

Integration Vendor Questionnaire

Built on Brandon Byars, "You Can't Buy Integration", martinoflower.com, 14 December 2021. Checked 2 August 2026.

Ten questions to put to a vendor, or to your own team, before you commit to an integration product. The first four decide the answer more often than the last six.

1. What Language Are The Rules In

Byars's central point is that integration products are not products that directly solve a business problem. They are programming languages, bundled with a toolchain and a runtime. So the first question is what language your business rules end up written in.

- ☐ What language will our integration logic actually be written in?
- ☐ Is that language usable outside this product, or only inside it?
- ☐ If we stopped paying, what happens to the logic we have written?

2. Can We Read A Change

The ability to diff changes between source code commits is, in Byars's words, a developer superpower. Ask whether you keep it.

- ☐ Is the source stored as text we can put in our own version control?
- ☐ Can we see a readable diff between two versions of a rule?
- ☐ Can we review a change before it goes live, as we review other code?

3. Who Can Work On It

- ☐ How many of our people can currently open and change this?
- ☐ How long to train someone who already writes software?
- ☐ Can two people work on different parts at once without conflict?

Byars notes that with these tools it is common to constrain parallel development on the same codebase and split the work into separate components instead. That is a scheduling cost, and it grows with the team.

4. What Does A Mandate Cost

- ☐ Are we being asked to use this tool for all integration, or only some?
- ☐ What happens when a case does not fit the tool?
- ☐ Who decides when an exception is allowed?

Every mandate to use low code tools for all integration will accrue significant accidental complexity, in Byars's terms. The exception process is where that cost becomes visible, so find out whether one exists before you agree to the mandate.

How To Read Your Answers

There is no score. The pattern to look for is whether the answers describe **a language you control** or **a language the vendor controls**.

If the rules end up in a format only one editor can open, only a handful of people can change, and nobody can diff, then you have bought a programming language rather than a solution, and the ongoing cost sits with you rather than with the vendor.

Byars's own conclusion is that the principles behind a modern view of integration are best managed with a general purpose language, and that commercial tools are better aimed at simplifying tactical implementation concerns. This questionnaire is a way of finding out which of the two you are actually being sold.

What This Sheet Does Not Cover

Price, licensing and support terms. Those matter, and they are also the parts a procurement process already asks about well. These ten questions cover the part that usually is not asked until the second year.