

# HOW SAAS CHANGES AN ISV'S BUSINESS

A GUIDE FOR ISV LEADERS



**DavidChappell**  
& Associates

**Sponsored by Microsoft Corporation**

Copyright © 2012 Chappell & Associates

# Contents

- Understanding the Move to SaaS ..... 3**
  - Assessing SaaS .....3
  - Benefits and Risks for Customers .....4
  - Benefits and Risks for ISVs .....5
  - SaaS or Not? A Simple Decision Tree for ISVs .....6
- How SaaS Changes Your Business ..... 7**
  - Target Customers .....7
  - Pricing and Revenue .....8
    - Pricing Options .....8
    - Revenue Accrual in SaaS Applications .....8
    - Measuring Success: Monthly Recurring Revenue .....10
  - The Sales Process .....10
    - The Customer Buying Decision .....10
    - Sales Commissions .....12
    - Working with Partners .....12
    - The Impact of Average Selling Price .....13
  - Marketing .....15
  - Customer Support .....15
  - Operations .....16
  - Software Development .....17
  - Organizational Structure .....17
- Choosing a Platform for Your SaaS Application ..... 19**
- Final Thoughts ..... 20**
- About the Author ..... 20**

For independent software vendors (ISVs) with an established on-premises business, ***the rise of Software as a Service (SaaS) brings both technology changes and business changes.*** Most often, the technology changes get the lion's share of the attention. After all, transforming an existing on-premises application into reliable, scalable software capable of supporting many customers simultaneously is not a small thing. Challenging as this transformation is, however, the truth is that ***the business changes are harder.*** The move to SaaS goes to the heart of what it means to run a software business, changing pricing, the sale process, and much more.

Despite the magnitude of these changes, there are two main reasons why you might decide to adopt SaaS:

- ***You might be able to grow your business with SaaS.*** Perhaps you can expand your customer base by going after smaller firms, or maybe you can enter an entirely new market, riding the SaaS wave to success in a new area.
- ***You might need to defend your business against SaaS competitors.*** The move to SaaS can let new firms enter your market. One visible example of this is Google Apps, which use the paradigm shift of SaaS to attack Microsoft's established on-premises businesses in email, document creation, and more. Creating your own SaaS offering, as Microsoft has done, might be the only effective way to deal with this new competition.

Whatever your reason, adding SaaS offerings to your product portfolio requires changing how you do business. The goal of this paper is to help you understand these changes and make better decisions about adopting SaaS in your organization.

## Understanding the Move to SaaS

---

Every ISV leader needs to understand the basics of what SaaS brings. You also need to make good decisions about how, when, or even whether your firm should embrace this approach. This section looks at these fundamental aspects of the move to SaaS.

### Assessing SaaS

Like most new things, SaaS brings benefits and risks. ISV leaders need to examine these from two perspectives: the benefits and risks for your customers—why they do and don't buy SaaS applications—and the benefits and risks for your company.

#### ***From ISVs to CSVs***

Traditional ISVs aren't the only ones offering SaaS applications today. It's common for software start-ups to choose this approach, for instance, and some system integrators are also taking advantage of this platform shift to provide cloud-based applications. Even organizations that were traditionally end users of technology are getting into the act, using their own specialized knowledge to create new SaaS offerings. It's useful to think of all of these groups, together with existing on-premises ISVs that embrace SaaS, as making up a broader category of *cloud service vendors (CSVs)*. The rise of the cloud is changing many things about the traditional business of selling packaged software, including the terminology we use and how we categorize the players. Thinking broadly in terms of CSVs rather than traditional ISVs can help you get a better feeling for what's happening.

## Benefits and Risks for Customers

Organizations that buy SaaS applications are diverse, ranging from two-person firms to global enterprises. Yet the things they like and don't like about SaaS tend to be much the same. Figure 1 summarizes the benefits and risks of SaaS from the perspective of your customers.

<b><i>SaaS Benefits for Customers</i></b>	<b><i>SaaS Risks for Customers</i></b>
<b><i>SaaS allows faster deployment</i></b> , because no on-premises installation is required. Rather than wait weeks or months for software to be deployed and configured on its own servers, an organization can directly access the application in the cloud.	<b><i>Using SaaS requires trusting an outside provider</i></b> for the application's availability and security. Most SaaS providers don't let their customers physically examine the details of how they provide security. Instead, those customers must learn to trust the SaaS provider, something that can take time.
<b><i>SaaS typically offers usage-based pricing</i></b> , such as per user per month, letting customers pay only for what they use. Rather than estimating its needs, then buying a license in advance, an organization can pay only for what it actually uses.	<b><i>SaaS can raise legal/regulatory concerns</i></b> with storing data outside the customer's premises. Many countries have laws and regulations about where some kinds of data can be stored and how it must be handled. SaaS applications can run afoul of these, especially when the SaaS application's datacenter and the customer are in different countries.
<b><i>A SaaS application has less financial risk</i></b> than an on-premises application. Most cloud applications offer try-before-you-buy options, so a customer can be sure the application has business value before paying for it. Pay-as-you-go pricing also brings less risk than a typical large up-front license fee.	<b><i>A SaaS application can limit customization</i></b> if many customers share a single multi-tenant application. Multi-tenancy, with multiple customers using the same instance of software, lowers costs, but it also constrains how different that software can appear to those customers. While SaaS vendors often provide configuration options, a licensed on-premises application is typically more customizable.
<b><i>SaaS reduces the need for on-premises resources</i></b> , such as servers and IT staff. Because the software runs in the cloud, customers don't need to invest in the hardware and people to run and manage the application on premises.	<b><i>SaaS applications can be harder to integrate</i></b> with existing on-premises applications. Connecting applications within a single datacenter is often challenging. When some of those applications are running in the cloud, integration can get even harder.
<b><i>SaaS applications offer easier upgrades</i></b> , with no on-premises software to update. Instead, the SaaS provider deploys upgrades to a centrally managed copy. This makes it easier for customers to get the benefits of new features added to an application.	<b><i>SaaS can have lower performance</i></b> than on-premises applications. This issue appears most frequently when network bandwidth between the customer and the SaaS application's datacenter is limited, and so how much it matters commonly depends on geography.

Figure 1: SaaS applications bring both benefits and risks for customers.

As this list suggests, SaaS isn't right for every application or every customer. Some kinds of applications, such as email, are rapidly moving to SaaS—the benefits commonly outweigh the risks. Others, such as those that work with tightly regulated data or require a great deal of customization, are remaining on-premises (and they might stay there forever). SaaS is a great solution for some problems, but it's not always the right choice.

### Benefits and Risks for ISVs

Along with pros and cons for customers, SaaS also brings benefits and risks for ISVs with an established on-premises business. Figure 2 summarizes both.

<b><i>SaaS Benefits for ISVs</i></b>	<b><i>SaaS Risks for ISVs</i></b>
<b><i>A SaaS application can offer the potential to reach new customers</i></b> in broader markets. Taking this approach might let an ISV offer a lower-priced option for smaller firms, for example, or perhaps reach new geographies more effectively.	<b><i>SaaS requires substantial changes in your business</i></b> , with new approaches to pricing, selling, and more. Moving from a traditional on-premises software business to a SaaS business can cause a significant amount of disruption in an established ISV.
<b><i>SaaS lets you sell directly to business decision makers</i></b> without going through the customer's IT department. Since a SaaS application runs in the cloud, it avoids problems with IT organizations that don't support the hardware or software needed to run the application.	<b><i>A software vendor must demonstrate real value</i></b> up front, since customers typically use the software before buying it. Rather than relying on demos, dinners, and discussion, the customer can make a buying decision based on the experiences of real users using the software.
<b><i>A SaaS application can provide more predictable revenue</i></b> than traditional licensing. Pricing models such as per-user per-month subscriptions provide a smoother revenue path than the up-and-down sales of traditional software licenses.	<b><i>Revenue builds up more slowly</i></b> because of typical SaaS pricing models. With traditional on-premises licensing, each organization that buys a copy of the software writes a good-sized check to the ISV. With the subscription model common in the SaaS world, customers pay only a small amount per user per month.
<b><i>SaaS can lower your support costs</i></b> due to shared multi-tenant applications. Rather than supporting a separate software instance running in each customer datacenter, an ISV instead supports a shared copy of the code running in a cloud datacenter.	<b><i>SaaS may lessen your ability to sell customization services</i></b> due to multi-tenant applications. While SaaS doesn't eliminate customization, it can limit how much can be done. ISVs that make a large share of their revenue from customizing their offering can find that moving to SaaS reduces this number.
<b><i>SaaS gives you more knowledge</i></b> about how customers use the application. Because the running application is under the ISV's control, you can see who is and isn't using the application, which customers use which features, and more.	<b><i>SaaS can bring new sales challenges</i></b> . SaaS offerings are commonly sold directly by their creator rather than through intermediaries, which can create channel conflict. SaaS also tends to reduce the amount of up-front revenue available for sharing with partners.

**Figure 2: SaaS applications bring benefits and risks for established on-premises ISVs.**

Just as with customers, an ISV's decision to embrace SaaS should be made on a case-by-case basis. Some markets are likely to be dominated by SaaS in the not-too-distant future, while others will move more slowly. Each ISV needs to evaluate its offerings and the market for each one, then make a well-thought-out decision.

### SaaS or Not? A Simple Decision Tree for ISVs

The risks shown in Figure 2 can make entering the SaaS business look unappealing to ISVs with an established on-premises business. Why invest in a new product with slower revenue growth and the potential for more sales challenges? Making this choice can seem stupid, especially if current customers aren't asking for a SaaS option.

And sometimes it is stupid—SaaS isn't always the right answer. But the choice is in fact more complicated than this. Figure 3 shows a simple decision tree that summarizes the options for an established on-premises ISV.

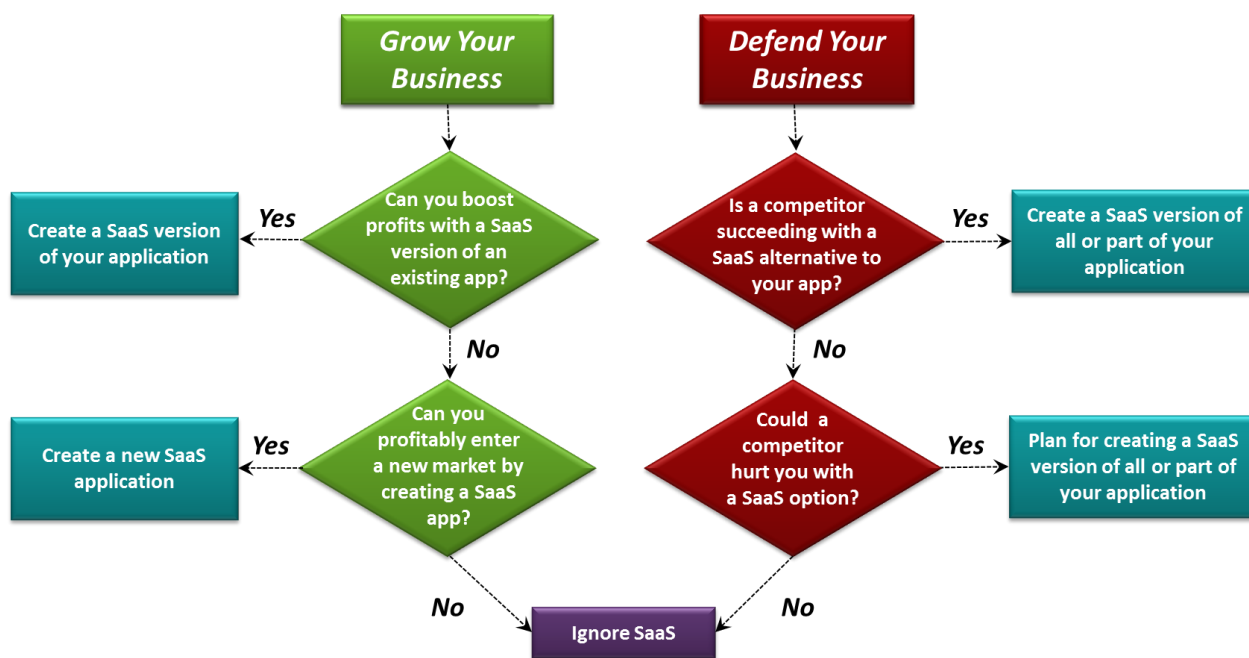


Figure 3: For an established on-premises ISV, the SaaS decision tree has two paths.

The two paths in the tree start from the two options described earlier: *Grow Your Business* and *Defend Your Business*. Put more bluntly, these two paths might instead be named *Greed* and *Fear*, respectively.

On the Grow Your Business path, the first thing to ask yourself is whether you can increase your firm's profits by creating a SaaS version of an existing application. Might you be able to reach new customers, for instance, that are currently too expensive to go after? If you decide that the answer is yes, go ahead and build this SaaS version. If the answer is no—maybe investing in your on-premises offering really is the best path forward in your current market—there's another question to ask: Do you want to enter a new market with a SaaS application? Just as Google used the platform shift of SaaS to enter the wholly new world of desktop productivity applications, might you want to do something similar? Every ISV leader with even a spark of entrepreneurial drive ought to at least consider this option. If the answer is yes, get to work creating this new SaaS application.

But suppose you decide that the answer to both questions on the Greed path is no. Before writing off the idea of SaaS entirely, you also need to answer the questions on the Defend Your Business path, the Fear path. The first of these is obvious: Is a competitor succeeding with a SaaS alternative to your application? If so, you need to respond, just as Microsoft has responded to Google's incursion into its market.

When an established ISV faces this kind of threat, it might not respond by creating a SaaS application with all of the features of its on-premises offering. After all, while the SaaS competitor has some advantages, it probably isn't as full-featured as the established on-premises product. One possible response from an established ISV is to meet the challenge with a SaaS version that offers a limited feature set that's comparable to what the competitor provides. Given the slower revenue growth of SaaS (and perhaps your need to match the lower price a SaaS competitor is offering), it can make sense for an established on-premises ISV to move slowly into this new world.

Alternatively, if many of your customers are still using an older version of your on-premises software, a new SaaS competitor might still be a step up for them even without matching every feature in your current offering. In a situation like this, providing a more advanced SaaS offering immediately might be the best way to avoid losing existing customers to a SaaS competitor. Every situation is different, and established ISVs need to think hard about the right approach.

If an ISV faces no current SaaS competition, there's still one more question its leaders need to ask: Could another firm enter its market with a competitive SaaS option? For most on-premises ISVs, the answer to this question is surely yes—SaaS is always a potential threat. At the very least, every ISV leader should think hard about how his or her firm will respond if this threat materializes: You need to have a SaaS strategy in place. Even though the decision tree in Figure 3 terminates with a box labeled *Ignore SaaS*, the reality is that very few existing ISVs have this option. All of them will likely answer yes to at least one of the four questions posed in this simple diagram.

## How SaaS Changes Your Business

---

It's hard to overstate how much moving to SaaS changes the business of an established on-premises ISV. What appears at first glance to be a technical change—creating shared software that runs in an external datacenter rather than an application for each customer's datacenter—in fact has far-reaching effects on your company. Among the things that can change are target customers, how products are priced, how they're sold, how they're supported, and how the development process works. If these changes are big enough, ISVs might even decide to change their organizational structure. This section looks at these changes, starting with the impact of SaaS on the kind of customers you target.

### Target Customers

You might choose to ***create a SaaS application to reach customers in your current target market more effectively.*** For example, customer relationship management (CRM) applications are a good fit for the cloud. Given the choice, many customers prefer SaaS, and so an ISV selling an on-premises CRM application might add a SaaS version to its offerings. Microsoft once again provides an example here, adding Dynamics CRM Online to its existing on-premises Dynamics CRM offering.

You might also choose to ***create a SaaS application to reach customers in a different target market.*** For instance, because SaaS applications often have a lower initial cost than their on-premises counterparts, an ISV might go after more price-sensitive customers with a SaaS version of an existing on-premises application. Doing this well is

challenging—the cheaper SaaS option might cannibalize the market for your on-premises product—but it can still be worth doing. And since this option might be forced upon you anyway by an upstart SaaS competitor, it probably makes sense to eat your own lunch before a new entrant eats it for you. Alternatively, an established firm might use SaaS to enter an entirely new business. In this case, your target customers are in a wholly different market from the one you normally address. Google Apps provides a clear example of this, with the company using SaaS to attack Microsoft’s established position in email and desktop productivity applications.

Whatever path you take, the key point is that adopting SaaS can let you change the customers you target, and so it’s important to be clear about the market you’re going after.

## Pricing and Revenue

It’s certainly possible to make money with SaaS applications—plenty of firms do it today. But the way those applications are priced is usually different from conventional licensing. This pricing change affects how quickly revenue accrues and how you measure success. All three of these changes are worth looking at.

### Pricing Options

While there can be many variations, pricing for SaaS applications most often breaks down into three options, as shown in Figure 4.

<b>Subscriptions</b>	<b>Per-Unit</b>	<b>Free</b>
<i>Per user per month</i> pricing is the most common subscription model, although options such as <i>per device per year</i> are also possible.	The choices include charging <i>per transaction, per gigabyte of storage, per connection</i> , or based on some other measurable unit of product usage.	Options include <i>freemium</i> , where the base product is free and only premium features carry a charge, and <i>advertising-supported</i> .

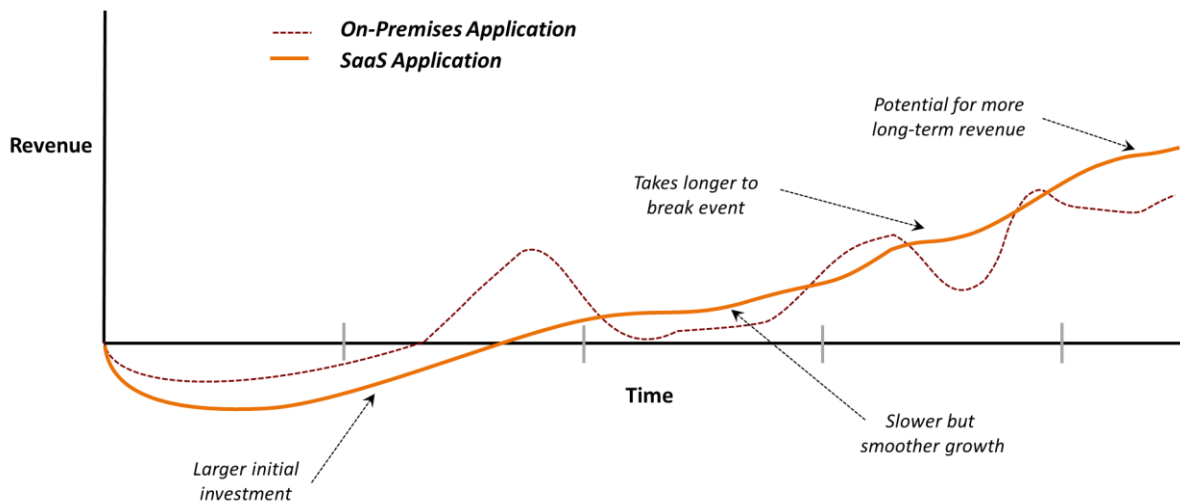
**Figure 4: SaaS applications today most often use one (or more) of three approaches to pricing.**

Different kinds of applications in different markets lend themselves to different choices, and some applications combine two or three of these pricing options. The most popular choice today is some kind of subscription-based pricing, typically per user per month. With this approach, it’s common to give customers a discount for paying in advance, such as offering twelve months of access to a SaaS application for an up-front payment of only eleven months. Users often appreciate this—everybody likes a discount—and it helps lower churn, since the customer is more committed to the offering. Advance payment can also improve the SaaS provider’s cash flow and help it grow revenue more quickly. And given the slow accrual of subscription-based SaaS revenue, SaaS businesses are often hungry for capital. Pre-payment can be an inexpensive way to acquire this capital, much cheaper than bank loans or venture capital funding.

### Revenue Accrual in SaaS Applications

It’s worth looking more closely at the differences in how on-premises and SaaS applications accrue revenue over time. Figure 5 illustrates this comparison in a bit more detail.





**Figure 5: Revenue accrues differently in SaaS applications and on-premises applications.**

As the figure shows, a SaaS application typically requires a larger initial investment than an on-premises application. This is especially true for the first SaaS application an organization creates, since it's learning how to work in this new way. The ISV must also pay to run the application while customers are being acquired, which further increases the initial investment.

Once the application is available, a successful on-premises product can bring in a significant amount of revenue immediately, since customers pay a license fee up front. A SaaS application has slower but smoother revenue growth. In the example shown here, for instance, license revenue for the on-premises application slumps soon after its release—maybe sales drop due to a global financial crisis, for instance—while the subscription revenue of the SaaS application remains constant. Sales of new SaaS subscriptions might stop, but revenue from existing subscriptions continues to come in.

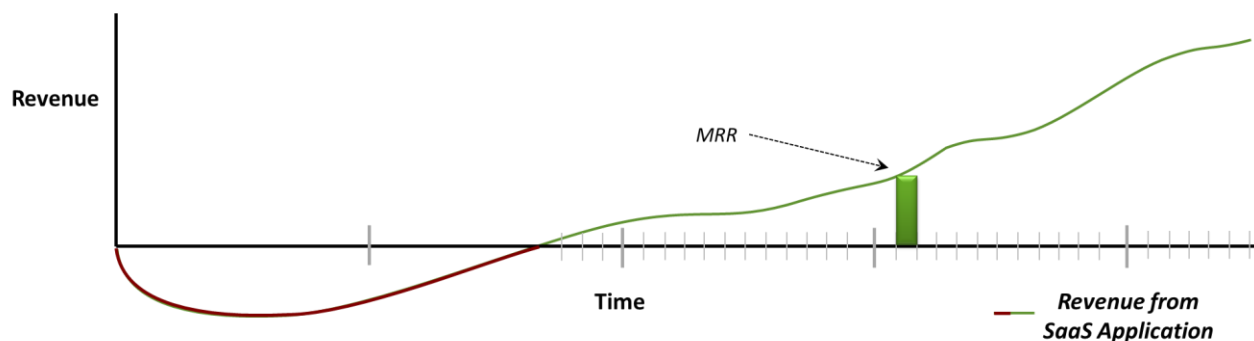
An unavoidable consequence of this slower revenue growth is that the SaaS application also takes longer to break even. Over time, however, it's entirely possible that a SaaS application brings in as much or more total revenue as an on-premises application. The money just comes in more slowly.

### ***Does SaaS Eliminate Customer Lock In?***

Customers often don't like being locked in to a vendor's application. This commitment has real value for ISVs, though, and at first blush, SaaS can appear to eliminate it. Without a large investment in an up-front license, what stops a customer from walking away from the application at any time? The answer is that lock in consists of much more than the initial license fee. Once an organization has been using your application for a while, its people know your application's interface and have probably stored lots of data in your proprietary format. Moving to a competing solution will still incur significant switching costs. While SaaS can reduce the financial aspects of a customer's commitment to your application, it certainly doesn't eliminate lock in entirely.

## Measuring Success: Monthly Recurring Revenue

With traditional on-premises packaged software, it's common to measure success by tracking how many licenses are sold. Since SaaS applications aren't usually priced this way, some other metric is needed. One common choice is to track Monthly Recurring Revenue (MRR). **MRR consists of all recurring revenue in a month, not including one-time fees or services.** For example, MRR for a SaaS application sold as per-user per-month subscriptions would include all subscription revenue for that month. Non-recurring revenue, such as fees for customization, isn't counted. Figure 6 illustrates this idea.



**Figure 6: Monthly recurring revenue (MRR) can be a useful metric for measuring the financial health of a SaaS business.**

The return on a SaaS application is initially negative—you're spending money building it. Once the application is available to customers, tracking MRR provides a useful way to measure success for the thing you care most about: stable, recurring income generated by this application<sup>1</sup>.

## The Sales Process

Not only does SaaS change pricing and revenue accrual, it also changes the sales process. **The differences include when customers make a buying decision, what they know when they make it, and how your firm works with partners.** What follows looks at these changes.

## The Customer Buying Decision

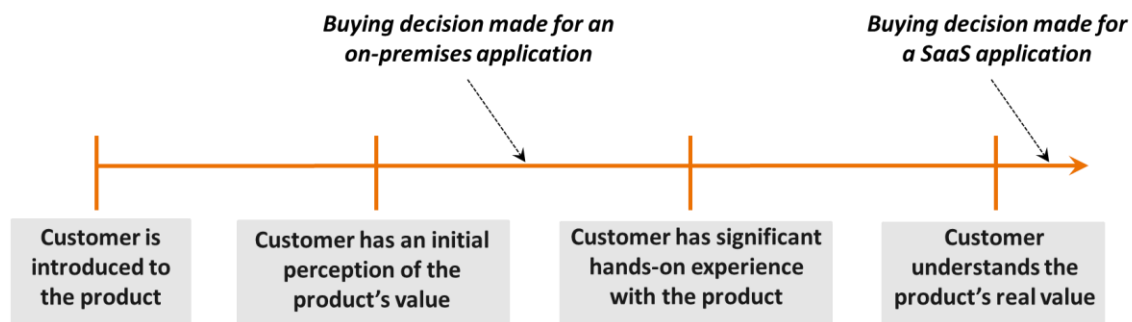
In the traditional process of selling on-premises packaged software to businesses, an ISV's sales team does its best to make a potential buyer understand how well the software solves the customer's problems. While customers get demonstrations of the product's capabilities, they typically don't put the product into production before they buy it. (The challenges of installing and configuring the software often make this impractical.) If the sales process is successful, the customer writes a check, then invests more money in installing the software, configuring it, and training its users. Only then, after a large investment in money and time, does the customer learn whether the software really offers the business value promised during the sales process.

---

<sup>1</sup> For more on financial metrics for SaaS, see [Bessemer's Top 10 Laws of Cloud Computing and SaaS](#), Philippe Botteri, et al., Bessemer Venture Partners

ISVs like this approach. If the product doesn't meet the customer's expectations, it's not good for an ISV's reputation, but the customer's money is still in the bank. Yet how do customers feel about this process? The answer is not so positive. The person making the buying decision takes a potentially large financial (and career) risk based on limited knowledge. He or she can't know whether the product will live up to expectations until it's too late.

Reducing this risk is a primary attraction of SaaS for customers. One way the SaaS approach does this by increasing the amount of knowledge a customer has about the product's real value when a buying decision is made. Figure 7 summarizes the situation.



**Figure 7: Unlike on-premises applications, a customer understands the real value of a SaaS application when the buying decision is made.**

Any sales process starts with the customer's introduction to the product. In the traditional process for on-premises applications, the customer next gets an initial perception of the product's business value, largely based on demos and abstract evaluation. There's usually not much chance to get hands-on experience in real business scenarios before a buying decision is made; perceived value is all the customer has to go on.

Compare that with the process of buying a SaaS application. After getting an initial perception of value, the customer can continue by acquiring significant hands-on experience with the product. Virtually all SaaS vendors provide a try-before-you-buy option, and some offer free use of the complete product for months, providing plenty of time for potential buyers to determine whether the application has business value for them. **The buying decision for a SaaS application is made only after the customer understands the product's real value.**

This typical SaaS sales process often leads to an initial sale of only a few seats. This certainly has drawbacks for the ISV—it's yet another reason why SaaS revenue accrues slowly—but it also has real benefits. A small initial sale implies a relatively low initial price, which minimizes the requirement for drawn-out evaluations of return on investment. Taken together, these things imply that less senior people in an organization can make the decision to purchase a SaaS application. While Figure 7 might suggest that SaaS implies a longer sales cycle than on-premises applications, the lower initial price and simpler ROI justification can help shorten the process. And once the initial sale is made, a SaaS application that has value will inevitably acquire more users within an organization. Sometimes called **land and expand**, this approach to sales has become common among SaaS vendors.

## Sales Commissions

Since SaaS typically changes how an application is priced, **ISVs commonly change how they pay sales commissions.** With on-premises software, a traditional license sale brings in a good-sized chunk of revenue immediately, a significant portion of which can be paid as a sales commission. Selling a subscription to a SaaS application doesn't have the same characteristics. Getting ten users in a new company to buy subscriptions at, say, \$20 per user per month, doesn't generate the same immediate revenue. While this sale might be the foundation for significant long-term revenue—remember the value of land and expand—the ISV's initial income isn't big. How should the people who make this sale be compensated?

There's no one right answer. Here are some options:

- **The seller might be paid a commission immediately based on the total expected revenue for the first year,** i.e., for twelve months of expected MRR. While this can help motivate sellers, it incurs risk—the customer might cancel early—and it also eats up the SaaS business's scarce capital. Giving customers a discount for paying a year in advance can help make this a better option, since the revenue on which the commission is based is received immediately.
- **An ISV might pay monthly commissions based only on the actual subscription revenue it receives.** The seller might then be paid a smaller amount per month for each year that this customer renews. (To keep sellers motivated, an ISV can even commit to paying these renewal commissions if the seller leaves the company.) In many cases, this approach won't generate enough money to motivate a field sales force, so the ISV might need to lower its cost of sales for SaaS applications. It might move to inside sales, for instance, with sellers on the phone rather than meeting customers in person.
- **An ISV might sell only large chunks of product usage, then pay standard commissions.** Pricing can still be usage-based, but with customers only allowed to buy substantial amounts of usage at a time. This approach brings in more up-front revenue—it's not far from buying a traditional license—and so the ISV is able to pay commissions in the usual way. If the same sales force is selling both your traditional on-premises products and a new SaaS offering, however, and if you believe that the SaaS offering is strategically important, you might need to incent the salespeople more for SaaS sales. Without this, many of them are likely to sell what they know rather than learning a new offering.

Typical SaaS pricing also implies another important change in how you (and your sales force) should think about your business. Both sellers and customers have long been conditioned to view the fourth quarter of the ISV's fiscal year as the most important—it's when the sellers need to make their number for the year, and so discounts abound. With a subscription-based SaaS business, however, the fourth quarter is no longer as critical. Instead, **the first quarter becomes the most important for sales.** The reason is simple: A sale in the first quarter brings in revenue all year, while a fourth quarter sale brings in revenue for only a small part of that year. First-quarter sales therefore can contribute much more to annual revenues. Because of this, it's worth thinking about how your commission structure can incent first quarter sales. Unlike on-premises licenses, SaaS business can't rely on a great fourth quarter to save an entire year.

## Working with Partners

Like it or not, **the move to SaaS changes how an ISV works with partners.** These partners, such as distributors, resellers, and system integrators (SIs), play an important role with on-premises software. They might sell your

software to customers, for example, or offer services such as installation, customization, and ongoing management. Like any business, your partners do these things because it benefits them—they make money by working with you.

SaaS changes all of these things. Here’s how:

- With a SaaS application’s typical subscription pricing, there’s significantly less revenue to share, at least initially. Just as SaaS pricing can make it harder to motivate your own sales force, it can also make it harder to keep partners interested in your product. Their incentive to help you shrinks significantly.
- Because a SaaS application runs in the cloud, no installation is required. The need for ongoing management also declines, since most of this work is done by the SaaS provider. A partner that’s built a business around providing these services for your customers will see their revenue decrease.
- While SaaS applications do typically allow some configuration, the extensive modifications that an SI often makes—and gets paid for—might not be possible. Once again, SaaS can shrink what was a good business for the SI.

The bottom line is clear: SaaS decreases the incentives for traditional partners to help you sell, and so **ISVs should expect to bear the majority of the SaaS sales burden themselves**. While you might be able to find new organizations to work with that are set up to survive in the quite different world of SaaS, you should expect changes in how you work with your current partners.

### The Impact of Average Selling Price

The average selling price of a SaaS application determines many other aspects of that SaaS business. Figure 8 summarizes some of the most important.

Average Selling Price	Customer Acquisition Cost	Sales Force	Customer Relationship	Allowable User Training	Example Customer
\$\$\$\$\$	High	External	Personal	Significant	Enterprise
\$\$\$	Moderate	Inside	Telephone	Moderate	Department, SMB
\$	Low	None	Web	Minimal	SMB, Individual

Figure 8: The average selling price of a SaaS offering determines many other aspects of that business.

It's worth looking at each column in this figure separately:

- **Customer acquisition cost:** A SaaS application with a high average selling price can also have a high customer acquisition cost. If each sale brings in a significant amount of money, you can afford to spend a lot to get each new customer. If the application has a medium price, the customer acquisition cost should be moderate as well, while a low-cost SaaS application can't afford to spend much at all on getting each new customer.
- **Sales force:** A high average selling price can support an external sales force, people in the field who visit personally with customers. A medium price typically doesn't allow this; inside sales, e.g., people on the phone, is a more common choice. And for low-priced SaaS products, having any kind of sales force is difficult or impossible—you must sell in other ways.
- **Customer relationship:** A high-priced SaaS product with an external sales force can have personal relationships with its customers. (That's a primary purpose of having the sales force.) A medium-priced offering with inside sales can have telephone relationships, while a low-cost SaaS product typically has no personal relationships at all. To customers, the company is its web site, since that's all they see.
- **Allowable user training:** A high average selling price can support creation and delivery of significant training. This implies that the product can be complex enough to need this training. And since customers spend more to acquire it, they're probably willing to invest more in learning the use the product effectively. A medium-priced offering can support some training, but a low-cost SaaS product needs to be simple. There's no money to support user training.
- **Example customer:** All of the differences described so far lead to differences in typical customers. High-priced products are commonly sold to enterprises, organizations that need the power they bring and probably want a sales person to work with. Medium-priced products commonly target small and medium businesses (SMBs) and departments of larger firms, while low-cost SaaS offerings can target SMBs and individuals.

One way to think about this chart is to realize that SaaS offerings generally need to be consistent across a row. For example, if you're planning to sell your SaaS application at a low price—you're in the bottom row—but also expect to have an external sales force or offer a complex product that requires significant user training, this probably won't work. Similarly, if you plan to offer a high-priced product—you're in the top row—but you plan to sell it solely through a web site, you're also setting yourself up for disappointment. Enterprises are typically reluctant to buy important and expensive software without some kind of personal contact, even if it's just a voice on the phone. While there are certainly exceptions, the rows in the figure can provide a useful starting point for thinking about a new SaaS offering.

Another reality is that SaaS providers might find it challenging to move down in this chart. A product designed for the top row, such as a complex solution that requires significant user training, might not be appropriate for the lower rows, where the resources and user commitment are less. The creators of a top-row solution might also have expected it to be sold via a sales force, a luxury that's too expensive for SaaS offerings in the bottom row.

It's common for ISVs to think initially of SaaS only in terms of the bottom row in the figure, i.e., as being only for low-priced applications. This isn't correct, however. ISVs today offer applications successfully at all three levels. In fact, high-priced SaaS offerings can require less change to your business, since traditional commission structures and even partner relationships can still work—there's more money to share.

## Marketing

If you're in the top row in Figure 8, your marketing probably isn't too different from what you'd do with an on-premises product. With a high-priced SaaS offering, marketing retains its traditional role of generating leads and supporting sales. One obvious difference is that your customers are online—they're looking for a SaaS solution—so doing more online marketing probably makes sense.

If your SaaS offering is in the bottom row of Figure 8, however, marketing looks quite different from the conventional approach. With a low-priced SaaS offering, you have no sales force—your company is your web site. Because of this, your web site becomes both a key part of marketing and your main sales agent; the two come together. ***Especially for applications with a low average selling price, spending time and money on a great website is critically important.***

In this situation, your site must make it possible for potential customers to discover your application and appreciate its value as quickly as possible. The initial goal is to entice each prospect into a free trial of your product, something that should ideally be possible without any human interaction with your firm. While you might want to offer this option, self-service trials are bound to be less expensive than trials that require interacting with a live representative of your firm.

The initial application trial should be as simple as possible, providing a clear and speedy path to value. ***Just as demoing well is essential for on-premises software, providing clear user value quickly is essential to a SaaS application.*** Doing this well can play a big part in converting trials of your product into subscriptions. And of course, every SaaS vendor should monitor its trial conversion rates closely. They're a concrete metric for how well your marketing and sales process is doing.

Helping a potential customer see value quickly in a SaaS application can require some thought. For example, suppose you currently offer an on-premises human resources application that you're moving to SaaS. One easy thing to do is provide a website with information about the product, then a free trial of the complete offering. Yet even though the free trial will help the prospect understand the product's value, it probably requires quite a bit of work to get there. The customer will need to create some kind of database, for instance, probably by typing in a number of records. A better approach might be to offer an online sample of the application that lets the customer do representative work against a pre-populated database. While this takes more work for the ISV to create, it also lets the customer better understand the product's value without going to the trouble of a full trial. Getting to the trial is still important, but a SaaS ISV's website should make it as easy and fast as possible for a customer to understand the product's value. Once again, it's the SaaS analog to demoing well.

Having a great website has no value, however, unless potential customers visit that site. Drawing them there means paying attention to search engine marketing (SEM) using tools such as Google AdWords and Bing adCenter. It also means paying attention to search engine optimization (SEO), doing your best to ensure that your site appears on the first page of organic search results in your area. Focusing on SEM and SEO makes sense for on-premises software products too, but in the online-centric world of SaaS, they're even more important, especially for low-priced SaaS offerings.

## Customer Support

Providing good customer support for a SaaS application has much in common with supporting an on-premises application. Even customers who've found and subscribed to an application completely via the web commonly



want a person to speak with when they have a problem, and so purely web-based support might not be sufficient. More routine aspects of customer interaction, however, such as subscription renewal and billing, might well be done online, especially for lower-priced SaaS applications.

Another critical aspect of customer support is how you communicate with your users when your SaaS application goes down. Virtually every SaaS provider has outages—nobody is perfect—and while customers aren't happy about this, most of them accept it as inevitable. It's become clear, however, that how a cloud provider communicates with its customers during and after an outage is important. Keeping customers informed in real time about the event is valuable, as is being open and honest about the causes and expected time until service is restored. As is so often the case, just thinking about what you want as a customer can be a useful guide to what you should offer your own customers.

## Operations

Until you've experienced it, the far-reaching impacts of moving to SaaS can be hard to see. Even apparently mundane aspects of an ISV's business can be affected. For example, if customers pay for your SaaS offering via monthly subscriptions, you'll need a billing system capable of handling this. And because different customers are likely to want different things—some will want to be billed every month, others every six months—that system will need some flexibility. SaaS pricing models can also affect your accounting, impacting things such as when revenue is recognized.

For an established on-premises ISV, another big change that SaaS brings to your business is the responsibility for running a SaaS application yourself. Unlike the traditional approach, where each of your customers runs your application in its own datacenter, SaaS requires an ISV to run the application in one or more external datacenters. To add to the pressure, the cost of downtime goes way up. When your application crashes while running in a customer's datacenter, it affects only users at that organization. When your SaaS application crashes, it can affect all of your users at all of your customers. ***Being good at running your SaaS applications yourself is deeply important.***

One aspect of this is the platform you choose to run on, a topic addressed later. Another important piece of the puzzle is building a staff with the skills and reliability ethic needed to run an always-on application used by many customers in many locations. For most traditional ISVs, this skill set is quite different from what's currently in their organization; change is required.

Yet another difference is the need to pay the operating costs of the running application. Choosing a platform that lets you pay as you go, incurring more cost only when growth brings in more revenue, can help. Nonetheless, ISVs need to plan for this expense, even for a new SaaS application which hasn't yet acquired enough customers to cover the cost of running it.

This change isn't all bad—you now get to control the environment your software runs in, which means you no longer need to rely on each customer's IT staff to create and maintain a reliable foundation for your application. As discussed earlier, running the software yourself also lets you see what your customers are doing, providing great insight into both their behavior and your application's strengths and weaknesses. Still, any ISV moving to SaaS shouldn't underestimate the importance of getting good at running the applications its customers use. ***You can outsource the mechanics of running a SaaS application to a hoster or cloud provider, but you can't outsource the responsibility for its reliability.***



## Software Development

Creating new software is a fundamental business process for every ISV. As with so much else, adopting SaaS has an impact on this process.

This impact flows from an innate reality of SaaS: Because the application is running in at most a small number of instances under your control, you can update it more easily. Unlike updating a deployed on-premises application, which requires changing installed code at each of your customer sites, updating a SaaS application requires changing only code you run yourself. This shift has a couple of important implications:

- **SaaS applications are typically updated much more frequently than on-premises applications.** Since it's relatively easy to do updates, why not roll out new features to customers as fast as possible? Unlike on-premises applications, where a new version every year or two is typical, SaaS applications can be updated much more often, e.g., every month or two.
- **Classic waterfall development practices tend not to work well.** Given the speed of update that SaaS allows (and that SaaS customers have come to expect), long planning and development cycles aren't appropriate. ISVs that haven't yet adopted agile processes for software development are likely to find that the move to SaaS forces them in this direction. Being able to quickly define, implement, and test new features is an important part of staying competitive with SaaS applications, and agile processes are a more effective way to do this.

Successfully updating SaaS applications typically requires close cooperation between the people responsible for creating new software and those responsible for deploying that software. An increasingly common approach is to merge development and operations teams as part of what's commonly known as *devops*. In a devops environment, the wall between software development and IT operations comes down, with the people in both roles working closely together. Without this tight cooperation, it can be difficult to achieve the rapid creation and deployment of new features that many SaaS markets demand.

## Organizational Structure

As is obvious by now, adopting SaaS changes your business in many ways. In some cases, those changes are so large that an ISV's leadership decides to alter the fundamental organizational structure of their business. As Figure 9 shows, there are three main options.



Figure 9: An established ISV has three main organizational options for embracing SaaS

The options are as follows:

- **1: Do SaaS within your firm's current structure.** This option requires the least change, and so it's where most firms start. Attractive as it is, however, this simple approach can be hard to maintain. SaaS changes how you sell, how you price, how and when revenue arrives, and even how you measure success. Assuming your new SaaS business is running alongside your existing on-premises business, doing both well can be difficult. How do you motivate the same salespeople to sell both SaaS and on-premises products when the on-premises product brings a much better initial commission? How do you fairly compare the success of salespeople asked to do both? How do you manage separate development groups with very different release schedules? All of these things are possible, but they're hard.
- **2: Create a new SaaS group within your firm.** The challenges inherent in option 1 lead some ISVs to adopt this second approach. A new group that's entirely focused on SaaS can have processes and metrics designed expressly for this model, and so it lets ISVs avoid many of the pains of option 1. Still, creating a new group that's selling the cool new thing has its own problems, such as engendering tension between the people in this group and the rest of your organization.
- **3: Spin off a new SaaS-focused entity from your firm.** Cross-group tensions can be managed—after all, they show up with every new advance, not just the move to SaaS—but some ISVs have decided to go all the way to option 3, spinning off a distinct SaaS business. This approach lets you create a wholly new entity organized around SaaS, much like a SaaS startup, which greatly reduces the need to handle tensions between the on-premises business and the SaaS business. This can be an attractive option, but it also requires the most change. Do you really need to replicate the entire business, e.g., human resources and payroll, just to accommodate the SaaS business model? And how can you do this if the SaaS product you're offering is a version of an existing on-premises product?

There's no one best solution. Different organizations take different paths, based on their own unique mix of products, customers, competitors, and goals.

### ***Can Established ISVs Successfully Adopt SaaS?***

Plenty of SaaS start-up CEOs will tell you that the answer is no. They believe that while changing technology is doable, the business changes required for an established on-premises ISV to succeed with SaaS are too great. And after looking at the list of what's required for this new kind of business, leaders of existing on-premises ISVs can be forgiven for feeling a bit daunted. Evolving into a SaaS vendor is not a small task.

Yet being successful in the software business has always required embracing change. The move to SaaS is no different. Yes, some established ISVs won't cross this chasm successfully; some of them will die. But many others will understand what needs to change, then diligently—and successfully—make those changes. Navigating into this new world isn't easy, but it's certainly not impossible.

## Choosing a Platform for Your SaaS Application

Every SaaS application runs on some kind application platform. This platform has a number of requirements:

- It must be reliable, supporting an application that might serve thousands or tens of thousands of simultaneous users in many countries around the clock.
- It must be scalable enough to handle the load generated by these customers.
- It must be as economical to run as possible, since every cent spent running a SaaS application is one less cent of profit for the ISV that owns that application.

The main platform options can be grouped into four categories. The first two, *colo* (short for *colocation*) and *dedicated/managed servers*, are varieties of hosting. The last two, *Infrastructure as a Service (IaaS)* and *Platform as a Service (PaaS)* are varieties of cloud computing. Figure 10 summarizes these choices.

	<b><i>Colo</i></b>	<b><i>Dedicated/ Managed Servers</i></b>	<b><i>Infrastructure as a Service (IaaS)</i></b>	<b><i>Platform as a Service (PaaS)</i></b>
<b><i>Description</i></b>	Space for ISV-provided hardware	Hoster-provided hardware	Virtual machines (VMs) on demand	Managed application platform
<b><i>Pricing model</i></b>	Typically priced by the month; ISV also buys hardware	Typically priced by the month	Based on resources used, e.g., per VM per hour, per gigabyte per day	Based on resources used, e.g., per VM per hour, per gigabyte per day
<b><i>Required management</i></b>	Significant; ISV is responsible for purchasing and managing servers and everything that runs on them	Partial; For dedicated servers, ISV is responsible for all but hardware and basic OS image; for managed servers, ISV is responsible for application and perhaps more	Partial; ISV is responsible for application, starting and stopping VMs, deploying patched OS images, and more. IaaS provider manages all hardware	Minimal; ISV is responsible only for application. PaaS provider manages all hardware, OS images, middleware, and more
<b><i>ISV control over environment</i></b>	Almost total; the ISV owns and manages its own hardware	Significant, especially with dedicated servers	Partial; ISV can choose OS images, install arbitrary software in VMs, etc.	Limited; PaaS provider constrains an ISV's OS image choices and software installation options

Figure 10: ISVs have four main platform options for running their SaaS application.

Other factors to consider in making this decision include:

- **Provider size:** Small hosters are commonly more expensive than very large hosters, and they typically offer fewer services. They might provide more personalized support, however, and ISVs sometimes have existing relationships with local hosting firms.
- **Global distribution:** Many hosters have datacenters in just one country, which can work well if the target market for your SaaS application is just that country. ISVs that wish to target a larger market, whether initially or eventually, should consider a platform with multiple datacenters spread around the world. Without this, customers far from your application's single datacenter are likely to see slower response times.
- **Reliability:** It's hard to overemphasize the need for reliability in a SaaS application. If the application goes down, all of your customers can be affected. Accordingly, choosing a platform provider with replication, perhaps including datacenters in different parts of the world, can be important. At the very least, the provider should have well-tested mechanisms for data replication, ensuring that failures don't lose customer data.
- **Regulatory compliance:** Depending on the market your application targets, running in a datacenter that complies with appropriate laws and regulations can be required. Sometimes, this means that your application must run in the same country as its customers. In other cases, it's enough to run in a datacenter that conforms to those regulations, wherever it might be. Understanding the relevant regulatory and legal issues in your market, then complying with them, is an important part of being successful with SaaS.

As with so many other areas in the evolution to SaaS, there's no one correct answer for everybody. Your responsibility is to understand the options and tradeoffs clearly, then make the best decision for your application and your company.

## Final Thoughts

---

For established ISVs, embracing SaaS is no small thing. While the technical challenges of building a reliable, scalable, multi-tenant application are considerable, the changes SaaS brings to your business are likely to pose even bigger hurdles. Changing people and processes is usually harder than changing software.

But change you must. As the SaaS wave rolls over our industry, more and more ISVs find themselves required to offer this new model, if only to blunt competitors. Even though SaaS isn't right for every company or every application, this approach to providing software is spreading rapidly. If your firm is focused solely on on-premises software today, there's only one choice: Get ready to meet the challenge of SaaS.

## About the Author

---

David Chappell is Principal of Chappell & Associates ([www.davidchappell.com](http://www.davidchappell.com)) in San Francisco, California. Through his speaking, writing, and consulting, he helps people around the world understand, use, and make better decisions about new technologies.