M&A TECHNOLOGY ADVISORY & DIGITAL DIVESTMENTS

De-risking the Deal.

A practical playbook for acquirers, investors and divesting leadership: how to see inside a target's technology before you sign, cost the integration honestly, and cut over cleanly — including the hidden risks no data room will show you.

3
DEAL PHASES: DILIGENCE, INTEGRATION, SEPARATION

7
DILIGENCE LENSES — CODE TO CULTURE

0
DOWNTIME IN A WELL-RUN CUTOVER

THE UNCOMFORTABLE TRUTH

Most technology risk is invisible until it's expensive.

Harvard Business School research puts M&A failure rates at 70–90% — and technology problems are behind a large share of them. Almost none of those problems were visible in the data room.

Deals are priced on what can be seen: financials, contracts, customer lists, an architecture diagram and a management presentation. But the technology risks that actually destroy value live elsewhere — in unreviewed code, unowned systems, undocumented dependencies, open-source licence exposure, and above all in **people: who really keeps this thing running?**

Industry audits repeatedly find high-risk security vulnerabilities in the majority of acquired codebases, and hidden technical debt routinely shifts deal economics by 20–30%. Yet most technology diligence is still a checklist exercise: a week of interviews, a documentation review, and a report that says "medium risk" about everything.

> **The riskiest part of a deal is never in the data room. It's in how the target actually works when nobody is presenting.**

This guide covers the full deal lifecycle — diligence before you sign, integration after you close, separation when you divest — and the evidence-based method for finding the risks that conventional diligence structurally misses.

IN THIS GUIDE

- **The deal lifecycle** — one discipline from first look to clean break
- **Seven diligence lenses** — code to culture, scored on evidence
- **Hidden risks & unspoken dynamics** — what commit graphs can't tell you
- **Integration costing** — numbers defensible at board level
- **Carve-outs & TSA exit** — separating without breaking what's shared
- **Red flags** — five deal-breakers and the tells that expose them

THE DEAL LIFECYCLE

From first look to clean break.

Diligence, integration and separation are usually sold as three different projects by three different firms. They are one discipline: turning unknowns into evidence, and plans into delivery.

PRE-DEAL

Health check & risk profile

An objective, evidence-based read on the target's technology **before the price is set**: automated codebase scans, architecture review, DevSecOps and cloud-readiness scoring, open-source and disaster-recovery exposure.

Output: a risk profile the deal team can price, not a hedge-everything memo.

POST-MERGER

Integration costing & planning

A bottom-up model of the true cost and timeline of bringing two estates together — application rationalisation, platform consolidation, and the people-and-process change behind every synergy number.

Output: an integration budget that survives board scrutiny — day one to steady state.

DIVESTMENT

Clean-break carve-outs

End-to-end separation delivery: isolating shared infrastructure, untangling dependencies, and executing zero-downtime cutovers that protect customers and the deal timeline — through to TSA exit.

Output: a standalone business, delivered — not a plan for one.

The moments where technology makes or breaks the deal

- Acquiring a business whose technology you can't yet see inside
- Carving out a division without breaking the systems it shares
- Integrating two estates while protecting the deal's synergy case
- Exiting a transitional service agreement (TSA) before the clock runs out

Why one discipline matters: the team that scanned the codebase in diligence already knows where the integration cost hides — and the team that costed the integration knows exactly what a carve-out will snap. Every handoff between firms loses that knowledge, and you pay to rediscover it.

PRE-DEAL DILIGENCE

Seven lenses.
Evidence behind every one.

Proper technology diligence looks through every lens below — and scores what it finds against evidence, not against what management presented.

LENS	THE QUESTIONS THAT PRICE THE DEAL
1 · Architecture & scalability	Can the platform carry the growth the deal model assumes? What breaks first, and what does fixing it cost?
2 · Code quality & delivery	Automated codebase scans: technical debt, test coverage, deployment maturity. How fast can this team actually ship?
3 · Team & key-person risk	Who really keeps this running? What leaves when they leave — and would you know before they resigned?
4 · Security & compliance	DevSecOps posture, vulnerability exposure, incident history, regulatory readiness. Audits find high-risk vulnerabilities in most acquired codebases — assume you're not the exception.
5 · IP & open-source exposure	Licence obligations that can poison proprietary claims; provenance of everything the valuation calls "owned".
6 · Data & AI exposure	Data handling, residency and consent; for AI features — model provenance, training data rights, third-party model dependencies.
7 · Roadmap credibility	Is the roadmap engineering reality or investor fiction? Does delivery history support the promises the price assumes?

Automated scans set the floor...

Codebase scanning gives you objective, comparable, fast coverage of lenses 1–2 and 4–5 across the whole estate — no reliance on the target's self-assessment, no cherry-picked demos.

...but they cannot see people

Lenses 3 and 7 — the ones that most often kill post-close value — are invisible to scanners and spreadsheets. They require a fundamentally different method. That method is the next page.

BEYOND THE DATA ROOM

Hidden risks and unspoken dynamics.

Conventional diligence measures the mechanical: commit graphs, ticket velocity, org charts. But a company's real operating system is social — and it only reveals itself when you watch the work actually happen.

A TRUE STORY

Seven productive engineers. One point of failure.

Assessing a startup for acquisition, we reviewed the engineering team's GitHub history. Seven engineers, healthy commit volume across the board — by every mechanical measure, a balanced, productive team. Then we joined their Scrum sprints and watched the work happen. Within days the real picture emerged: **one engineer was quietly guiding and unblocking the other six.** Design decisions, tricky reviews, every hard problem — all flowed through a single person. The commit graph showed seven contributors; the sprint showed one engine.

That is a critical business-continuity risk — one resignation away from a stalled product — and it was invisible to every firm that had assessed the company before us, because they only looked at the mechanical signals. It changed the buyer's retention plan, the integration sequencing, and the risk priced into the deal.

How we find what others can't

- **We join the sprints.** Sitting inside real ceremonies — planning, stand-ups, reviews, retros — shows how decisions are actually made and who everyone turns to when it gets hard.
- **We watch the flow of help, not just the flow of code.** Who unblocks whom, whose review everyone waits for, whose absence stalls the board.
- **We map the real bus factor.** Not "how many people commit to this repo" but "what stops if this person leaves" — system by system, decision by decision.
- **We listen for the unspoken.** Quiet attrition risk, morale, shadow hierarchies, the roadmap the engineers actually believe in — none of it is in the data room.

> Anyone can count commits. The question is who is actually carrying the company — and whether they'll still be there a year after you buy it.

TWO KINDS OF DILIGENCE

What the scan says. What the sprint shows.

Both are necessary. Only one is standard practice. Every row below is a risk we have seen mechanical-only diligence miss — and price into a deal incorrectly.

THE MECHANICAL SIGNAL SAYS…	…BUT THE BEHAVIOURAL REALITY CAN BE	WHY IT MATTERS TO THE DEAL
Seven engineers, balanced commit volume	One senior engineer quietly architecting and unblocking everyone else's work	Business continuity hangs on one resignation; retention plan and key-person terms must change
High ticket velocity, healthy burndown	Velocity is theatre: tickets sliced thin, hard problems perpetually deferred	The roadmap the price assumes will not arrive on schedule
Org chart shows a full leadership team	Decisions actually flow through one founder; the CTO title is ceremonial	Integration governance designed around the org chart will stall
Documentation repository looks complete	Docs are stale; the real system knowledge is tribal and concentrated	Knowledge-transfer cost is 3–4× the plan; TSA timelines slip
Low attrition in HR data	The two people who matter are already interviewing elsewhere	The asset you are buying may not include the people who built it

Why nobody else does this

Because it can't be automated and it can't be delegated to juniors. It takes senior engineers who have run teams themselves, embedded long enough to see the real dynamics — days inside sprints, not hours in interviews.

What it changes

Retention packages aimed at the right people. Key-person risk priced before signing, not discovered after. Integration sequencing that protects the actual engine of the business — not the one on the org chart.

POST-MERGER

Integration costs you can defend at board level.

Fewer than one in five acquirers actually improves IT costs post-merger. The difference is rarely effort — it's whether the synergy case was ever grounded in engineering reality.

Model bottom-up, not top-down

Synergy targets set as a percentage of combined IT spend are fiction. Cost every system consolidation, licence rationalisation and platform migration individually — with the engineers who will do the work in the room when it's estimated.

Application rationalisation is the prize

Overlapping stacks — two CRMs, two ERPs, duplicated tooling — are typically the single largest bankable synergy. But each consolidation carries people-and-process change; cost that too, or the saving is a mirage.

Plan day one, 100 days, steady state

Day one is about continuity: email, connectivity, customer-facing systems. Full integration — unified ERP, consolidated platforms — takes 12–18 months minimum. Pretending otherwise doesn't compress the timeline; it just breaks trust when it slips.

Track synergies from day one

PMI research shows acquirers that track synergies and integration costs monthly from day one succeed at dramatically higher rates. Distinct metrics for realised value, headcount change and one-time cost — reviewed against plan at project level.

> **A synergy number nobody costed is a promise the engineering team never made.**

The behavioural angle applies here too: integration plans fail on unspoken dynamics — teams protecting their own systems, quiet non-cooperation, key people disengaging. The same embedded observation that de-risks diligence tells you which integration workstreams are actually moving and which are performing motion.

DIVESTMENT

Carving out without breaking what's shared.

A divestment is surgery on a living system: the business must keep operating while you separate everything it shares with its parent. Five principles keep the patient alive.

- **Revenue and customers first.** Sequence the separation around the systems that directly support revenue and customer experience. Everything else can tolerate a workaround; these can't.
- **Accept tactical duplication.** Running two of something for six months is usually cheaper than an elegant shared solution that extends the TSA by a year. Optimise for a short dependency window, not architectural purity.
- **Default to cloud and SaaS landing zones.** Rebuilding the parent's data centre in miniature is the slowest, most capital-hungry path to independence. Land on cloud infrastructure and SaaS unless there's a hard reason not to.
- **Choose the separation pattern deliberately.** Replicate-and-cut, lift-and-shift, or greenfield build — each system gets an explicit choice, costed and sequenced into cutover waves: data migration, application carve-out, infrastructure build, security hardening.
- **Engineer the cutover for zero downtime.** Dual-run validation, rehearsed cutover runbooks, tested rollback paths. Customers should learn about the divestment from the press release — never from an outage.

Untangling the invisible

The hardest dependencies are never in the architecture diagram: the shared script nobody documented, the licence that doesn't transfer, the parent-company employee who quietly operates a critical job. Finding these is diligence work — done on your own estate, before they detonate the timeline.

Run it like a programme, not a project

A Separation Management Office with real decision rights — TSA scope, spend, priorities, risk acceptance — keeps a hundred parallel workstreams honest. Weekly evidence, not monthly status decks.

> **The deal timeline is a promise to the market. The cutover plan is what makes it true.**

THE CLOCK IS RUNNING

Exit the TSA before it exits you.

Transitional service agreements are rented time on the parent's systems — with fees that typically escalate by design. Every month inside the TSA is margin lost and independence deferred.

DISCIPLINE	WHAT GOOD LOOKS LIKE
Scope with sunset dates	Every service tower specified with scope, SLA, price and a hard sunset date. An open-ended TSA is a subsidy from your margin to your former parent.
Objective exit signals	Each tower gets a testable exit criterion — e.g. an independent payroll run validated against a dual run, a full month of operations on the standalone platform. "We feel ready" is not a signal.
Exit in waves	Align TSA exits to cutover waves so each wave retires whole towers. Partial exits that leave one dependency alive deliver all of the risk and none of the savings.
Escalation-proof planning	Typical carve-outs run 6–18 months from signing to full TSA exit; disciplined data strategy and early landing-zone builds can compress that dramatically. Model TSA fee escalators into every timeline decision — speed usually pays for itself.
Independence proven, not declared	The end state is evidence: the business runs a full operating cycle — close the books, run payroll, serve customers — with zero calls home to the parent.

The zero-TSA option

For some deals, building the standalone environment before signing eliminates the TSA entirely — more up-front investment, but no rented time, no escalators, no dependency on a seller whose incentives have moved on. Worth pricing on every deal, even if you don't choose it.

Who does this well

The same engineers who mapped the dependencies in diligence. TSA exit is not a procurement exercise — it's the delivery end of the same discipline that started with the codebase scan.

FAILURE MODES

Five deal-breakers — and the tells that expose them.

Every one of these has sunk real deals. None of them appears in a data room. All of them have tells — if you know where to sit.

RED FLAG	WHAT IT LOOKS LIKE FROM OUTSIDE	THE TELL
The hidden engine	A "balanced team" where one person quietly carries the architecture, the reviews and the hard problems.	In sprint ceremonies, every difficult question routes to the same person — whatever the org chart says.
Demo-ware	A product that performs beautifully in exactly the scenarios management shows you.	Ask to watch a real customer onboarding end-to-end, unscripted. Watch what gets worked around by hand.
Licence time-bombs	Proprietary product claims sitting on copyleft open-source components or unlicensed code.	Automated provenance scans against the full dependency tree — never the target's own inventory.
Security theatre	Certifications on the wall, policies in the wiki — and credentials in the codebase.	Scan results vs. paperwork. Ask when the last real incident drill ran and what it found.
Roadmap fiction	An 18-month roadmap that justifies the price, promising 3× historical delivery pace.	Compare promised velocity with the last year of actual delivery. Ask the engineers — in private — what they believe ships.

The common thread

Every tell requires either objective scanning or embedded observation. None can be caught by questionnaire — which is why questionnaire-driven diligence keeps certifying deals that fail.

The honest question

If your diligence provider's report could have been written without ever watching the target's team work, what exactly did it de-risk?

PULL IT TOGETHER

What good looks like, phase by phase.

Use this as the bar for any technology workstream in your next transaction — whoever delivers it.

BEFORE YOU SIGN

- Automated scans across the full codebase and dependency tree — not a sample
- All seven lenses scored on evidence, with findings priced, not just listed
- Assessors embedded in real sprints — key-person and continuity risk mapped by observation
- Retention plan for the people who actually carry the product, agreed before close

AFTER YOU CLOSE

- Bottom-up integration cost model, owned jointly by deal team and engineering
- Day-one continuity plan separated from the 12–18 month integration roadmap
- Synergies and one-time costs tracked monthly from day one, at project level
- Behavioural check-ins on integration workstreams — motion vs. progress

WHEN YOU SEPARATE

- Every system assigned a separation pattern, cost and cutover wave
- TSA towers with sunset dates and objective, testable exit signals
- Cutovers rehearsed, dual-run validated, rollback tested — zero downtime as the standard
- Independence proven by a full operating cycle with no calls home

> **Evidence before you sign. Honest numbers after you close. A clean break when you leave. That's the whole discipline.**

One team, end to end: the biggest single de-risking decision you can make is continuity — the people who scanned the code and sat in the sprints should be the people who cost the integration and deliver the separation. Knowledge you paid for in diligence should never be lost at a handoff.

PROOF

This discipline, in production.

We led the carve-out and divestment of a $40M technology estate — repositioning a loss-making product company into a profitable professional-services firm within twelve months, while cutting employee churn from 15% to 5%. And as primary technical authority for a global BioPharma group's M&A, we assessed targets' DevSecOps, cloud readiness and open-source risk with automated codebase scans — fast, objective, evidence-based reads before the price was set.

$40M
TECHNOLOGY ESTATE CARVED OUT & DIVESTED

12mo
LOSS-MAKING TO PROFITABLE

15→5%
EMPLOYEE CHURN THROUGH THE CHANGE

WHY ICAN

We've sat on your side of the table

Our leadership has served as CTO of a FTSE 100 company, owning technology decisions at board level. We speak the language of the boardroom and the deal room.

Builders at heart

The same engineers who advise on the deal run the codebase scans, sit in the sprints, untangle the infrastructure and execute the cutover. Diligence grounded in delivery.

We find what others miss

Hidden risks and unspoken dynamics — key-person dependencies, shadow hierarchies, roadmap fiction — surfaced by embedded observation, not questionnaires.

Tell us about the deal.

Assessing a target, planning an integration, or untangling a carve-out — we'll come back with a clear, evidence-based view of the technology risk and the path to delivery.

INFO@ICANGROUP.CO.UK
+44 7470 473113
WWW.ICANGROUP.CO.UK