

Build Vs Buy / Off the Shelf vs Bespoke - The complete picture

When you need new software, what do you do? Does your company have in-house IT? Do you take care of things on your own? Call the tech-savvy gamer kid down the street? Whatever it is you need, companies feel enormous pressure for full digital implementation now.

The decision of whether to build or buy is generally a question asked by companies that have in-house IT or are already partnered with an external firm. Either way, you'll need programming expertise to make the solution fit the need.

Here we will go over the difference between bespoke, out-of-the-box, and internally developed software. We will talk about different situations and the solutions that work best for them.

The first question to ask is:

- What am I looking to change, implement, or build?

Do I want to:

- Add functionality to software that already exists?
- Replace my entire legacy system?
- Create an entirely new offering or service to be used in-house or by customers?

Other things to consider:

- How large is the project?
- How complex is the project?
- How specific does the result need to be?

We all know the speed of technological developments moves exponentially. It is easy not to realize how outdated your systems are and how many other options are available to you. First thing first, in this decision-making process, it is important to remove biases. Next, decide who will be responsible for choosing and where responsibility will lie when it comes to implementation.

If you have a sneaking suspicion that parts of your team are married to particular products or they are stuck in the stone age, consider consulting. Hiring a consulting firm can be expensive, but it may be well worth the dollars compared to the expense of having to "fix" things later.

Now that we've aired that out, let's focus on solutions.

Build vs. Bespoke Basics

In a black and white world, the answer to this question is simple. If you are looking for niche software and are a leader in your industry, you build. If you are looking for a system widely used among different types of companies, you buy.

What happens when you have constraints to manage?

If you own R-Fuel, a used plastic processing plant that turns plastics into recycled fuels, and you are looking for an HR system, you buy something like PeopleSoft. If you are the same small to medium cap company and are looking to update a process for higher efficiency you build it yourself.

R-Fuel is a medium sized company with an in-house IT team and the money to have software developed externally. An engineer developed a new process that will save them millions of dollars a year. They already have the equipment to implement the new process, but the software to run and analyze it does not yet exist.

R-Fuel's IT team is just large enough to keep up with maintenance and the operation's daily needs. They do not have dedicated R&D technology experts. R-Fuel will not find an off-the-shelf solution for what they need, so we will disregard that choice for now.

The company has three basic choices:

- Temporarily or permanently hire developers specific to R&D needs.
- Temporarily or permanently hire IT staff and dedicate current staff to take care of the problem.
- Hire a team outside of the company to develop the new software.

R-Fuel should first identify if they will continually need developers to be working on new processing implementations. They should also consider:

- How long will it take to train new people to develop this software for them?
- How long will it take to get new IT staff up to speed with IT's current responsibilities?
- Will the IT team be able to integrate software developed outside of the company?

The nuts and bolts of making the decision.

Cost of Building

The upfront costs of building will be high. The company requires resources to get the job done. They will need to hire temporary IT staff to fill their day-to-day needs and educate the staff they plan to dedicate to the job.

R-Fuel also risks paying an opportunity cost because development of new software will slow operations for a time.

Cost of Buying

The cost of paying a third party will be much less than building it in-house. Once the software is integrated into the systems they have in place, new hires would not be necessary. This saves the company money in terms of taking on new people and opens them up to the possibility of gaining technology they had not been using or using correctly.

Time

Time is not so much a factor here; newly dedicated employees should be able to turn out a product just as fast as a team outside of the company. R-Fuel risks being hit with opportunity costs from both ends. The new process will allow them to save money, but they will need to shuffle people around and could be missing valuable insight from a group that understands new technology in the space.

Upkeep and Long-Term Costs

Maintenance of the new system as well as data acquisition and analysis will be similar to processes that are already in place. If R-Fuel fully purchases the software rights from a developer and keeps the systems they already have in place, R-Fuel would save money in the long run by paying a third party to develop the software for them.

For a company in R-Fuel's position, buying a prototype and working it into their system makes a lot of sense. This is an example where buying and building are required to fully meet the needs of the company.

This is just one speck in the large spectrum of the "build it or buy it" scenario. Let's look at a company with outdated technology looking to move into the digital age.

Dinosaur Legacies

Elder Health, an elderly care facility, recently had a group of doctoral students running an experiment for a publication they were working on. The residents and others that visited the facility loved what the students had done and petitioned the facility to make an app that would monitor residents of the facility and outpatients in the same way the students had.

The IT team at Elder Health claimed that they could create the app in a couple of weeks. They created the app, and it would not integrate well with the legacy system they had in place. Elder Health has run into a major problem. Their legacy system is out of date and cannot handle new technology.

This is a problem many large organizations run into. They need new technology to be integrated, but the structure they have in place is simply unable to support it. Rather than overhaul the entire system, companies patch things together until they finally break.

Much of the time, this has been done to save money upfront, but in the long run, it can be costly. With many companies forced to finally see the digital light, issues with legacy systems have been brought to the forefront.

Elder Health has some decisions to make. They want to keep long-term costs down but are willing to pay upfront to modernize.

As things currently stand:

- All data is stored in-house.
- They do not use cloud computing.
- The IT team is large enough to take care of daily operations.
- Most of the in-house equipment has not been updated in over 15 years.
- Their data is siloed and not always easy to access from every application that uses it.

We can conclude that Elder Health needs some serious help. Should they build a new system on their own, or have it done by a third party? They already have a new application ready for it when it's up and running.

Benefits to Building the System on Their Own?

It can be tailored exactly the way they want it.

- This is not to say that a third party would be unable to deliver what is needed. In fact, much of the system will contain software that has already been developed and widely tested. Elder Health will be able to build and scale it in real-time while being sure that the system meets their needs.

Full Authority Over the Solution

- Elder Health will have total control over any implementations created by the IT team and can do whatever they please with them. If they stumble on a new solution that can be used in the healthcare industry, they will have the rights to distribute the software. This could give them a competitive edge.

Disadvantages to Building the System on Their Own

Cost

- Having the IT team build an entire legacy system on their own will cost a lot of money. Additionally, they must be sure that the team understands what products will work best for the organization. A poorly informed team could end up costing a lot more than the time and equipment to create a new system.

Time

- It will undoubtedly take more time for the team in place to design and implement a new system than a third party that specializes in modernizing legacy systems.

Benefits of a Bespoke Product

Cost

- The labor and opportunity costs will be lower if Elder Health finds a firm specializing in modernizing systems. The organization could offload some on the day-to-day IT work and future maintenance by using a firm that will host and manage the bulk of the system. Doing this could help them save money in the future.

Time

- Offloading the work to another company specializing in what Elder Health needs will ensure the speedy delivery of the final product.

Maintenance

- Using a product supported by someone else ensures that you are not responsible when things break down. The supported product will also be updated as new software hits the market.

Disadvantages of a Bespoke Product

Control

- The organization will have less control over updating and changing features within the software.

Cost

- The longer-term costs could increase with contract renewals and additional subscriptions.

Compatibility

- New products or solutions will have to be configured to run on the new system.

What is a good game plan for Elder Health?

Elder Health may benefit the most from buying and building. They need a new solution now, and they have the resources to service it themselves once it is running. A business in this situation with some money and a knowledgeable IT team would benefit the most from consultation and initial implementation from a third party.

This way, Elder Health gets the solution they need fast, and they are able to retain control of their systems over the long run. In this case, it would be best to consider intermittent consulting from IT professionals in the space. This will help ensure their technology is always up to date, and they will not run into problems like this again.

Here is an example of a small company with no IT team but the money to hire one if needed.

Technologically Stunted

After a decade of retail-only distribution, the CEO of million-dollar Bow-Wow Chow realized that direct-to-consumer would be a much better fit for his brand. He came up with a great new idea for an application connected to a smart dog dish that will let the owner know when his puppies need to be fed and how much. The dish comes with an attachment that will also feed the dogs for the owner if desired. The company employs about 20 people and interacts with distributors and vendors of all sorts.

He has a grand vision for a flagship store displaying his amazing new dog dish. Currently, Bow-Wow Chow has no IT, and R&D is limited to Mr. Bow-Wow and his dog. With no IT team, this company needs to decide if they want to take on that expense at this point.

Right off the bat, Mr. Bow-Wow's dream of worldwide distribution of his Hungry Puppy app and dishes become almost unreachable. He can barely keep up with his QuickBooks entries to stay on top of the accounting, let alone learn how to make a smartphone application to integrate with his dog dish.

Mr. Bow-Wow would greatly benefit from utilizing a third-party IT and development team. The benefits in this situation are clear. Mr. Bow-Wow can have his application built and managed by a third party.

If the application becomes hugely popular, Mr. Bow-Wow plans to make an arrangement where he will have all the rights to the software and can hire an IT team to service the product. As long as he retains his rights, using a third party to develop his app is the cheapest way for him to test it out on the world's hungry puppies.

Breakdown

From the three examples above, there are great reasons to build in-house and purchase bespoke software. Here are the benefits and drawbacks of building in-house and buying.

Benefits of Building

Control

When you build, you will have complete control of what goes into the software and how it functions. Any design implementations that you wish to make can be made the moment you envision them. Not only that, but you have full ownership of what has been created and can do anything you please.

Design

You have 100% control of the design. You can customize the software to any end. You will also be able to add new functions and features when and how you want. You will also ensure that future implementations and designs will be compatible with what you have developed.

Leg-Up

By developing your highly customized software, you could be giving yourself an edge over your competitors. If your competitors are utilizing off-the-shelf solutions that provide less efficiency, you will be putting yourself at an advantage.

Drawbacks of Building

Costs

When building your own systems or software, you must dedicate employees to the task. This could take away from them being able to perform their daily duties or cause you to need additional help. It will be important to weigh these costs along with the cost of maintenance and upkeep you could incur by hiring a third party.

Opportunity Cost

Depending on what your business is primarily focused on, the cost of building software solutions could be seen in lost revenues. This will happen when employee attention is focused on building solutions rather than products. Other opportunity costs arise when IT teams are not fully educated in the solutions you are looking to implement. If your team is good at what they do but doesn't know a whole lot about developing your solution, you could lose money by developing something that is inefficient.

Time

If the software you need is a pain point for customers, then you need a solution now. Understanding the workflow and designing a new solution in-house could be much more time-consuming than if you hire a dedicated team to do it for you.

Benefits of Bespoke Solutions

Cost

Your upfront costs will be lower using a third party. A third party with experience in your industry will already have modern solutions they can personalize and provide to you.

Time

A dedicated third party will be able to develop your solution rapidly. They should also be able to tailor it to your needs depending on what that solution is.

Maintenance

If you are replacing a legacy system or looking for external IT services, your maintenance and updates will be taken care of. If this is a solution that the third party provides industry-wide, you will receive updates and service for as long as you hold a contract with the firm.

Drawbacks of Bespoke

Long-Term Costs

Contract maintenance costs can change from contract to contract. You could also be charged for updates, and if you expand in any way, you can expect to pay more.

Control

Or lack thereof. Depending on how you have organized the agreement. You could have less control over your new software's functionality and what can be done with it in the future.

What Do I Choose?

There are many situations where the answer is easy. There are many more situations where it is not as clear. Evaluate the scope of your project. Are you looking to add one or two new features, or are you looking to overhaul your system entirely? Then consider the following questions:

- Are you going to want the rights to the finished product?
- Will you need an external company to service the software or system in the future?

If you have the money and employees to build, consider looking outside of the organization for inspiration in new methods and technology. Then take what they give you and build from that. Remember to, search out a firm that is willing to work with your specific needs.

Making the decision is not always easy, use this guide and the examples above as a tool to visualize your situation and options.