Prismatic

The Guide to

# Integration Strategy for B2B SaaS

# Table of Contents

# Introduction

Integrations are essential to the value proposition for B2B software products. Business customers expect modern software to provide seamless, out-of-the-box connectivity to the rest of their tech stack.

Today's software companies face a considerable challenge because of this expectation. Along with their apps, these companies must deliver dozens – sometimes hundreds – of integrations to their customers' other apps and services. Often, these integrations are table stakes for getting to market, winning deals, and retaining customers.

And, while customers usually think of the integrations as "data from app X goes to app Y," there is so much more to an integration than it appears at first glance.

# How can you provide these integrations?

Given all this, how will you quickly build the integrations your prospects and customers need? How will you maintain and support them? As your customer base and integration demands grow, how will you do this at scale? Can you do it without using too many dev resources or neglecting your core product?

These are only a few of the questions you should ask as you determine your integration strategy. We've compiled this guide to provide insight into the topics that should be part of your integration strategy discussions.

This is not a step-by-step guide because the specific strategy you'll formulate is based on what you and your customers currently need. And that strategy will change over time as your product matures and the market changes, as it always does.

## Why are integrations essential for B2B SaaS companies?

As noted, integrations are table stakes in many markets – a non-negotiable requirement for building a viable product, closing sales, onboarding customers, and retaining them.

Several significant trends drive customers' expectations for more and better integrations, making integration delivery a critical part of B2B SaaS companies' strategies. Here are those trends:

- ✓ **The proliferation and specialization of SaaS make connectivity essential.** As companies regularly add new apps to run their businesses, it's even more important to buyers that each app serves its core function and fits into the company's larger tech ecosystem. Today's business customers expect any new software they purchase to connect to the other apps they use.

- ✓ **The pace of automation is accelerating.** Businesses are automating processes and workflows to cut costs and boost productivity, increasing integration demand.

- ✓ **Integration expectations have come down-market.** Smaller companies that don't have the IT resources to build those integrations still expect them.

- ✓ **Business users expect less IT involvement with their apps.** Users want to be able to set up and operate their apps without constantly relying on IT. This thinking also extends to integrations.

## How can you use integrations as a competitive advantage?

Beyond simply building the integrations that prospects and customers require, product leaders who create a strong integration capability can use it as a competitive advantage.

By using native integrations, you can:

- ✓ **Increase product value.** Integrations can extend your product, sometimes in outsized ways. For example, a project-tracking app that integrates with its users' calendars and messaging systems provides much more value to its users than one that does not.

  > *"It was easier for us to identify the data coming into the system and to add fields. Now we could pull that data in and all of a sudden it was unlocking additional functionality for customers."*
  >
  > *Adam Jacox, VP of Engineering for Hatch. Read Hatch's full story here.*

- ✓ **Make your product the center of your customers' ecosystems.** When your product is the software integrating your customers' business apps, driving their data flows, and automating their processes, it becomes the hub of their ecosystems.

- ✓ **Increase product stickiness and reduce churn.** Integrations can provide a strong moat, making your product hard to replace. Businesses don't want to take a step backward in automation and connectivity. If your product is deeply integrated, replacing it requires your customer to find another product with the same integration.

  > *"Customer retention and annual revenue per user or subscription, both of those have significantly changed since we've got the Prismatic embeddable solution baked into our product. Our churn month over month has gone down, I'd say, almost 3%."*
  >
  > *Frank Felice, CRO of Sisu Software. Read Sisu's full story here.*

- ✓ **Be strategic about what not to build.** When the market asks for a feature with serious R&D effort or one that doesn't fit your roadmap, you can choose not to build that feature. Instead, integrate with an app that provides it. This lets you meet customer needs while focusing on what your app does best.

# Common approaches to integration delivery

Historically, software companies have taken various approaches to customer integrations. Most likely, you have experimented with one or more of those approaches and have experienced some of the pain points for yourself. Let's briefly review the ways SaaS companies commonly deliver their integrations.

## Put the integration responsibility on the customer

While placing the burden on customers to build their integrations may seem easy, this is a non-starter in many markets. This is especially true if you serve small businesses that don't have IT personnel. If you can't provide the integrations your prospects need, a competitor will step into that gap.

## Use a third party to build integrations

Involving a third party to provide integration services to your customers does reduce work for your team. However, there are drawbacks:

- ✓ You give up revenue.
- ✓ You give up control over integration quality.
- ✓ You add a third party to your customer relationship.
- ✓ Customers in many markets won’t accept it.

## Use an enterprise iPaaS and build around it

Some SaaS teams find creative ways to use enterprise iPaaS to offer integrations to their customers.

Enterprise iPaaS solutions are made for businesses of all kinds to build internal integrations. They offer low-code integration designers and large SaaS connector libraries that make it easy to build one-off integrations. But providing integrations to your SaaS customers adds complexity that an enterprise iPaaS does not address:

- ✓ Enterprise iPaaS doesn't have the concept of customers or customer-specific config. Therefore, it does not offer a way to build reusable integrations that are deployable to multiple customers. Teams that use enterprise iPaaS for customer integrations must deploy an instance of the iPaaS or rebuild similar integrations for each customer.

- ✓ These solutions rely heavily on a no-developer, low-code mindset, leaving teams stuck when building a complex integration or connecting an unsupported app.

- ✓ You can't easily incorporate an enterprise iPaaS into software teams' existing systems. As a result, you end up with integration tools and processes separated from the rest of your dev environment.

- ✓ Enterprise iPaaS can't deliver integrations with a native, in-app experience.

## Build integrations and integration tooling in-house

It is common for some SaaS companies to build integrations in-house. You get the strategic benefits of integrations, keep the revenue, and control the quality of the integration experience for your customers.

But most companies find that building integrations in-house scales poorly and creates other concerns

- ✓ Building integrations and tooling in-house requires a dev effort that often distracts from core product work.
- ✓ Teams typically accrue a long backlog of integrations, slowing sales and onboarding.
- ✓ Getting integration infra right is hard, especially with security and scalability concerns.
- ✓ Your dev team spends time on things irrelevant to your app domain, like integration logging.
- ✓ Building the tooling for non-devs to deploy and support customer integrations is too time-consuming and expensive to do right. As a result, most teams don’t do it at all, putting integration deployment and support on your devs.

## Use an embedded iPaaS

There is another option, and that is to go with an embedded iPaaS. But before we get into how an embedded iPaaS might pertain to your scenario, let's look at a few broader integration topics you should discuss as you plan your integration strategy.

# What about technology partnerships?

If you plan to build integrations from your SaaS product with third-party apps, technology partnerships (tech partnerships) are in your future. Considering what form those tech partnerships may take should be part of your integration strategy. Tech partnerships may be very loose, with neither company establishing a formal partnership arrangement. Or the relationship may entail NDAs, partner agreements, and other legal documents.

If both partners use open APIs, the relationship can be very informal. However, if they use public or partner APIs, there tend to be more rules defining the partner relationship.

Tech partnerships based on integrations can substantially impact your company and your position in the market.

Here's a detailed look at some of the benefits of tech partnerships:

- ✓ **Direct value for customers.** Tech partnerships can provide easily measurable value to your customers. And they often do so in outsized ways. When customers have complicated manual processes, they often grow accustomed to them (and the time and money it takes to maintain them). However, the time savings can be substantial when you partner with other SaaS companies and provide your shared customers with integrations that automate formerly manual processes.

- ✓ **Improved product focus.** There are some things you shouldn't build. Customers often ask you to add functionality to your app that doesn't fit your vision for your product or that you'll build eventually but isn't a priority now. Tech partnerships allow you to keep your app focused on its core purpose while providing integrations for that additional functionality.

- ✓ **Increase product stickiness.** We've mentioned direct value to the customer, but the customer experience goes beyond that. Your ability to integrate with tech partners can make your product a central piece of your customers' ecosystems. You are providing your app and enabling critical communication with other apps on which they rely. As a result, your app becomes stickier, and customers are less likely to churn.

- ✓ **Expanded marketing and sales reach.** There is value in leveraging tech partners to get your brand into market segments you are not reaching today. More exposure generally leads to more sales.

And now, here are some of the challenges you should expect when you become a technology partner:

- ✓ **More relationships to manage.** When you set up a business partnership, your partner is, by definition, essential to your business. And that means you need to cultivate and manage that relationship. This involves everything from giving partners advance notice of technical changes affecting your integrations to simply talking with them periodically to ensure everything is going well.

- ✓ **Integration investment.** Building integrations can require a substantial investment. It's not uncommon for an integration to take weeks or months to move from initial requirements to production. Working with tech partners can add to that timeline as you work with your tech partner's resources and scheduling constraints.

- ✓ **Resource allocation.** Adding integrations to your app (and doing so with the support of a tech partner) is great for your customers. What's not so great is that integration development can cause your devs to lose their focus on your primary app. Continuing development on your primary app while also building customer integrations can be tricky.

For more details on tech partnerships, check out our post on SaaS technology partnerships.

## Which integration should come first?

One of the most frequent integration questions we hear from SaaS teams is, "Which integrations should we build first?" And this question should absolutely feed into your B2B SaaS integration strategy discussions.

Outside of core product development, integration development may be the most important thing you can do to define the future of your product.

You may have identified dozens of integrations you'd like to build, but which ones should come first? Sales can give you all sorts of data about which missing integrations impact the pipeline. Your success team has a list of customer requests they would like to deliver. Engineering has a feel for the level of effort and knows which ones it would like (or not like) to build.

Let's look at some strategies and tactics for prioritizing integration development.

## Strategies for prioritizing integration development

We have found three common approaches that successful SaaS companies use – often at the same time – to determine which integrations to build first.

### Which integrations are most needed by customers and prospects?

Depending on your product, you may have a largely homogeneous customer base, or your customers may have substantially different integration needs based on the other software they use. That can vary greatly based on company size, industry, sophistication, and region.

Some situations are straightforward. Maybe 40% of your customers use the same third-party accounting system, with the other 60% spread over a score of different accounting systems. Integrating your product with that first accounting system will bring needed efficiencies to a substantial portion of your customer base.
But what if the data isn't that obvious? What if only 10% of your customers use the same third-party accounting system? Is that enough reason to work on that integration first? Knowing that one of those accounting systems is gaining massive market share among your ICP or that another 10% of your customers will shift to it in the next year can give you the data you need to prioritize.

### Which integrations will extend your product's functionality and SAM?

Or which integrations will supply the desired functionality to your product and help it do more for your customers? In many cases, integrations can expand your serviceable addressable market (SAM) by providing functionality that solves the problems of larger prospects, more sophisticated prospects, or prospects in a new industry or region.

In some situations, building that functionality directly into your product makes sense. But sometimes, adding certain functionality would take your product in a direction that detracts from your core value proposition, making it better to integrate with an app that already has that functionality.

### Which integrations will increase your competitive advantage?

Maybe you've had your product on the market for a while, and it's mature, and you've built a few integrations for it but are struggling to differentiate yourself from your competitors. Or you don't have any product integrations, but you are considering an upcoming major release of your product and would like to do something to make your product stand out from the competition.

How often have you seen "integrates with X" as a value proposition on a SaaS homepage? If you don't have the integrations your prospects want/need, they'll go with someone who does.

## Tactics for prioritizing integration development

These strategies are essential for big-picture, long-term planning. Most SaaS companies use a combination of them and ultimately look at the overlap to choose which integrations to build first.

Your planning should also include flexibility for handling integration needs that pop up periodically, often more frequently than you might want them to. Let's look at some common scenarios where a tactical approach is necessary.

## You need an integration to win an in-flight deal

Sometimes, saying yes to adding a prospect's integration requirement makes the difference between winning and losing the deal.

The key here is to agree to those integrations that are also needed by other prospects and customers, add value to your product, or help your competitive position.

## You need an integration to keep a customer

Sometimes, a missing integration causes enough pain or inefficiency to a customer that the customer considers churning – especially if a competitor has or agrees to build that integration.

Leaving aside whether you want to keep that customer, you should ask questions about the integration. Is it a current gap in your lineup? Does it make sense for your overall product plan?

## You need to replace an existing integration

Sometimes, an existing integration no longer meets customer needs. Ideally, you've planned for integration replacements and worked them into a long-term strategy. Communication is key. Let your customers know what's happening, how you plan to address it, and when and why they must migrate to the new integration.

For more details on prioritizing integrations, check out our post on how to prioritize integrations for your B2B SaaS product.

# How important are requirements for integrations?

Integrations are hard. By definition, an integration connects two or more systems to enable data exchange. These systems are often using disparate databases, auth, and APIs. And it's common for these systems to be under development also, which means you are trying to hit a moving target.

That said, using well-defined requirements to build integrations can speed up the time-to-market for your product's integrations, reduce development, deployment, and support costs, and ensure that end users are pleased with the results.
In short, collecting requirements is an essential part of your B2B SaaS integration strategy.

## What are integration requirements?

Integration requirements describe what an integration does and how it is expected to function. A collection of these is called a specification.

We sometimes talk about high-level versus detailed requirements. Here's an example of a high-level integration requirement: "The integration syncs order data between the CRM and Accounting systems."

If we were to start developing an integration based on this requirement, we would end up in an unknown space very quickly. High-level requirements help us define the box, but we need detailed requirements to fill the box.

## Why are requirements critical for integrations?

Requirements for integrations are often considered unimportant compared to requirements for other software. This thinking about integrations stems from a simplistic understanding: "It just copies data from one system to another. How hard can it be?"

Integrations are much like icebergs. Most of the integration isn't visible. So, if we only create requirements for what we can easily see, we'll build something ill-suited to the underlying business need.

Integration requirements are essential to establish an agreed-upon integration scope for all the stakeholders, from engineering to end-users. Yes, even if your devs take an agile approach to integration development, solid requirements make the difference between success and struggle.

## Common questions for integration requirements gathering

The nice thing about gathering requirements for integrations is that we know, at the highest level, what integrations do. This knowledge allows us to narrow the scope of requirements questions from the beginning.

Here are some standard questions to solicit integration requirements:

- ✓ What will cause the integration to run, and how often will it run?
- ✓ Will this be a one-way or two-way integration (or something else)?
- ✓ What auth method will each system use?
- ✓ What API will each system use?
- ✓ What data is to be transferred?
- ✓ What transport languages will be used?
- ✓ What transfer protocols will be used?
- ✓ Will any data need to be modified within the integration?

## Tips for integration requirements gathering

If you've received detailed answers to all the questions in the preceding section, you may not have a finished integration requirements specification, but you should have a good draft.

In addition to those specific questions, here are some tips to inform your overall process and help you turn that draft into a solid specification:

- ✓ Talk to end users
- ✓ Talk to more users
- ✓ Seek out corner cases
- ✓ Engage with your tech partner

For more details on integration requirements, check out our post on how to gather integration requirements.

# How do outcome-driven roadmaps help with integrations?

Software roadmaps have been around forever. However, the idea of outcome-driven roadmaps is relatively new. You've likely heard about it recently, and we believe it's a great way to focus on results instead of intentions.
You may think it is "easier said than done" to include integrations in your product roadmap. After all, integrations are often tricky to fit into a roadmap because they can be challenging to scope. Among other issues, estimates must be revised frequently due to changing requirements and conflicts with tech partners' priorities.

But, placing integrations into your outcome-driven roadmaps should be a key part of your overall B2B SaaS integration strategy.

## What are outcome-driven roadmaps?

Many things are called software roadmaps that are not. A list of product features and the dates you expect them to be available to customers is not a roadmap but a product release plan. And release plans have their place. But a product roadmap lets us see how the product will grow and change over multiple releases. In short, the release plan is tactical, but the product roadmap is strategic.

That means we must address the why rather than the what. The why leads us to outcome-driven roadmaps because they focus on addressing business needs and changing customer behavior.

Let's look at a few items written for an outcome-driven roadmap:

- ✓ Cut onboarding time for new users by 20% by incorporating an interview module.
- ✓ Reduce support tickets tied to unreadable/confusing UI elements by 50% in Q2.
- ✓ Save 10 current customers 30 minutes daily by building an integration with X.

By focusing on the why, you can see how properly designed outcome-driven roadmaps allow you to achieve valuable results for your company, your customer, or (ideally) both.

## What are the benefits of outcome-driven roadmaps?

When you shift from output-focused to outcome-driven roadmaps, you'll see the following benefits:

- ✓ Product differentiation
- ✓ Flexibility for market changes

- ✓ Recency bias protection
- ✓ Company goal alignment

## How to include integrations on outcome-driven roadmaps

Integrations are product features. And if you are putting features on your product roadmap, it's not outcome-driven. The key is to link specific integrations to the desired outcomes. To do that, you should:

- ✓ Work backward from your company goals
- ✓ Dig into the data
- ✓ Make the outcomes SMART (specific, measurable, achievable, relevant, and time-bound)

For details on incorporating integrations into outcome-driven roadmaps, check out our post on how to create outcome-driven roadmaps with integrations.

# How do you price your integrations?

A common question we hear from our customers is, "How do we price integrations?" That makes sense. They're usually in the process of implementing our embedded iPaaS so they can more efficiently build integrations between their product and their customers' other apps. And they are thinking about how to best capture value from all the integrations they can now offer their customers.

Pricing integrations is more straightforward than you might first think, and it's also a part of your overall B2B SaaS integration strategy.

## Integration pricing realities

If you are building integrations for your B2B SaaS product, you probably shouldn't price most integrations separately from your product. If that's surprising, consider these realities:

- ✓ Integrations are product features.
- ✓ Customers expect them with their products.
- ✓ Productization makes it easy to include them.

Since integrations have become expected features and B2B SaaS companies now have the tools and processes to support those expectations, customers generally won't purchase integrations a la carte – that's not how they purchase other product features today.

## When might you need to price integrations separately?

Now that we've provided reasons for not pricing integrations separately from your product let's cover a few scenarios where pricing (or at least charging for) integrations might make sense. Here they are:

- ✓ As a gating factor for plan levels
- ✓ Against a niche competitor
- ✓ When the integration is the product

## What pricing model should you use?

If you find yourself in a situation where pricing your integrations makes sense, we recommend using the value-added pricing model. We could go into crazy detail, but here are a few reasons to use that model:

- ✓ Covers every scenario
- ✓ Drives knowledge of customers
- ✓ Leads to product optimization

## Capture value in exchange for your product

We shouldn't talk about pricing without mentioning fundamental pricing principles. Here they are:

- ✓ Be consistent
- ✓ Make it easy to understand
- ✓ Ensure that it feels fair to customers

In the end, the function of pricing is to capture maximum value in exchange for your product. And the best way to do that, long term, is to make your pricing easy, consistent, and fair.

For more details on pricing your integrations, check out our post on how to price B2B SaaS integrations.

# What's involved in announcing integrations?

In general, SaaS companies want to promote new product features to attract new customers, increase adoption among current customers, and showcase product momentum. Integrations, however, are often overlooked when announcing new product functionality.

But it's best practice to include integrations with the other product features you announce. And it's another important part of your B2B SaaS integration strategy.

## How important is it to announce new integrations?

Most SaaS organizations use a launch tier framework to determine what types of marketing are needed to support different tiers or priorities of product launches.

The priorities might look like this:

- ✓ Priority 1: These integrations are on par with new major features (in terms of marketing efforts). They grab the attention of potential new customers and provide new or uncommon functionality.

- ✓ Priority 2: For an integration that breaks new ground but is expected by your customers. Or an integration that is common in the market and will make you more attractive to new customers.

- ✓ Priority 3: If you are putting out an integration that your competitors already have and need it to ensure that your customers stay around.

But what do these priority levels mean? As a rule, the higher the priority, the more noise your marketing team makes for the product launch.

## Covering the basics of an integration announcement

What is included in an integration announcement? The details will vary based on many things (company, industry, type of announcement, etc.), but the following elements should be present in some form or another:

- ✓ What is being announced: Yes, it's an integration, but what does it do?

- ✓ What this means for your company and product: Does it allow you to function in a new market? Does it position you to do something better, faster, cheaper? Does it improve existing functionality, replace existing functionality, or add new functionality?

- ✓ What this means for your tech partner and its product: This includes similar questions to the prior question but from your tech partner's perspective.

- ✓ What this means for your customers: How does this benefit them, make their work lives easier, and save them time and money?

- ✓ How to get more info: Links or other means of quickly accessing further details.

For more details on announcing integrations, check out our post on how to announce new B2B SaaS integrations.

## How do you know if you'd benefit from an embedded iPaaS?

So far, we've covered several things you should consider as part of your B2B SaaS integration strategy. And we noted that teams can use different approaches to building integrations. However, using an embedded iPaaS can provide your team with a substantial head start when building integrations and a comprehensive framework for handling the integrations through their entire lifecycle.

Having worked with many software teams, we've found a few common indicators for those who would benefit from an embedded iPaaS.

### If you're a startup or a new division or product line in an established company

- ✓ Your prospects and customers frequently ask for more integrations.
- ✓ Your competitors have integrations you don't offer, and you need to catch up.
- ✓ Your developers spend time on integrations instead of your core product.
- ✓ Your developers spend more time than planned on integration infrastructure and tooling to deploy integrations, monitor them, and inspect when something goes wrong.
- ✓ Integration work slows down your overall dev pace and time-to-market for your product.

- ✓ You have a small dev team and lack expertise with integration patterns and best practices.
- ✓ You have concerns about your integrations' reliability, scalability, security, or compliance.
- ✓ Your current strategy will not scale well as you grow.

## If you are a scale-up or enterprise company

- ✓ You've already built many integrations but have an expanding backlog.
- ✓ Missing integrations are causing friction in sales and implementation.
- ✓ You can't innovate on your core product as rapidly as you once did.
- ✓ You've built up integration tech debt that makes internal gains difficult.
- ✓ You have reliability and scalability issues with current integrations.
- ✓ Your dev team spends too much time on integration maintenance and support.
- ✓ You're delivering integrations as services or "bolt-ons" rather than a first-class part of your product.

# Ready for scalable SaaS integrations?

Delivering integrations to your customers can be demanding. Building, deploying, and managing those integrations can be brutal if you don't have a well-developed integration strategy.

We have engineered our embedded iPaaS to provide your SaaS team with the tools necessary to execute your integration strategy efficiently at scale in today's demanding SaaS market.

Schedule a demo to see how Prismatic can supercharge your B2B SaaS integration strategy.

Prismatic