge cases
- Prioritize 6–8 end-to-end value streams over feature lists

Scripting Demos That Prove Fit, Not Theater

- Build 2–5 scenarios from your real workflows, data, and exceptions
- Cover four must-wins: value, risk, integration, and day-2 admin change
- Require live in-product proof, not slideware or recorded video
- Force mechanism clarity: native, configurable, custom, add-on, or roadmap
- Score at the scenario-step level with evidence, scores, and follow-ups



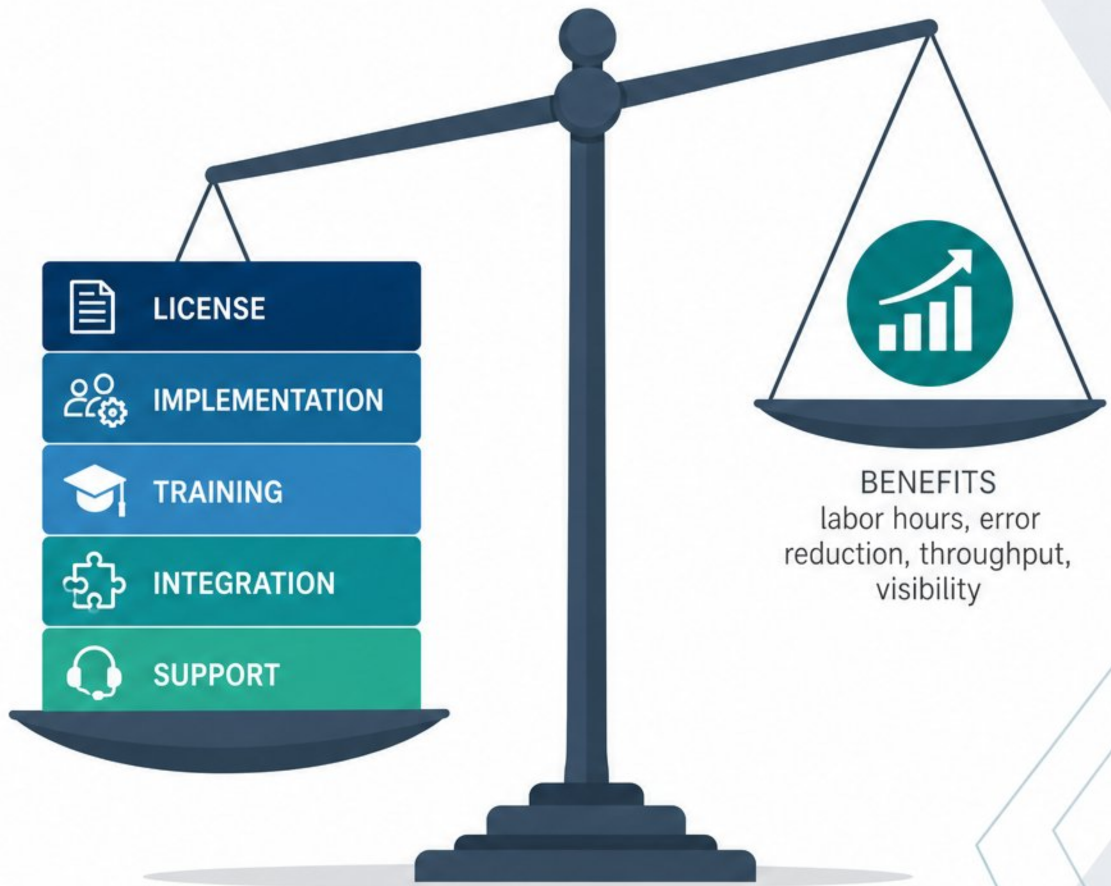


Integration, Data, and Operational Readiness

- Review ERP, finance, and reporting integration needs before signing
- Treat data migration as a business workstream, not a data move
- Dirty data is the most common migration blocker; validate at source
- Confirm deployment model, uptime, support, and release cadence
- Plan cutover, rollback triggers, and hypercare during selection

Total Cost of Ownership and ROI That Survives Scrutiny

- License fees are only 20–35% of total cost
- Implementation services dominate: 30–45% of TCO
- Compare 3–5 year costs: license, implementation, training, integration, support
- Estimate benefits: labor hours, error reduction, throughput, visibility
- Know pricing models: per user, per transaction, subscription, usage-based
- Compare dissimilar pricing models on TCO, not list price



Focus on Total Cost of Ownership (TCO) over 3–5 years, not just list price.

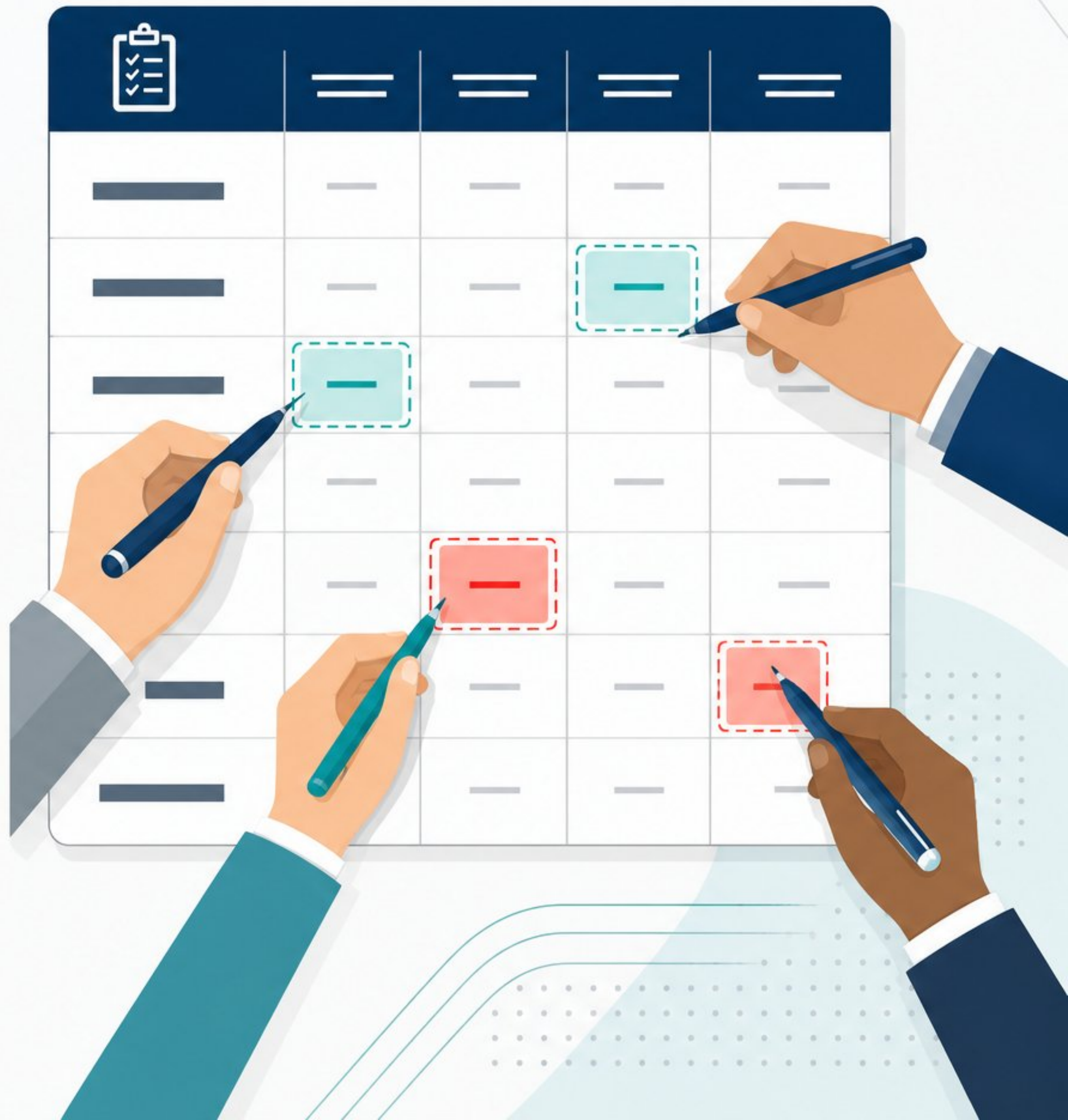
Stakeholder Alignment and Evaluation Team Design

- Required roles: operations leaders, process owners, analysts, IT, finance, end users
- RACI assigns one accountable owner per decision, preventing hidden vetoes
- Keep the core team at 5–7 people; pull in SMEs on demand
- Define decision rights and escalation paths early, with time limits
- Prevent consensus fatigue through structured scoring and facilitated decisions

THE DECISION FUNNEL

Running a Fair Vendor Comparison

- Standardize questionnaires, demos, and reference checks across all vendors
- Weight the scorecard: functional fit, integration, security, TCO, viability, risk
- Score independently first, then discuss divergences of two points or more
- Time-box proof-of-concept and pilot phases to 2–4 weeks for high-risk needs
- Document evidence so the final decision is defensible



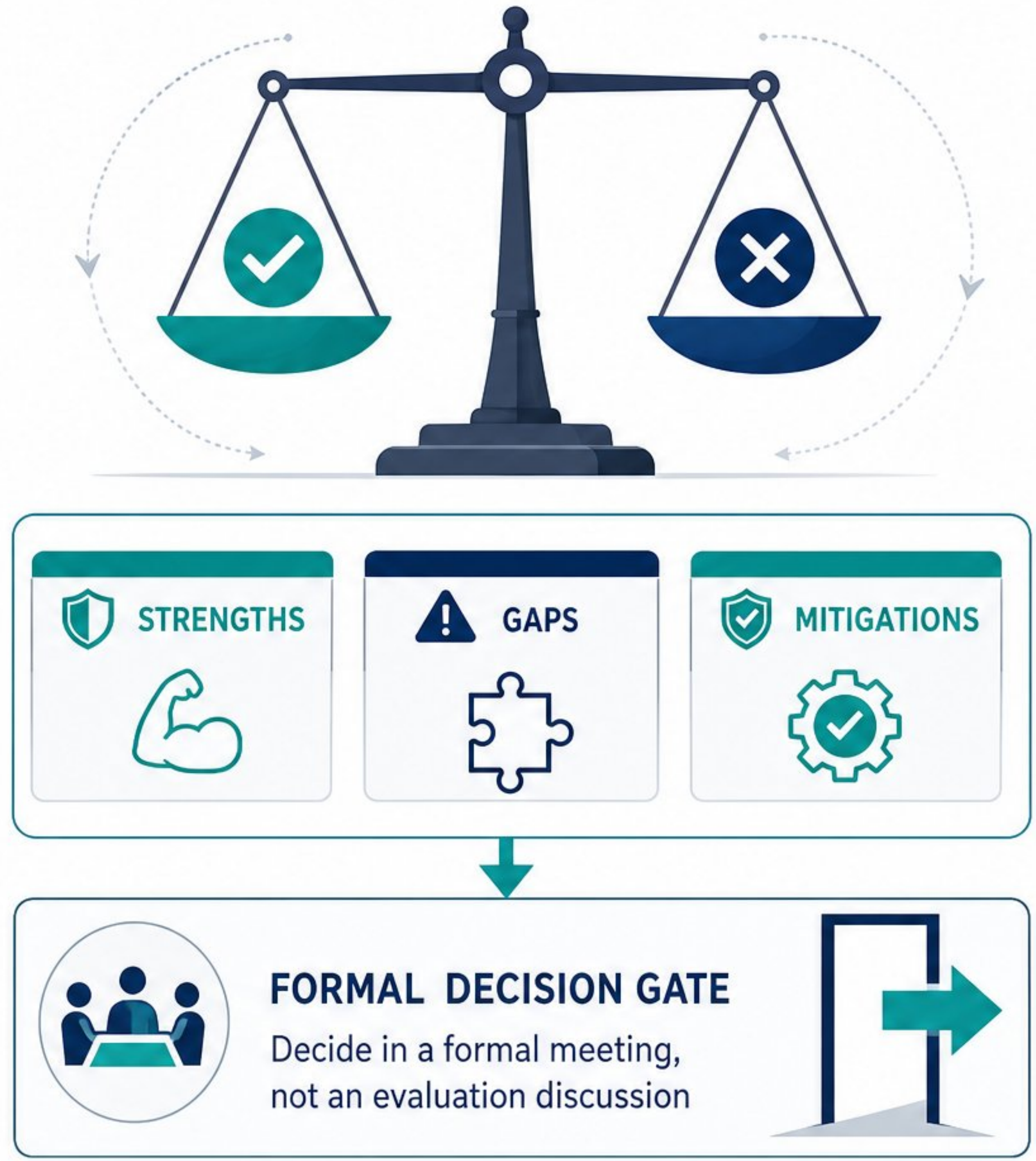
Vendor Viability and Partner Due Diligence

- Score vendor health separately: financial stability, ownership, roadmap
- Partner matters more than product; score implementation partners
- Reference customers at your revenue scale, not vendor showcase accounts
- Vet support model, response SLAs, escalation paths post-go-live
- Contract named team members, not just the demo A-team

THE RISK SIGNS

Decision Framework and Business Case

- Convert scored results into a clear recommendation with explicit tradeoffs
- Quantify business case around operational failures, in dollars
- Run sensitivity analysis: best, base, and worst-case scenarios
- Present a one-page executive summary: strengths, gaps, mitigations
- Decide in a formal meeting, not an evaluation discussion



Post-Selection Transition and Change Management

- Fund implementation, training, and change management before contract signing
- Allocate 10–15% of implementation budget to change activities
- Track 30/60/90-day review cadence with defined success metrics
- Measure adoption via telemetry: active users, transaction volumes, feature utilization
- Plan hypercare with clear ownership and escalation paths

Common Mistakes and a Repeatable Comparison Playbook

1. AUDIT
- Recurring errors: demo theater, hidden integration costs, ignored end users

2. WEIGHT CRITERIA
- Feature-count traps reward polished pitches, not operational fit

3. SCRIPT DEMOS
- Time-box evaluations; lock scope with must-have gates before demos

4. POC
- Playbook: audit, weight criteria, script demos, POC, references, TCO

5. REFERENCES
- Post-implementation review captures lessons for the next evaluation

6. TCO
- Post-implementation review captures lessons for the next evaluation