

# SaaS Model Economics 101

## Competitive Advantage in Software-as-a-Service

# SaaS Model Economics 101

## Competitive Advantage in Software-as-a-Service

---

### Table of Contents

Why should you read this? .....	3
The Internet and Sources of Competitive Advantage in SaaS .....	4
Low Cost Competitive Advantage in SaaS.....	5
Internet-enabled Economies-of-Scale .....	5
Switching and Adoption Costs: The Double Edged Sword of Data .....	6
Total Cost of Service: The Mirror Image of Total Cost of Ownership .....	8
Differentiation via the Internet.....	10
Reengineering Business Processes across the Firewall .....	10
Accelerating Network Effects.....	11
Internet-enabled Monetization .....	11
Software Business Models and the SaaS Hybrid Question .....	12
Economic Costs of Software Distribution Models .....	12
Alignment of Business Model with Technology Distribution Model .....	13
Alignment of Customer Needs with Business Model .....	13
Is Your SaaS Business Competitive? Take the Test!.....	15

### Why should you read this?

I'll be honest. This is not light reading. It is not a straightforward recipe like the [Top Ten Dos and Don'ts of SaaS Success](#). But, there is no simple recipe for success in any business. A top ten list can only provide a quick leg up and perhaps help you avoid some common mistakes. Long term success requires a solid grasp of your customers' needs and the value your service offers combined with consistent, good business judgment...and a little luck. If your personal success is tied to the success of a software-as-a-service business or to the success of the SaaS sector as a whole, and the soundness of your own business decisions could make the difference, then I am confident that you will find the return on reading this analysis well worth the investment.

If your business is confronted with issues such as the following:

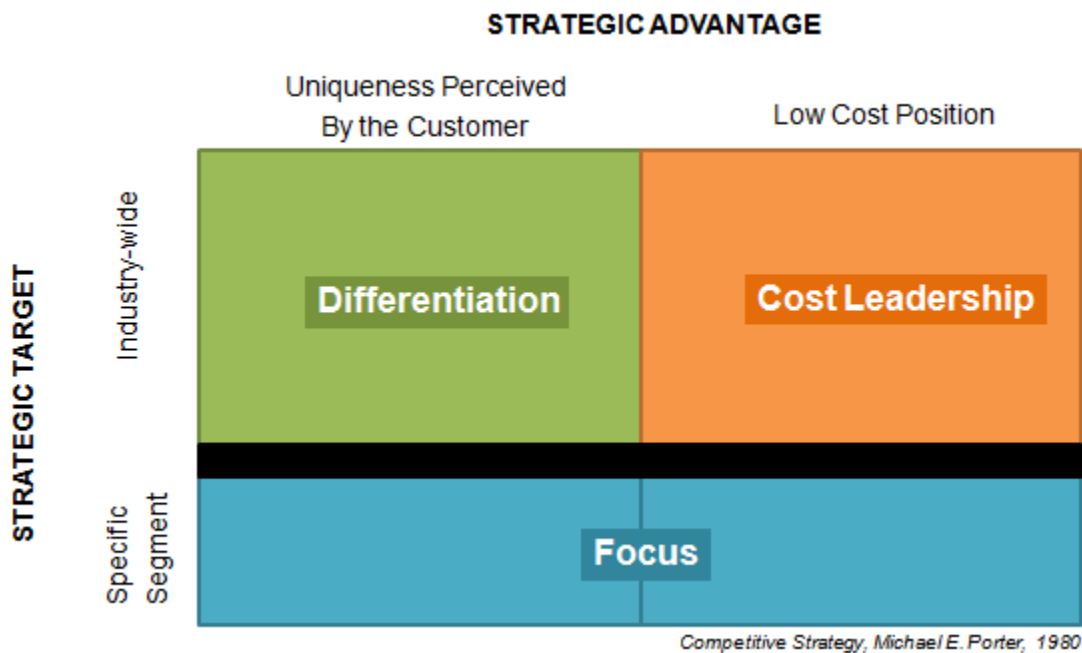
- Why is my SaaS business losing money?
- How do I grow my SaaS business faster?
- How does a SaaS vendor create a long term competitive advantage?
- When does it make sense to take a "hybrid" approach?
- Will SaaS displace on-premise software entirely?
- Are there applications for which the SaaS is simply not the right choice?

Then, there is a good chance you will find the principles laid out in this e-book of significant value in your decision making. To answer these questions accurately, it is essential to have a strong understanding of how SaaS creates economic value over in-house, licensed and home-grown software. If SaaS enables the creation of new economic value, then the model will prevail. If it does not, then on-premise software is an equal or better choice. And, as a SaaS vendor you will have no competitive advantage.

Software-as-a-Service is moving from the cutting edge to the mainstream, but there is enormous disparity in the performance of SaaS businesses from rapidly growing startups to embarrassingly unprofitable IPOs, and only a handful of true success stories like Salesforce.com and Taleo where growth, profitability and customer satisfaction have gone hand-in-hand. Everyone seems to be jumping on the SaaS bandwagon, but at the same time opinions about the future of the sector range from writing it off as an overrated fad to predictions of complete displacement of on premise software. Still others espouse the wisdom of so-called hybrid models. And, there is as much bad advice out there as there is good. The purpose of this e-book is to help SaaS entrepreneurs focus on business fundamentals, make sense of this chaos, steer clear of the hype, and create innovative and wildly successful new SaaS companies.

## The Internet and Sources of Competitive Advantage in SaaS

It is only by creating a sustainable competitive advantage that your business will achieve all three financial objectives of unusual profits, rapid growth and long-term success. However, before digging too far into the details of what tactics will help you achieve these results, it is sensible to make a quick review of the economic fundamentals that lead to competitive advantage. Perhaps the simplest and most lucid model for understanding the economics of competitive advantage was laid out by Michael Porter in his book *Competitive Strategy*—required reading in virtually every business school program. Porter’s model asserts that vendors can achieve competitive advantage by choosing one of two generic business strategies: lowest cost or differentiation, and a variation on both of these themes by focusing on the needs of a specific market segment. In plain English, you can beat the competition if you build a cheaper mouse trap, a better mouse trap, or a cheaper, better gerbil trap.



Virtually everything in the analysis that follows will be based on a simple, overriding premise as to the source of competitive advantage in software-as-a-service:

*The only difference between software and software-as-a-service is that SaaS is delivered over a standards-based network called the Internet. Therefore, all new economic value and competitive advantage must flow from this difference.*

Using the Porter model, SaaS economic value over software comes in two Internet-enabled flavors:

- 1) A lower cost structure from economies-of-scale achieved by aggregating a multitude of customers *via the Internet* onto a single, vertically integrated infrastructure i.e., hardware, software, maintenance, etc. This cost savings is then passed on to the customer through a low subscription price and is generally referred to as the lower total cost of ownership (TCO) of SaaS.

- 2) Reengineering and automating high-value customer processes *by leveraging the Internet*, e.g., online purchase, deployment and support, connecting customers with their customers (B2B2C), etc. and *Internet-enabled* network effects, e.g., crowd-sourcing, support forums, marketplaces, etc.

Both of these sources of value can create sustainable competitive advantage for a SaaS business, 1) is a low cost advantage and 2) is a differentiation advantage. In addition, either can be achieved by overlaying the strategic element of focus on a specific market segment. However, when the nature of your customers, application or technology limits your ability to create value from these two sources, then you have reached the boundaries of the SaaS model. Beyond these limits there will be no competitive advantage over on-premise software.

Some markets have customer's whose needs are so unique and applications that are so complex that they are intractably fragmented and customers cannot be aggregated onto a single, uniform infrastructure. For these customers, SaaS would simply be a foolish choice. Alternatively, a market may be appropriate for software-as-a-service, but the SaaS vendors may not be building products that leverage the potential of the Internet, simply pushing a well-known software application through the browser and calling it a day. In either case, the most likely business result lies somewhere between mediocre success and complete failure. If you read between the lines of the Top Ten Do's and Don'ts of SaaS Success, you will see one or both of these economic truths shining through. The overriding message is this: if you are building a SaaS business, then you are by definition building an Internet business—think and act like one!

## Low Cost Competitive Advantage in SaaS

### Internet-enabled Economies-of-Scale

In my very first blog post on Chaotic Flow, I made the assertion that software-as-a-service is a commodity business. My intention was to make the essence of the SaaS model easy to understand, but also to make it clear that the reality of doing what you need to do to achieve SaaS success is a little difficult to swallow. I mean, who wants to be in a commodity business anyway, especially in software? It's all about innovation and differentiation right?

You can achieve very strong differentiation in SaaS. The next section in this analysis will explore that. But, before you differentiate around the edges of your software-as-a-service offering, you must commoditize it at the core. Otherwise, your cost structure will not support your pricing, and you will not be profitable—at least not for a very very very long time, like many of the of the recent SaaS “success” stories and IPOs. (Disclaimer: having received my economic training from the University of Chicago, I have a strong bias toward the idea that a business should turn a profit, especially public companies.)

I'm not going to go into the technical details of multi-tenant architecture. I believe that this element of SaaS is well understood. What I do want to emphasize here is that the SaaS competitive cost advantage arises from the general principal of aggregating customers onto a single infrastructure to achieve new

economies-of-scale, not the specific technology used to accomplish it. Multi-tenant architecture is simply a means to an end for relational database-driven applications, like Salesforce.com.

The concept of aggregating customers onto a single infrastructure to lower costs extends far beyond the database. It impacts the entire application infrastructure. You are aggregating customers onto a common set of servers, a common user interface, and a common set of business processes. And, it extends beyond the technology to the entire business. You are aggregating customers into a common communication channel, a common purchase process, common pricing and a common support process. You are Wal-mart. Online.

It is in the second half of the cost equation, customer acquisition and support, where most SaaS companies lose their way; or rather find their way to long term unprofitability. If all your customers are identical: identical business needs, identical communication needs, identical purchase process, identical support needs, etc. then you will have no trouble aggregating them onto a common business infrastructure for an enormous cost advantage. But, to the extent that they are different, or simply believe that they are different, then you have your work cut out for you. For example, how much website content do you have to present to get a customer to register for trial? Is it a single, simple message for all customers, or do you need pages and pages that detail your benefits for each industry segment in the specific vernacular of that segment. To maintain your cost advantage you must do your best to streamline all this complexity without losing customers, and ultimately walk away from customers whose needs are so unique that you cannot profitably service them.

Luckily, while this may sound like your own customers are the roadblock to achieving your preordained SaaS cost advantage; they can also provide you with the secret weapon to overcome the roadblocks. It's the one aspect of your single, uniform, vertically integrated infrastructure that can be customized without limit and without eroding your cost advantage: data. Customer data. Unique data. In the Web 2.0 world they call it user generated content. In SaaS, you should think of customer data as the user generated application. Whether you capture it on your website to personalize the purchase process or you capture it in your application to customize security roles, it is the enabler of mass customization and it may allow you to push the economic boundaries of the commodity-based SaaS cost advantage through the complexities of the SMB market all the way out to the idiosyncrasies of the long tail.

### **Switching and Adoption Costs: The Double Edged Sword of Data**

The previous section discussed the costs of software-as-a-service delivery from the SaaS vendor's perspective. But, there are costs associated with choosing SaaS over software that the vendor never sees: the costs of adoption and switching. If you have built your software-as-a-service business well, following all of the Top Ten Do's and Don'ts, then you have made it easy for your customer to find it, buy it and use it online. And, you have a solid, scalable architecture with massive economies of scale. But, your product is unusable without your customer's data. Even if you are strictly a content provider, you must at least get registration and subscription data. Most SaaS business applications require a lot more than that.

Once a prospect is sold on the value of your product, data often becomes the most significant barrier to adoption. Getting data in and getting data out. If the primary users of your application are people,

then getting data in amounts to mouse clicks, typing, spreadsheet uploads, etc. and getting data out (and into the heads of your customers) amounts to learning how to use the application, both individually and organizationally. If other computer systems are significant users of your application, then there will also be an additional adoption cost for integration.

The unusual thing about data is that what starts as a barrier to adoption, over time becomes the cost of switching. As a for-profit business, your goal is to lower the costs of adoption by providing a super easy, yet valuable initial offering, i.e., easy to buy, easy to learn and easy to add data. And, to raise the cost of switching, by making it easy to learn more and more features, add more and more data, and get more and more value over time. This is a process I call application discovery.

Application discovery can be illustrated well by comparing a consumer oriented SaaS, like Amazon.com or Google to enterprise software. There is actually quite a lot you can do at Amazon.com other than buy a book. You can create wish lists, write reviews, manage your one-click account, you can sell books, or music, or videos, and you can have your books printed on-demand. But, you don't get it in your face all at once. You discover it as you use the service. In contrast, enterprise software is notoriously difficult to use and integrate. Why? Because, it is purchased up front. The license cost is sunk and the audience is captive. So, strap the users to their chairs and make them sit through two weeks of training. Kick off that middleware project. Get the data in. Get the data out. Now we can really start realizing that ROI the sales guy promised us.

The enterprise model of adoption is still available to SaaS vendors. There is nothing about the Internet that takes this option away. And, some applications may be so inherently complex that it is required. But, this labor intensive approach provides no competitive advantage. In truth, it is possible to build application discovery into enterprise software, however, license pricing and revenue from high-touch services remove any incentive to do it. SaaS on the other hand is sold on-demand over the Web, and if done well has data-driven mass customization built in. Managing the process of application discovery is simply another example of mass customization, except customer needs vary over time instead of varying over market segments. By fully automating initial adoption and enabling easy application discovery, *adoption costs can be reduced to the lowest possible point and switching costs increased naturally over time* as the customer discovers and invests in more advanced capabilities of the product—giving the SaaS vendor another powerful competitive advantage.

## Total Cost of Service: The Mirror Image of Total Cost of Ownership

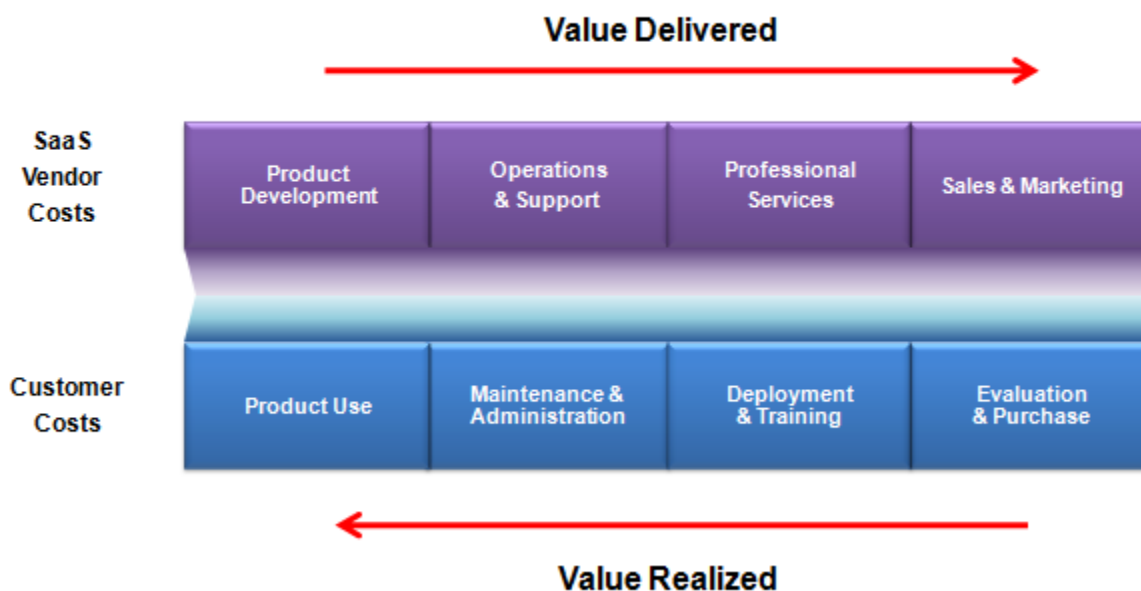
Much has been written about how software-as-a-service can lower a customer's total cost of ownership (TCO) by eliminating the high up-front investment and ongoing maintenance costs of on-premise software and hardware infrastructure. Most SaaS vendors deliver on this promise with a bargain basement subscription price, putting their faith in the miracle of multi-tenant architecture, and working to build enough volume to make the theory a reality. Meanwhile, they bleed cash.

Multi-tenant architecture simply is not enough. In fact, it can distract you from the real economic challenge of achieving economies-of-scale and *driving out costs across the entire value chain*. You've lowered TCO for your customer, now it's time to think about lowering Total Cost of Service (TCS) for your business. TCS is the total cost of delivering your service to a customer, and if you expect to run a profitable, cash-positive business it can only happen if your lifetime customer value exceeds TCS.

$$\text{Life Time Customer Value} > \text{Total Cost of Service}$$

Total cost of service is the mirror image of total cost of ownership. If you think of the value that your customer realizes from your product as resulting from the sum of all the work that you do (TCS) and all the work that your customer does (TCO) from raw idea through product delivery to realized benefit, then it becomes clear that creating a disruptive technology is really about taking costs out of the value chain, regardless of which side of the fence they sit on, because you pass your cost savings on to your customer in the form of lower prices.

## Software Delivery Value Chain



*The Mirror Image of Total Cost of Ownership : Total Cost of Service*

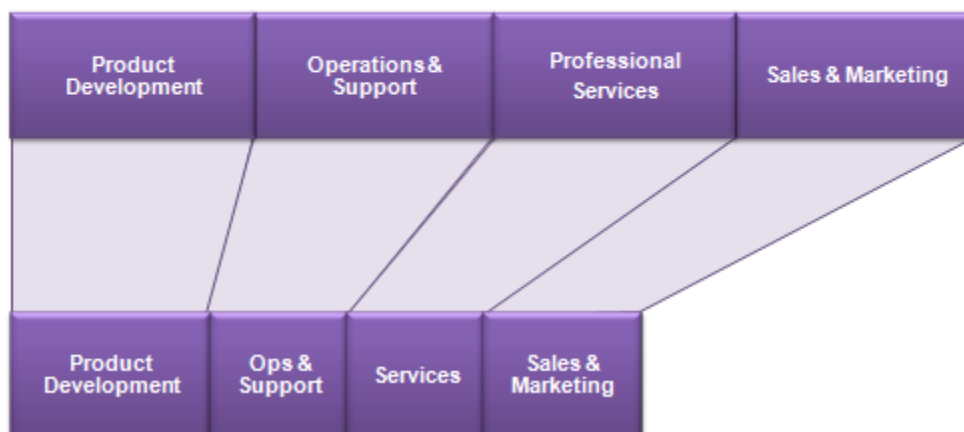
SaaS vendors should look to drive costs out of the entire value chain and out of their operations across every function, not just through multi-tenant architecture. The same business concepts and techniques

used to deliver your core product should be applied to every business activity to drive out unnecessary service delivery costs:

- Standardization
- Mass Customization
- Process Automation (internal and external leveraging the Internet)
- Customer Self-service

More often than not, the results are not only a reduction in cost that can be passed on to customers in the form of lower prices, but also an increase in customer service speed, performance and quality due to streamlined, integrated businesses processes that seamlessly connect you to your customer through your online presence.

## Total Cost of Service



*SaaS vendors should lower TCS by driving costs out across the value chain*

Standardization and mass customization are essential to a true multi-tenant SaaS product, but these concepts can be applied to achieve economies-of-scale across the entire value chain. For example, mass customization can be applied to customer acquisition through website personalization and automated nurturing programs that serve up content to prospects based on their history with your company, e.g., if the prospect spent 5 minutes on the widget page, then send that prospect an invite to watch a video on your widget capabilities. Or, apply Internet-based automation to your external support processes by offering a portal, knowledgebase, forums and online chat.

Customer self-service may sound like bad business, but the truth is that sometimes your customers would simply rather do things themselves, particularly if it means saving money. When done well it not only drives out costs through automation, but also increases convenience, speed and quality. When was the last time you paid for full service at the gas station? I'm probably dating myself, because I honestly can't remember the last time I pulled into a gas station that even offered full service. It used to be the standard, but it was eliminated by disruptive technologies for pumping and payment that eliminated costs within the value chain. Today, most people are perfectly happy to pump their own gas.

## Differentiation via the Internet

Competing through differentiation is the essence of the software business. Feature wars, solution selling, performance testing, roundup reviews—its all about proving superiority, so you can win the deal and charge a premium. One of the hardest lessons to learn in SaaS is that differentiation usually comes second to cost efficiency. But, if you have put your cost structure in order with solid economies-of-scale, then differentiation can be the ultimate competitive advantage. If you have not, then your software competitors will have price parity, and you are unlikely to out-differentiate the masters.

That said, if low cost advantage is the cake of SaaS, then Internet-based differentiation is the icing. All the coolest stuff that happens online, e.g., search, forums, social networks, media sharing, viral marketing, micro-financing, syndication, crowd-sourcing, etc. follows systematically from the inherent nature of the Internet, or in economic terms network automation and network effects. Ironically, enterprise software companies have been some of the slowest adopters of the Internet as an open network. In the Web 1.0 wave, B2B software firms just scratched the surface of the Internet's potential to create competitive advantage. They put up a marketing website, set up a support email and maybe a knowledgebase, but not much more. B2C software has and still does lead the way on the Internet. Unfortunately, this near-sightedness has heavily influenced SaaS when viewed as an outgrowth of enterprise software. SaaS is simply a dumbed-down enterprise application delivered through a browser with lower TCO, so SMBs can afford it...right? Wrong! This perspective all but abdicates the core advantage of being online and the natural birthright of software-as-a-service. Don't just deliver your application over the Internet—become part of the Internet.

## Reengineering Business Processes across the Firewall

Business process automation across the firewall has paled in comparison to internal enterprise process automation. When your SaaS application reaches out to customers, partners and vendors across the Internet, and then goes further to help your customers reach out to their customers, partners and vendors you begin to unleash the potential competitive advantage of network automation and network effects. Any in-house inventory management system can eliminate cycle counts, but only the Internet allows you to look ahead into customer inventory and automate replenishment from suppliers. Any old enterprise helpdesk will allow you to take a phone call and track tickets, but only the Internet allows you to reach out directly to customers and integrate the experience across phone, email, forums, chat, etc. and have the products themselves report back through remote monitoring.

One of the biggest challenges of many SaaS businesses is the high cost of customer acquisition. They drink the Kool-Aid of multi-tenant architecture, and expect it to be a panacea of cost reduction. They offer their service at bargain basement prices. Then, they wake up to realize that their sales, marketing and support efforts don't look any different from the average enterprise software company. High acquisition costs are killing the bottom line. The solution is network automation. Where are your customers? Online. How can you get them to find, try and buy your product cheaply? Reengineer and automate customer-centric business processes across the Internet. Master online search and social marketing. Make videos, not brochures. Don't just educate, entertain. Enable online trial. Crowd-source support. Crowd-source product development. Wrap it up seamlessly into a single online experience and

transform your SaaS application into an Internet business. Most importantly, do things your SaaS competitor hasn't figured out yet, or for that matter do things no one has ever done before.

## Accelerating Network Effects

Network effects are the benefits your customers derive simply because you have lots of customers. They are the difference between an empty singles bar and a packed nightclub. They have always existed in software. For example everyone uses Microsoft Word, because everyone uses Microsoft Word which makes it easy to exchange documents. But, network effects on the Internet are different. They are easier to create. And, they can spread much faster because you can enable them through network automation. In the Top Ten Do's and Don'ts of SaaS Success, I assert that the smart software-as-a-service entrepreneur will not build the business central value proposition around network effects. That is a house of cards. You have to get that first customer in the door based on the value you deliver to that single customer. But, if your customers can benefit by interacting with each other, and you facilitate the interaction through your product, then you can often take a \$50M company to \$500M and beyond. Network effects not only expand your value exponentially, they also create lock-in through incredibly high switching costs (What would it cost to switch from MS Word to another word processor today?).

## Internet-enabled Monetization

A unique aspect of the Internet is that it not only allows you to create new value through the network, but it also allows you to capture it. For example, you can't price a print ad by the number of people who read it, but you can price an Internet ad by the number of people who click on it. The ability to track usage opens up new pricing models. Whereas software companies charge by the copy or user, SaaS companies have access to usage data across the entire network. When combined with network effects, this capability creates entirely new business models. SaaS companies have the potential to monetize the value created by their network in addition to the value created by their application.

Here is a short list of new monetization opportunities open to software-as-a-service businesses.

### Network-enabled services

- Advertising
- Syndication (content/applications/data)
- Benchmarking and market intelligence
- Integration
- Cloud services
- Marketplaces

### Revenue models beyond subscriptions

- Referral fees
- Transaction fees
- Consumption-based pricing
- Performance-based pricing
- Reseller margin
- Revenue sharing

Does any of this sound familiar? It should. Many of the ideas expressed here were touted during the Web 1.0 era and are no invention of mine. But, it is all about timing. While everyone was worried about when their grandmother would be willing to provide a credit card online, no one realized that the businesses and software companies that were so anxious to reach out to consumers over the Internet would be the slowest to reach out to each other. SaaS vendors have the opportunity to realize this vision. But, do they have the creativity? Those that do will develop enormous competitive advantage over their rivals.

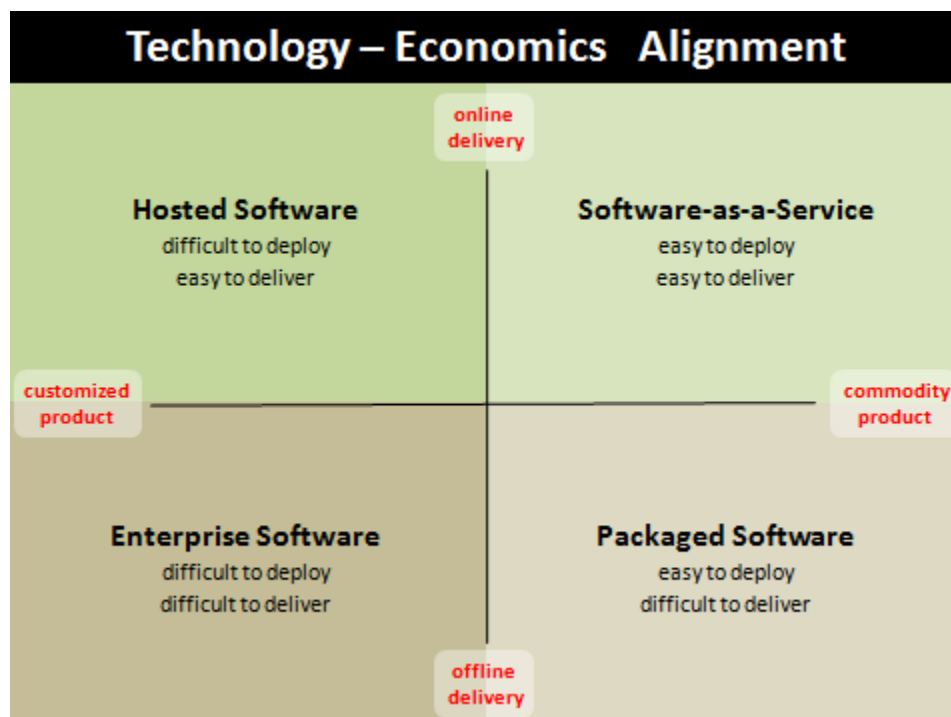
### Software Business Models and the SaaS Hybrid Question

Since its inception, software-as-a-service has labored under an identity crisis. What truly distinguishes SaaS from software? How should it be priced, sold and serviced? Is it possible to succeed with a hybrid approach where a vendor offers both software and SaaS versions of a product?

First and foremost I would like to say that if you are managing a startup, it doesn't matter if you are offering enterprise software, B2B software-as-a-service, online games, or even hardware—*stick to one business model!* Handling multiple business models almost always entails increased organizational complexity and heightened internal politics. It is often essential to separate organizational functions (e.g., sales, marketing, engineering, etc.) or even entire P&Ls (e.g., divisions, spin-offs, etc.) in order to achieve the right cost structure and culture required to be successful in each line of business. In short, it is death to a severely resource constrained company.

#### Economic Costs of Software Distribution Models

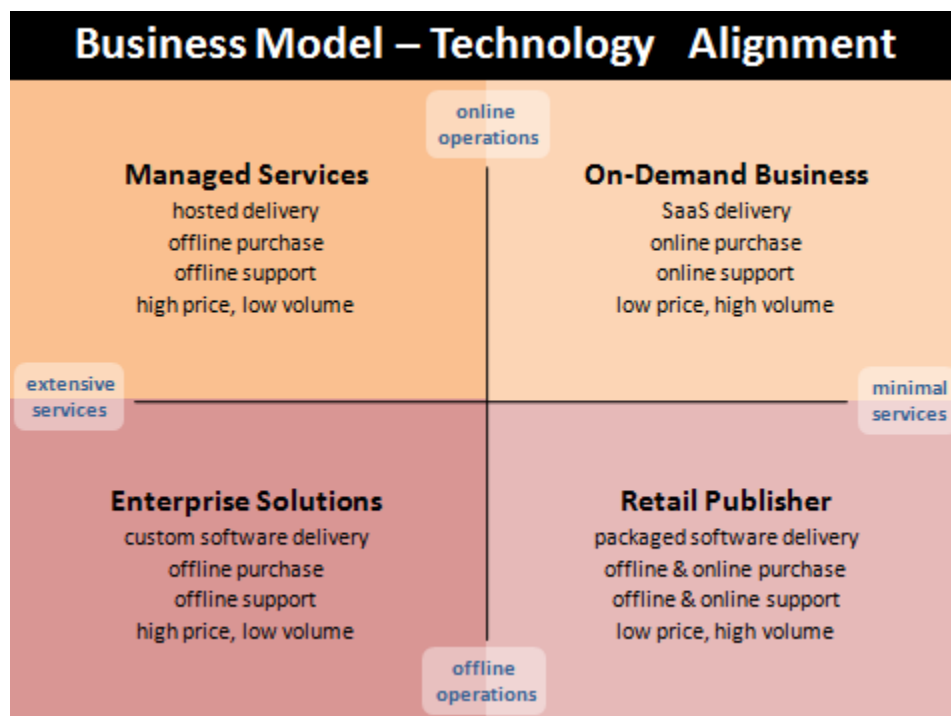
The most common software technology distribution models arise naturally from the economics of information goods. Computer software does three basic things: copies data, transforms data, and moves data. But, the unique thing about software (unlike hardware) is that it just happens to be made up of data itself. Why is this important? Because the costs of copying, transforming and moving data around is decreasing every day, and in many circumstances it is economically equivalent to zero. This is why so many Internet applications are free! The more a software application lends itself to this sort of self-referential automation, the lower its distribution cost. Pretty theoretical, so here it is in plain English. Custom applications are hard to deploy through automation (copy and transform) and fat or data-intensive applications are hard to deliver over the Internet (move).



How costs give rise to different software distribution models.

### Alignment of Business Model with Technology Distribution Model

The more you can align your business model with the underlying economics of the technology, the greater value created and the stronger your competitive position. Perfect alignment is rarely achievable, because your customers will pull your business in one direction, while your technology will pull it in another. For example, your application may be very lightweight and easy to delivery as SaaS. But, your customers may be used to getting their special needs met through resellers, and may not be used to buying online or willing to give up their bells and whistles for lower TCO, even if their special requirements deliver very little economic value in the end. So, choose your customers and your technologies wisely! The picture below depicts how the most common software business models line up with the most common software delivery technologies above.



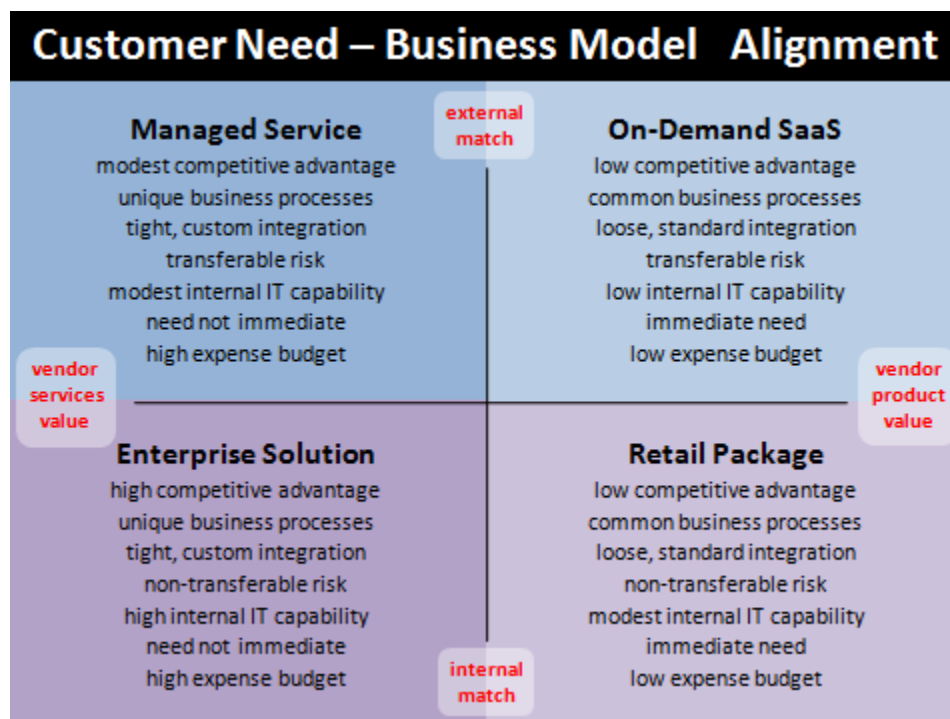
*Aligning the business with the distribution model increases competitive advantage.*

Note that business model encompasses not only the technology of application delivery, but also operational characteristics such as communication, purchasing and support. In short, the more SaaS product delivery can be leveraged into a complete on-demand business, where operations are naturally automated and delivered over the Internet through an online presence that is seamlessly integrated with the application itself, the stronger the alignment of technology distribution (SaaS) with the business model (on-demand), and the greater the potential for long term competitive advantage.

### Alignment of Customer Needs with Business Model

Now let's say you've done everything right. You've built it, but will customers come? Ultimately, it is the alignment between your total service offering and your customers' needs that will determine your business success. Even if you follow all the Top Ten Dos and Don'ts of SaaS Success, but you offer an on-demand product that no one wants to buy or use over the Internet, then you still have a clear recipe for

failure. Alternatively, selecting the approach that best fits your customers' needs, but then failing to execute on critical business and technical dimensions of your chosen model can lead to an inferior competitive position and poor profitability. Success depends on first choosing the right customers, then choosing the right business model for those customers and finally achieving coherent alignment between your customers, your business and your technology. Confusion usually arises when you start with the technology, but it doesn't align coherently with your chosen business approach or your chosen customers, i.e., we have this cool new technology, let's find someone who needs it, and then see if we can get them to pay for it. It's OK to start with the technology first, that is the nature of innovation, but the customer need and business model economics should be vetted prior to writing a single line of code.



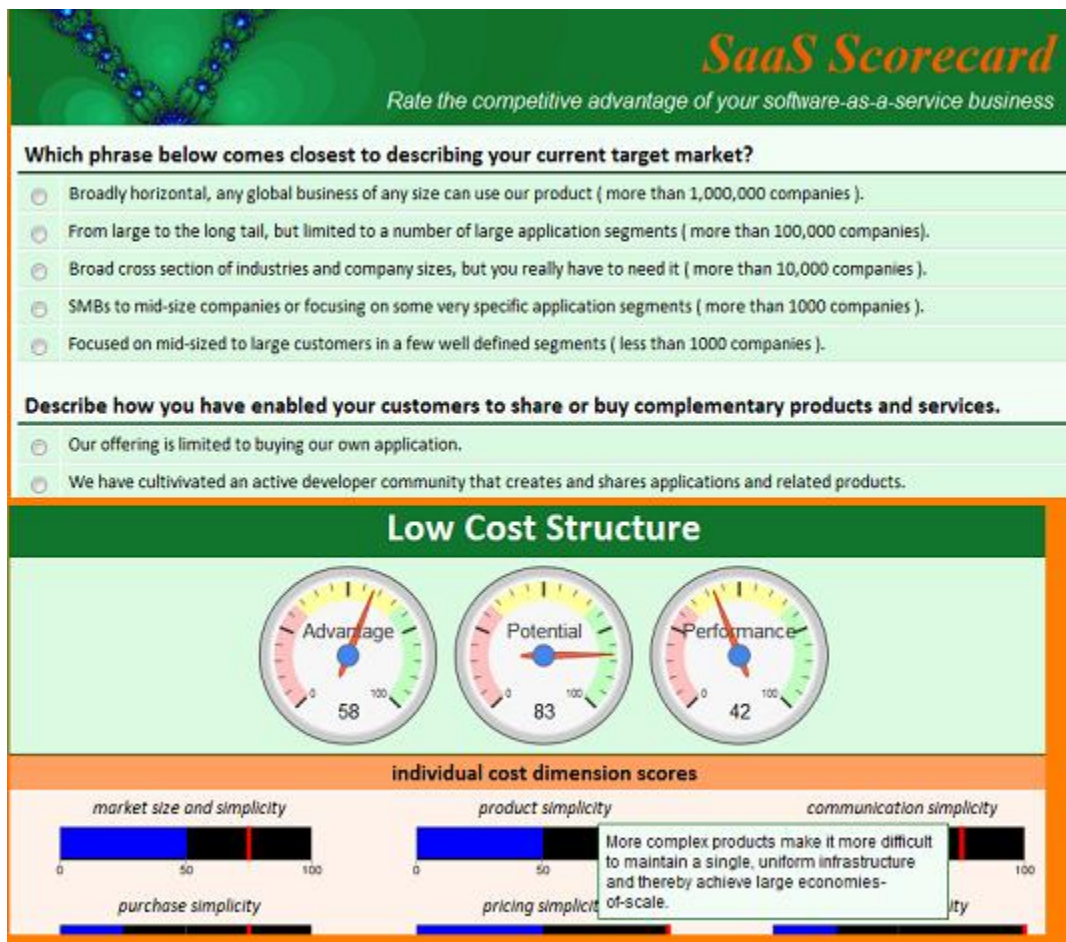
*Customer needs determine the natural business and technology distribution models.*

A comparison of the horizontal dimensions of the previous charts implies that customers that have custom requirements (e.g., unique processes, tight integration, etc.) will value the vendor's services as much or more than the application itself. These applications may be delivered as SaaS, but the business model will not be on-demand, and therefore the business will not achieve as strong of a competitive advantage over on-premise alternatives. The primary source of competitive advantage under a managed services model is the vendor's expertise and capabilities; the method of software delivery is an afterthought.

Along the vertical dimension, customers that have a strong preference to keep an application internal, (e.g., it provides competitive advantage, entails high risk, and is easily managed by IT staff) will be reluctant to use SaaS. Applications characterized by high ancillary services and high resistance to outsourcing fail on both dimensions and are simply poor candidates for SaaS. However, the bar is always moving. As the costs of data replication, transformation and distribution continue to decrease, the economic pressure to move applications online will continue to increase.

## Is Your SaaS Business Competitive? Take the Test!

Using the principles of SaaS Model Economics 101 and the Top Ten Dos and Don'ts of SaaS Success, this online self-test located at <http://saas-model.chaotic-flow.com> is designed to help software-as-a-service entrepreneurs and investors evaluate the competitiveness of their businesses relative to licensed software and other SaaS competitors. Every attempt has been made to create a test that accurately reflects these economic principles; however, the goal is simply to provide useful feedback. That is, it's just a game. Have fun!



Go to <http://saas-model.chaotic-flow.com> to test the competitiveness of your SaaS business.

There are 40 questions, so expect 10-20 minutes depending on your speed and seriousness. Answer each question as objectively as possible to ensure the best possible results. Scores are calculated across 20 key business dimensions that impact low cost advantage, differentiation, adoption costs, switching costs and network effects. For each dimension, a score is calculated for the potential competitive advantage that can be achieved, the performance of the SaaS vendor in achieving it and the combination of potential and performance resulting in the actual competitive advantage that is realized.

## About

---

This e-book is based in part on recent blog posts concerning software-as-a-service business strategy at Chaotic Flow ([www.chaotic-flow.com](http://www.chaotic-flow.com)) by Joel York ([www.crunchbase.com/person/joel-york](http://www.crunchbase.com/person/joel-york)).

Currently Chief Marketing Officer for Xignite ([www.xignite.com](http://www.xignite.com)), a pioneer in financial cloud services, Joel York has more than 15 years of experience bringing software and software-as-a-service (SaaS) products to market. He has managed global marketing and sales organizations serving over 50 countries, including local operations in the US, UK, Germany, Israel and India. His experience ranges from \$5 SaaS subscriptions to \$5M enterprise deals and he shares his expertise on SaaS, cloud computing and Internet strategy on his blog, Chaotic Flow.

Prior to Xignite, Joel provided SaaS strategy and marketing consulting services at affinitos, where he worked with rapidly growing Web 2.0 clients such as Conduit and Zendesk. He has held senior management positions leading the marketing and sales organizations of both startups and midsize public companies, including Navis, eMind, Passlogix and SPSS. He began his software career at Deloitte Consulting.

Joel holds a BS in Physics from Caltech, an MS in Engineering Physics from Cornell University, and an MBA from the University of Chicago.

Contact Information:

[joelyork@chaotic-flow.com](mailto:joelyork@chaotic-flow.com)

[www.linkedin.com/in/joelyork](http://www.linkedin.com/in/joelyork)