



# The Integration Illusion

Why “All-in-One” Does Not Always Mean Ready to Scale

---

**Dan Johnson - Cedar Ridge Consulting**

# The Integration Illusion

---

Most growing businesses reach the same crossroads. You have outgrown spreadsheets and manual work. You are adding channels. You are shipping more orders. You need better reporting. You want fewer tools and fewer headaches.

So the pitch of an “all-in-one” platform feels like the obvious answer. One vendor. One contract. One login. One implementation partner. In some cases, that is absolutely the right call.

But in many cases, businesses later discover a frustrating truth: They did not buy simplicity. They bought a new kind of complexity.

*The question is not “all-in-one vs best-of-breed”. The question is: where do you want the risk to live, and who owns the logic that makes the business run?*

## The bloat tax

All-in-one platforms often expand by adding more modules over time. This can be useful, but it introduces a hidden cost: you start carrying capability you do not need, while the parts you do need move slower than your business does.

- Teams change their workflow to fit the software, even when it makes the operation worse.
- Releases happen on the vendor’s timeline, not yours.
- Simple changes become expensive because they sit inside a larger commercial model.
- Reporting reflects the product roadmap, not your operating model.

## Stitching vs integration

Real integration is behavioural, not just technical. It means the business behaves consistently across systems. The hard part is not connecting apps; it is defining the business logic.

In practice, this is where projects get won or lost. Not in the happy-path demo. In the edge cases you only notice at 4pm on a Friday when the warehouse is trying to get the last carrier collection out.

I have seen teams “go live” and then immediately start running parallel spreadsheets again, not because anyone wants to, but because no one agreed what the system should do when an order is split, substituted, refunded, or re-shipped.

## **The roadmap trap**

If the way you run your business is gated by a vendor roadmap, you are not really in control of your architecture. You are renting it. You only really feel this when a workaround becomes part of your daily operation, and six months later it is still there, and now it has a name, an owner, and a Slack channel.

## **The fair counterpoint**

There are plenty of businesses where one big system is the best choice. Your team can be trained on one platform, and you have one throat to choke when something breaks. That is a rational trade, provided it is a choice rather than an assumption.

## **Choosing where your risk lives**

All-in-one platforms concentrate risk into vendor dependency and roadmap timing. Best-of-breed stacks distribute risk into integration design and data governance. Either route can work, but neither route removes the need for architecture. You are not choosing software; you are choosing a system.

## **Where agentic AI genuinely helps**

Used properly, AI can accelerate delivery by improving the “translation layer” between operations and systems. It can help with process mapping, surfacing edge cases, and generating delivery artefacts like test cases and cutover runbooks. However, what “correct” looks like is still a human decision.

## **The human layer**

AI does not sit with the warehouse team when orders back up. It does not deal with finance when numbers do not reconcile. It can help build the bridge, but it cannot own accountability for how the business stays standing.

## **A simple takeaway**

Who owns the logic that makes the business run, and how will it be maintained as you scale? If you can answer that clearly, the software decisions become much easier. Integrations stop being a risk you fear and start becoming a capability you control.