THE QUESTIONS YOU'D ASK IF YOU ALREADY KNEW

Seven questions that turn a software decision you've been avoiding into one you can defend

Almost every leader we talk to is stopped cold on some software decision. A renewal that went through because deciding was harder than paying. A replacement everyone agrees is overdue. An "AI add-on" nobody can explain. The freeze is not a character flaw. It is a rational response to three fears: making the wrong call, overpaying, and not having the time to learn enough to tell the difference.

This guide does not try to make you an expert. It hands you the seven questions an expert would ask, so you can ask them yourself, in a meeting, in writing, or across a renewal table, and know what a good answer sounds like. Fifteen minutes. One page you can keep.

1. WHO OWNS THIS WHEN WE STOP PAYING?

The first question, because it changes the answer to every other one. If the software, the workflows built in it, and the data shaped by it all belong to the vendor, then every dollar you spend is rent, and renewal is the vendor's decision, not yours.

A good answer: "You own the application and the code we build for you. It stays portable. If we disappeared tomorrow, it keeps working." **A bad answer:** any sentence that includes the words "export" or "migration path."

2. WHAT DID IT LEARN FROM US, AND CAN WE TAKE IT WITH US?

AI features learn from the work your people do inside the tool. Ask where that learning lives. Most contracts now grant the vendor broad rights to use your data to improve their product. You pay for the seats, and then you pay again by training the thing you rent.

A good answer: the model, the prompts, the rules, and the data stay in your environment, under your account. **A bad answer:** "Our AI gets better for everyone." Everyone includes your competitors.

3. WHAT IS THE PRICE PER PERSON, AND WHY?

Per-seat pricing made sense when software was a desktop license. It makes no sense for a system that does more work as fewer people touch it. Ask what the seat fee buys that a flat or usage-based price would not.

A good answer: a number that goes down when the software does more of the work. **A bad answer:** a number that goes up when you hire.

4. WHAT DOES LEAVING COST, IN MONTHS AND IN DOLLARS?

Not "can we leave." Everyone can leave. What does it cost, in staff time, in re-keying, in the customizations you would lose, and in the contract terms that make the last year the most expensive one. The exit cost is the real price of the renewal, and it is the number vendors most hope you never write down.

A good answer: a written estimate you could hand to your controller. **A bad answer:** "Nobody leaves."

5. WHAT HAPPENS TO OUR PROCESS WHEN THE VENDOR CHANGES THEIRS?

Every packaged tool has a ceiling. Ask what happens when your process needs something the roadmap does not have, or when the vendor sunsets the feature your operation runs on. Your processes work a certain way for a reason. Software that cannot bend to them will bend them instead.

A good answer: "Change the code. It's yours." **A bad answer:** "Submit a feature request."

6. WHAT WOULD A REBUILD ACTUALLY COST, AND HOW LONG?

Most leaders assume custom software means a seven-figure, two-year project. That was true fifteen years ago. Ask for a real number: fixed scope, fixed price, owned outcome. Compare it to five years of the renewal you just signed, plus the exit cost from question 4. The comparison is often not close.

A good answer: a written plan with a price and a date. **A bad answer:** "It depends." It does depend. A serious builder still gives you the range.

7. WHO WILL TELL US THE TRUTH ABOUT ALL OF THE ABOVE?

Every one of these questions has a party on the other side of the table with something to sell. Ask who in the room benefits from your decision going one way, and go find one voice that does not. A written read from someone who is not selling the tool in question is the cheapest insurance a software decision can buy.

A good answer: an opinion in writing, from someone who is not selling the tool in question, that you can forward to your board. **A bad answer:** another demo.

THE ONE-PAGE VERSION

Before you sign, renew, or replace anything, get seven answers in writing:

1. We own it when we stop paying.
2. What it learns stays ours.
3. The price goes down when the software does more.
4. The exit cost is written down.
5. Our process shapes the software, not the other way around.
6. A rebuild has a real number and a real date.
7. Someone not on the invoice has read all of the above.

You do not need to become an expert. You need one answer you can defend.