# The IT Integration Playbook

*What actually has to happen when two companies become one*

**BY RASHAD MORGAN**

Most writing about mergers and acquisitions is about deal structure, valuation, and synergy targets. Very little of it is about the part where someone has to make two Active Directory forests trust each other before Monday.

Integration is where deal value is realized or lost, and IT is usually where it gets stuck. The systems work is knowable. The judgment is in deciding what to standardize immediately, what to leave alone, and how much disruption a business can absorb while it is still expected to hit its numbers.

This is the runbook I wish I had before the first one. It is written from the IT seat. Integration is a team effort across finance, HR, legal, operations, and the business itself, and what follows is the part of it I have been responsible for.

---

## PHASE 0: BEFORE THE DEAL CLOSES

The work that saves the most time happens before you are allowed to touch anything.

**Get what you can pre-close.** Under the existing NDA you can usually obtain headcount by location, a rough application inventory, the acquired company's IT org chart, and whether they run internal IT or an MSP. Ask early. It is often available and nobody volunteers it.

**Establish three things immediately:**

*Who decides when the two companies disagree on a standard.* This sounds like a governance detail and it is actually the most important question in the integration. Two organizations have both made defensible choices about password policy, endpoint restrictions, ticket priority definitions, and a hundred other things. Someone has to own the call. If that is not settled before close, every decision becomes a negotiation.

*Whether the acquired brand survives.* This determines domain strategy, email addressing, identity architecture, and how much of the acquired environment you are migrating versus absorbing. A brand that gets retired is a straightforward migration. A brand that persists means running two identities against one back end, which is a materially different project.

*The Day 1 minimum.* What genuinely must work on the first morning, separated from what everyone would like to work. These are not the same list and conflating them is the single most common way integrations go badly in week one.

**A note on the diligence gap.** IT diligence is usually done by people looking for cost and risk, not by the people who will inherit the environment. The inventory you receive will be incomplete in predictable ways: undocumented integrations, business-critical spreadsheets, service accounts nobody can explain, and at least one application running on hardware under someone's desk. Plan for the gap rather than being surprised by it.

---

## PHASE 1: DAY 1 READINESS

Day 1 is a communications problem more than a technical one.

People at the acquired company are anxious. They have just learned who owns them, and IT is the most visible daily signal of whether the new owner is competent. A smooth first morning buys goodwill you will spend later. A bad one costs you credibility you will not fully recover.

**The Day 1 minimum, in most cases:**

- Email delivers in both directions between both organizations
- There is one phone number and one address for help, and everyone knows them
- Nobody loses access to something they need to do their job
- Leadership can meet, which means calendars and conferencing interoperate

That is close to the whole list. Everything else can wait.

**What deliberately waits:** endpoint standardization, ERP consolidation, network redesign, and anything requiring people to learn a new tool. Attempting these on Day 1 is how integrations acquire a reputation that follows them for a year.

**One thing worth doing on Day 1 that is not technical:** show up. Whether in person or on a call, have the IT leader introduce themselves to the acquired team and the acquired business. The acquired IT staff in particular are wondering whether they still have jobs. You may not be able to answer that, but you can tell them what you know and when you will know more. That conversation is worth more than any tool you deploy that week.

---

## PHASE 2: IDENTITY AND ACCESS

This is the longest pole and the one most likely to break things. It is also where most of the security risk lives, which is why it deserves more deliberation than it usually gets.

**Inventory before you touch anything.** Accounts, groups, service accounts, and the ones nobody can explain. Every acquired environment contains orphaned accounts with more access than anyone realized, and at least one shared credential that several people use daily. Find these before cutover, not during an incident review afterward.

**Decide the target state early.** There are three common paths and they have very different cost profiles:

*Migrate into your directory.* Cleanest end state, most work up front, and the most disruptive to the acquired users. Best when the acquired brand is being retired and the population is manageable.

*Establish a trust.* Fastest to stand up, defers the hard work, and leaves you operating two environments indefinitely if you are not disciplined about the follow-through. Useful as a bridge, dangerous as a destination.

*Run parallel with synchronization.* Reasonable when the acquired company must keep operating independently for regulatory or customer reasons. Adds permanent complexity, so choose it deliberately rather than by default.

The wrong answer here is expensive to reverse. Spend the time.

**Reconcile Group Policy as a security decision, not a technical one.** Two organizations have made different choices about password length and rotation, account lockout thresholds, screen lock timing, local admin rights, removable media, and software restriction. Merging them means somebody's users are getting a stricter environment than they had yesterday, and somebody's security posture is getting weaker if you take the looser standard.

Do not let this default to whichever policy was easier to apply. Put the deltas in a document, decide each one on the merits, and get the decision owned by someone who can defend it.

**Build the deprovisioning path before you need it.** Acquisitions come with departures. Some are planned, some are not, and some happen faster than HR can tell you. If offboarding is manual and you are absorbing a new population, you will accumulate active accounts for people who left. That is an audit finding waiting to happen.

**The thing people underestimate:** service accounts and application integrations authenticate against the old directory and they fail silently. Nothing throws an error a user would notice until a scheduled job does not run, a report does not generate, or an integration stops posting. Enumerate them before cutover and test them after.

---

## PHASE 3: SERVICE MANAGEMENT

You now have two help desks, two ticketing systems, two sets of SLAs, and two teams who each believe their way is correct.

**What has to be decided:**

- One platform or two, and if one, whose
- Whether the acquired team reports into your structure immediately or stays separate through a transition
- Which SLAs apply, and whether the acquired population's expectations were higher or lower than yours
- What happens to historical ticket data

**On the platform decision.** Consolidating onto one system is almost always right eventually, and almost never right on Day 1. The acquired team knows their tool and is already absorbing a change of employer. Give them a runway. Set the date, communicate it, and hold to it, but do not make the platform migration the first thing they experience.

**On historical data.** It usually does not migrate cleanly and usually does not need to. Export it, archive it somewhere retrievable, and move on. The number of times anyone will ask for a two year old ticket from the acquired environment is smaller than the effort to migrate it faithfully.

**On SLA expectations.** This is a trap. If the acquired company had a small, responsive internal team, their users are accustomed to walking down the hall and getting help in five minutes. Your formal SLA of four hours will feel like a downgrade even though it is a more mature service. Name this explicitly with the acquired business leadership before it becomes a complaint.

**On absorbing an inherited team.** They were doing their job competently before you arrived. Start by asking what works in their environment before telling them what changes. The ones who stay become your best source of knowledge about the acquired business, including the undocumented things that do not appear in any inventory. The ones who leave take that knowledge with them, so the retention conversation is also a risk mitigation conversation.

---

## PHASE 4: ENDPOINTS, NETWORK, AND THE LONG TAIL

**Endpoints standardize slowly.** Aligning on hardware, image, and management agent is right, and forcing it early is usually not. The refresh cycle is your friend. Bring new devices onto your standard, enroll existing devices in management, and let attrition do the rest.

**Network integration depends on one question:** do the sites need to reach each other, or just the internet and your cloud services? The answer determines whether this is a modest project or a significant one, and it is worth asking explicitly rather than assuming.

**Applications are the long tail and they will outlast your patience.** There will be a business-critical system running somewhere nobody documented. There will be an integration between two applications that only one person understands. There will be a license that transfers awkwardly or not at all on change of control.

Find these during inventory. You will not find all of them. Budget time for the ones you miss.

**Vendor contracts deserve their own attention.** Acquisitions create duplicate spend almost immediately: two ticketing platforms, two endpoint tools, two monitoring products, overlapping telecom. Consolidating these is the most legible cost saving IT can deliver post-close, and it is often the thing that justifies the integration budget. Get the contract list early, note the renewal dates, and sequence your consolidation to land before renewals rather than after.

---

## PHASE 5: WHAT "DONE" MEANS

Integrations do not finish. They stop being discussed. If you do not define completion explicitly, the work will consume attention indefinitely and you will never get a clean answer about whether it succeeded.

**Reasonable completion criteria:**

- One identity source
- One service management platform and one support model
- One set of standards for endpoints, access, and policy
- Duplicate vendor contracts eliminated or consciously retained
- An environment a new team member could learn without needing to know which company each site used to belong to

**Write down what you left unfinished on purpose.** Every integration has deliberate deferrals: an application nobody wanted to touch, a site that stayed on its own network, a process that stayed manual because automating it was not worth the effort. Documenting these as decisions rather than leaving them as mysteries is the difference between a completed integration and one that quietly becomes technical debt.

---

## THE PART THAT IS NOT TECHNICAL

Every integration is also a change of employer for a group of people who did not choose it. IT is the function they interact with most in the first week, and their impression of the acquiring organization forms fast.

The technical work is knowable. Directories merge, platforms consolidate, endpoints standardize. Any competent team can execute it given time.

The judgment is in deciding what to standardize immediately and what to leave alone, how much change a business can absorb while still hitting its numbers, and when to hold a standard versus when the standard is costing more than it is worth.

That judgment is the thing that does not come from a playbook. It comes from being in the room for it, watching what broke, and being there again for the next one.

---

> *Rashad Morgan leads shared services IT for a multi-brand industrial distribution group, where a merger and three acquisitions taught him most of what is in this document. His part of the work is identity, service management, and end user support.*