Software Runs the World, But At What Price?

Purpose: The Defense Innovation Board (DIB) released a Software Acquisition and Practices (SWAP) study to congress, as requested by the National Defense Authorization Act (NDAA) FY18. The word "Price" does not appear anywhere in the 63 page study; although the word "cost" appears several times. In accordance with (IAW) FAR Part 2.101, many defense programs have determined that software licenses, cloud services and software development are categorized as commercial. Commercial determinations make the acquisition exempt from certified cost & pricing data, making 'price analysis' opposed to 'cost analysis' the paramount form of analysis in the proposal evaluation. Examples include the Washington Headquarter Services Joint Enterprise Defense Infrastructure (JEDI) contract, USAF's Kessel Run software development and NAVSEA's SaaS contract for model based product software.

With the release of the Adaptive Acquisition Framework (AAF) software path, it is a reasonable assumption that defense technology will rely on advancing software and technology. The ability to understand the value & price and price of software will be crucial. This paper can serve as an informational list of elements, creative approaches, and considerations to include when evaluating price in software procurements.

Disclaimer: This document does not constitute Agency, Department, or U.S. government policy, instruction or regulation. This document will not bind a Contracting Officer or Agreement Officer (KO/AO) into a price determination.

Collaboration: This document is a collaboration of practitioners, and feedback is continually welcomed using the update form here.

DAU webcast on this topic included Air Force, DCMA, JAIC, and US Digital Service information here: https://www.dau.edu/events/DAU-Pricing-Forum-Series-Feb-21

This document is divided into five sections:

1. Introspective
2. Software Procurements & Price Influencers
3. Price Considerations – How to Evaluate
4. To be a Great Buyer, "Be the Seller"
5. Alternative Resources – DoD & external resources

Introspective:

To determine a fair price when buying software, a company should first perform an introspective look. It is a reasonable goal going into any market that the buyer has at least a general idea of the company's wants and needs. This is useful in narrowing down the selections to prevent extra spending on redundant features. Answering the following questions will help apply a filter for budget and functionality:

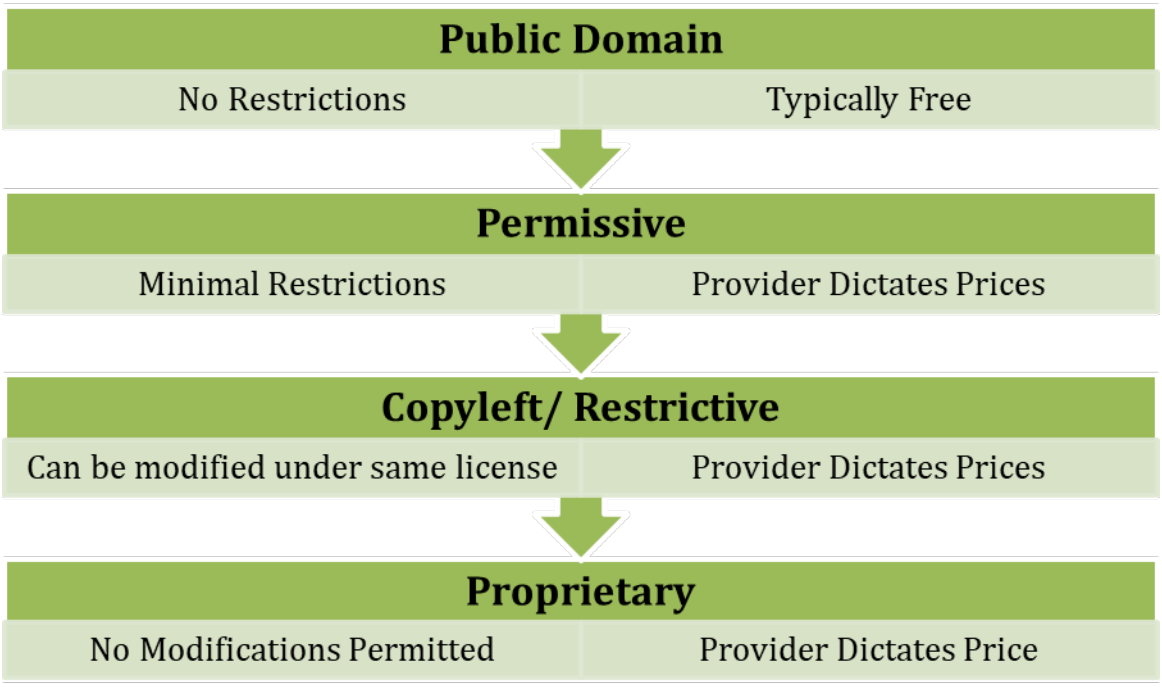
- What needs are not being fulfilled by the current solution?
- What is the goal for adding a new solution?
- What are stakeholders, such as investors, shareholders, and project leaders, willing to invest for a new solution?
- Is the company willing to spend extra work hours towards training a new solution?

Software Procurements & Price Influencers

The term "Software" is broad; therefore, this section outlines types of software procurements and the different types of strategic decisions, intellectual property, and service requested which can all affect price.

Licenses

Pricing a software license depends on the kind of license it is. Two primary factors can be considered here, how restrictive it is and who maintains it after purchasing. For the former factor, the product will fall somewhere in the four following levels of restriction:



For the latter factor, maintenance of the software falls into two categories:

Perpetual License

- Paid up-front in one, big lump sum
- Customer take full custody of software
- May have additional yearly maintenance fee typical at 15-30%

Subscription

- Priced on a yearly or monthly basis and is an on-going subscription
- Provider maintains custody and software updates

In many license examples which restrict data, the provider dictates price. There are many models for licensing, but assuming true market conditions exist, the market should dictate price.

Cloud

Cloud computing providers may charge a set amount for a certain period of performance (e.g., on a per year basis or hourly basis). Cloud Services can also charge based on customer usage, in addition to the time-related fees. DoD's Section 809 panel recommended a consumption-based model. Currently in the Government, a certain number of cloud credits are purchased and any extra are wasted. Also, it is important to understand that the price per number of cloud credits factors in the price for customer service & support. The table below captures four common types of cloud computing services:

Infrastructure as a Service (IaaS)

- Externally Managed Servers and Data Infrastructures
- Pay-as-you-go

Platform as a Service (PaaS)

- Co-Managed; Users maintain Own Applications
- Pay-as-you-go

Software as a Service (SaaS)

- Users Gain Secure Access to Provider's Applications
- Price Determined by Number of Features and Users

Serverless Computing

- Hands-off approach for Users
- Providers Handle All Data and Infrastructure Needs

With a cloud, price is typically dictated by responsibility and risk. The pizza as a service model (next page) is a great example outlining how responsibility shifts (affects price). Managing services yourself will result in a lower contract price; whereas as you migrate toward an IaaS, PaaS or SaaS structure, you shift more responsibility to the vendor, increasing contract price. Although the contract price may increase or decrease based on the level of responsibility given to the vendor, consideration should also be given to the internal costs to manage the service.

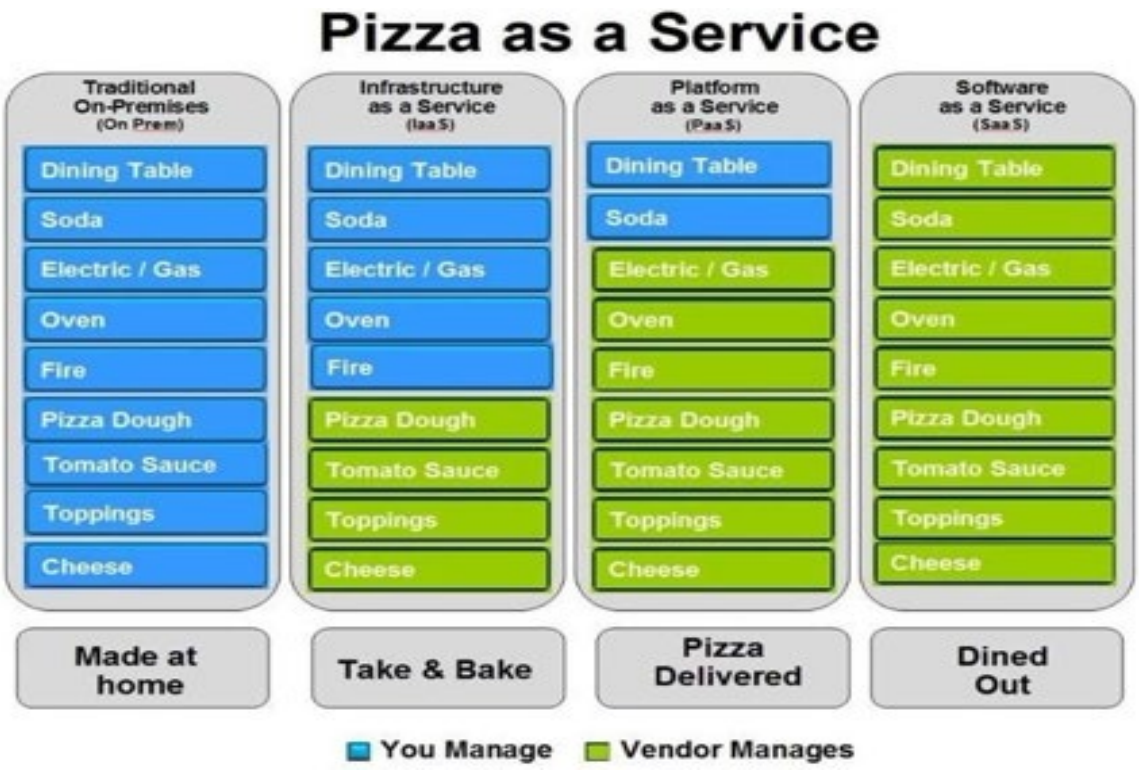


Chart was designed by Albert Barron, Executive Software Client Architect, IBM Software.

Software Development (Buying Labor)

Software Development within the DoD has trended towards the DevSecOps model. Developing software begins with a formally trained software engineer (e.g. labor). Therefore, pricing software development can be thought of like budgeting for any other employee project. In other words, buying labor is evaluating cost data without certified cost data. You can use your technical resources to help estimate the burdened labor rates & hours involved with development. The DevSecOps structure (below) is a continuous loop of labor expenses throughout the lifecycle.

Due to the risk posture, FFP is generally more expensive (as the contractor is taking the cost-risk) whereas T&M proposed price is generally less expensive (as the buyer is taking on the cost-risk up to the ceiling). See Section "Estimating for Labor - An Interactive Approach"
Funding the development typically takes two different contracting approaches:

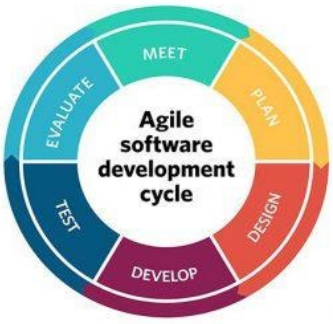
1. Firm Fixed Price (FFP)
2. Time and Material (T&M)

Firm Fixed Price	Time & Materials
Defined parameters	Developers set parameters
Low flexibility	Good for flexible projects
Good for small projects	Grant creative freedom: Price can vary

Agile Software Development

Agile software development starts with a product vision and does not specify exact system features, but addresses the desired high-level functionality of the system. Deliverables often are the functional, working software (deployable code) that is produced through a repeatable iterative delivery process in a production environment. Continuous Integration/Continuous Delivery (CI/CD) is a practice in which code is regularly/iteratively being integrated and delivered. Procurement of this type of development, can also be done using agile principles throughout the contracting process. You may end up awarding contracts with a series of sprints using agile acquisition methods. In this type of environment, you could look at fair and reasonable prices by evaluating the labor of each sprint, using expertise level, length of time, size of team, etc. (see Estimating Labor section). When evaluating pricing sprints, be cautious that pricing isn't solely based on labor, and there are considerations of the delivered value. Software code that passes a CI/CD pipeline successfully and is in a production environment is what is valuable to the warfighter. Significant weight should be placed on getting software into production.

The chart (right) denotes the agile software development cycle:



Price Considerations – How to Evaluate

Value & Functionality

The acquisition strategy should balance Innovation and Budget: "Design-to-price" was a distinct departure from traditional acquisition programs, which typically focus on achieving the highest possible performance, often resulting in cost increases. We want to focus on choosing the most ADVANTAGEOUS solution; which *should* include price as a distinguishing factor. When choosing other-than-the-lowest-priced technical solution, the Contracting Officer (KO) should be able to explain why the solution or function exceeds the others, and the increased value to the warfighter is worth the additional dollars. Consider the Price/Value equation (image on the right) as a theoretical way to think about value. This scorecard method can help evaluate price reasonableness in a sole source environment by forcing the comparison to like-items. This analysis is where elements such as functionality, feature, data, storage, age, version, etc., can be considered.

Company A Vs. Company B

Stakeholder's Chosen Criteria Score (5 best to 1 least)		Company A		Company B
Functionality / Compatibility: Does the product contain the necessary programs to complete the work? Do they require extensive Training to use properly?	Score 5	All needed functions present, easy to learn.	Score 3	Most functions present, some not fully developed.
User Friendliness / Customer Service: Training & Customer Service Packages	4	Training - 4 classes and unlimited customer service 24/7	2	Training - 2 classes and 60 minutes of customer service a month
Compatibility: Does the product easily replace the previous Solution? Is there going excessive downtime waiting for the program to finish installation?	5	Solution easily integrated into current one. Minimal work time lost	4	Solution estimated to take 2-3 weeks to finish. Several work hours lost
Agility/Adaptability: Does the vendor allow for easy transaction between bundles/products in response to business growth?	4	Contracts are time based. Prevents quick response to growth.	2	Contract can be changed for a fee. Modest response time.
User Friendliness: Is the solution easy to learn how to use? Are Customization options available to compliment Individual work styles?	3	Training time minimal. Modest options for customization.	2	Significant training time required. Few options for customization
Total Score out of 25	21		13	
Over and Above:				
Purchase Software		**$155,000.00**		**$85,000.00**

Price/Value Formula:

Score rating for First software (1-25)	21
Score rating for Second Software (1-25)	13

Price for First software score	$155,000
Price for Second software score	$85,000

- In the example above, the team reviewed the software and subjectively scored Company A 21/25 and Company B 13/25. Dividing Company A's (21) score by Company B's score (13), concludes that Company A's software is 1.6x better than Company B's.
- The price of Company A's software package ($155K) is then divided by the price of Company B's software package ($85K), resulting in a 1.8x markup.
- Conclusion: "I am paying 1.8x the price for something I rated 1.6x better".

Create the narrative "I am paying 1.8x more for something I rated being1.6x better"
"IS THAT RIGHT?"

Compatibility

There is certainly a continuation of thought from the prior value section. How easily will it integrate with and/or replace the current solution? A product with high compatibility will also have high value and may dictate how many features the new solution has, and how easy they are to learn. Compatibility is both human to machine, and machine to machine. It is crucial to think about compatibility as a pricing element in addition to the actual price tag. Learning a complex new software can be time consuming and costly to an organization. Resources required to implement the software solution should be considered when reviewing the overall value of the procurement. Instead, it is better to utilize a simple and easy-to-learn solution that can save valuable administrative work hour costs. The product itself is only worth the manual work hours it saves. If the product costs more than the manual work hours incurred the return on investment is either negative or considerably low.

Brand Recognition

First and foremost, brand recognition and popularity (when crucial to functionality) absolutely influence price. Consumers may be willing to pay more for simple brand recognition; however, more explainable, consumers could be more likely to pay more for software platforms which integrate communication as a key element – the number of users affects the viability of the product. It is likely that characteristics such as increased functionality or user satisfaction are what led to a brand's popularity; however, above all else, the value the solution provides the user should be the basis for price.

In order to determine the effect of brand recognition on a product's pricing, two prominent products were examined. The first, Slack, was created as a workplace communication app that served as a platform for customers to send messages, collaborate, and to host virtual meetings with each other. Mattermost was then created as a direct competitor to Slack and provides a virtually identical list of features. In fact, the only difference between the two is that Slack features an "Activity/News Feed" feature while Mattermost allows for surveys and feedback. Following are more comparisons.

- Currently, Slack reports millions of users daily and, according to Capterra, a website that rates software, rated Slack at a 4.6 out 5 with 18,903 reviews. It is also utilized by 77% of Fortune 100 companies, whereas Mattermost is rated at a 4.4 with only 98 reviews from the same source
- In terms of price, Slack will run a user $6.67 per month, while Mattermost charges $3.25 per month. The difference between these prices could confirm that the brand recognition for Slack increases their demand which allows them to increase their price. Not to rule out that their considerably greater traffic may increase their upkeep costs for things such as servers and customer services
- Mattermost's lower price may be explained as their upstart and upkeep costs are generally lower. They likely did not have to spend as much on research and development costs due to being so like Slack. Their lower customer traffic may also help save costs on servers. However, Mattermost recently closed a $20 million funding contract to expand their applications and services. Therefore, the effects of brand recognition should be more prominent when the competition meets or exceeds the original

Estimating Labor

Cost is not price. That said, a method of evaluating price can be estimating elements of cost. Determining the price of larger projects can be accomplished by adopting an iterative development estimate. This plan seeks to break down the whole project into elements of cost for a specific team/sprint which allows for a more simplified approach to determining the overall price of the proect, in other words, a bottom-up analysis. Although this looks like cost data, there is a distinction between certified cost & pricing data (required in FAR15) and using cost elements to estimate price reasonableness(see table right). There are no restrictions from Government personnel evaluating price information by estimating costs. Generally, a buyer can use resources such as salary.com, glassdoor.com or even Bureau of Labor Statistics (BLS) to estimate the direct labor costs.

The buyer can also use information from the Defense Contract Management Agency (DCMA), or Defense Contract Audit Agency (DCAA) to translate direct labor or salary rates into fully burdened labor rates. This is simply a different way to look at a proposed price by using commercial market labor rate data and multiplying them by estimated industry burdens. Even if the proposed solution was not presented with separate, identifiable cost elements, or a level of cost element granularity, the Government can reverse-engineer the proposed price to gain an understanding of the estimated hours proposed, or the Level of Effort (LOE) needed to accomplish the task at hand. The 'markup' rate is commonly between 1.5x - 2.5x depending on the industry; however, it can deviate outside of this range.

Estimating for Labor - An Interactive Approach

		Total	20-Jan	20-Feb	20-Mar	20-Apr
	PE Hrs	180	20	30	60	70
	Factory Cost	32,730	3,205	4,808	11,558	13,160
Engineering - Type 1						
	PE Hrs	120	20	30	30	40
	F/C ($160.25 for 2020)	19,230	3,205	4,808	4,808	6,410
Engineering - Type 2						
	PE Hrs	20	-	-	10	10
	F/C ($162.55 for 2020)	3,251	-	-	1,626	1,626
Project Specific	PE Hrs	15		5	5	5
	Factory Cost	3,843	-	1,281	1,281	1,281
Central Shipping	PE Hrs	10			5	5
	Factory Cost	2,562	-	-	1,281	1,281
Schedulers	PE Hrs	25		10	10	5
	Factory Cost	6,406	-	2,562	2,562	1,281
Hardware		50,000			25,000	25,000
Tooling		17,500		17,500		
Purchased Engineering		100,000		100,000		
Other Costs						
Factory Cost		**268,525**	**6,410**	**135,958**	**53,240**	**55,164**
BE	1.30%	3,491	83	1,767	692	717
SUBTOTAL COST INPUT		272,016	6,493	137,726	53,932	55,881
G&A	4.25%	11,561	276	5,853	2,292	2,375
IR&D	3.87%	10,527	251	5,330	2,087	2,163
SUBTOTAL		294,103	7,021	148,909	58,311	60,418
FEE / PROFIT	15.00%	44,115	1,053	22,336	8,747	9,063
COM - BE	0.10%	269	6	136	53	55
COM - G&A	0.07%	188	4	95	37	39
Total Price		338,675	8,085	171,477	67,148	69,575

* Cost of Money (COM)

- As a form of estimating, an Analyst who regularly performs cost analyses on traditional defense contractors, can obtain a markup rate by labor category simply by taking an existing proposal, putting one (1) hour in a labor line, and comparing to the final proposed price in the model.
- The DCMA Commercial Item Group (CIG) built a model to estimate this markup using public census data. This model can be located on the CIG's public facing website at: https://www.dcma.mil/commercial-item-group/ Then find "Additional Resources" and click "Labor Rate Census Pricing"
- GSA CALC is a tool which has *wage-at-price* information *(https://calc.gsa.gov/)*

User Friendliness and Customer Service

When looking at how the product is priced, keep in mind the product's user friendliness and customer service. A product's "user friendliness" is determined by how extensive training will be for the new solution, and whether the vendor is able to help with training and other problems in the future. It is common for vendors to offer "premium or basic packages". These packages often contain different numbers of features and potentially different levels of customer service. Premium packages will cost more than basic ones but will also provide a greater number of features and more engaged support from the sellers.

Agility/Adaptability and Exit Strategy

Today's needs seldom represent the needs that may develop in five to ten years. It is only natural that a company experiences changes over time and develops different needs for its software. "Agility" refers to a vendor's willingness and ability to match the needs of the buyer over time. This may entail simple changes such as software updates, purchasing and installing a greater number of licenses, or completely reformatting the solution to account for growth and restructuring. This is the step to consider if a vendor has costs for modifying their product to better fit the buyer's needs.

If a vendor, for whatever reason, cannot keep up with these changes, what are the necessary steps for breaking away from that vendor? An example of a simple exit strategy is one where you can simply contact the vendor and request the current contract be terminated. A more complicated example of an exit strategy may involve having to continue a contract for a set period of time or having to pay a fee for the time remaining (i.e. early termination). Exploring an exit strategy for a product should be an early step in the decision-making process. It could be extremely costly for a company to have to jump through multiple hoops and termination fees to break free of a contract. It is also generally more expensive to have to change software solutions entirely as the cost of re-training employees and modifying the new solution can quickly increase costs.

Financing/ Payment Structure

The costs incurred by purchasing software almost always tend to be more than the sticker **price**. Data transfer fees, set-up costs, customizations, and a myriad of other different fees all contribute to expanding costs. It is also worth noting that it is becoming increasingly popular in internet-based SaaS (Software as a Service) markets to charge fees on a monthly subscription basis rather than as a single license. This can be advantageous to smaller companies as it typically keeps start-up costs low and may help establish customer service lines with the seller. However, this platform can quickly become a problem for larger companies as monthly fees can exceed the cost of a single license given enough time.

For example, if Product X offers a single license fee for $1,000 or a monthly fee of $100/month, it would take less than a year for that monthly payment plan to exceed the cost of the single license. While the single license ultimately costs less than the monthly payment plan over the long term, the monthly plan offers the consumer a viable option if they don't have $1,000 to pay upfront. Additionally, the monthly plan offers benefits, such as software updates, that are not offered through the single license. It all boils down to what the company needs and what stakeholders are willing to invest. SaaS monthly plans may include software updates, which may not be included on upfront cost.

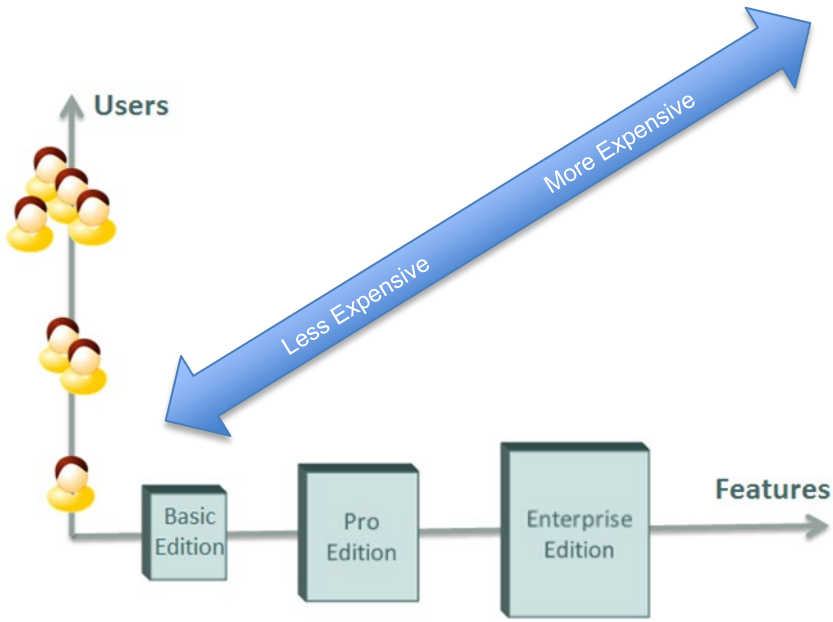
To be a Great Buyer, "Be the Seller"

Starting Point

The first thing that developers consider when they set their prices is their return on investment (ROI) to produce their product. It costs a lot of time and money to produce software. The more complex the software is to develop, the more developers need to spend/charge. The second thing developers consider is their target audience. James Parsons, a writer and marketer for "Entrepreneur" claims that software never has a "best price" for either buyers or sellers. This is because two virtually identical software products can both succeed because they have different target demographics. Keeping these factors in mind while researching software developers can help give an understanding on why prices are set the way they are. It will also help to better find a developer that will work with the specific company needs.

Pricing Models Used By Developers

Most developers tend to practice some form of pricing strategy based on who their buyers are, in that they almost never charge two companies the same price for their product. This is because not every business shares the same needs for features. The most effective structure vendor's use for accomplishing this task is the Two-Axis Pricing Model (see chart below). The first axis refers to "volume" and usually accounts for how many personnel require the software as well as how much digital storage space they will need. The second axis refers to the features of the product and whether they can be sold individually or packaged.



Increasing the number of features increases the cost of the solution

Large firms will find greater value in a more complex solution

Small firms can keep it simple and cheap

To put it simply, if a product has 10 features, and one company only needs four of them, while another needs six, a competitive vendor will likely not charge these respective companies for the full 10 features, opting to only charge for what is needed. Redundant charging of features is likely to turn potential customers away. Determining what features can be bundled into one package or whether each feature needs to be charged

separately can help keep costs low. This level of flexibility should be seen in a developer as their services can be upgraded as the company grows and their business needs change.

Billing and Discount Strategies Used By Developers

A Stanford Graduate and Marketer, Jake Saper, gives the advice that it is marginally more effective for a SaaS vendor to bill annually as opposed to monthly, and even more critically, to get paid up front. Discounts may also be given to new customers by lowering the number of units the customer pays for rather than lowering the core price. The reason core prices, or the price of the most basic product without extra features, are seldom lowered is because this has a tendency of developing a referral-based price expectation for future customers, which can result in negative publicity, or higher startup expenses for the vendor.

Alternative Resources & References

Jia, M. (2017, April 21). 7 Important Factors To Consider When Choosing Enterprise Software. Retrieved January 1, 2020, from https://www.topbots.com/7-most-important-factors-evaluation-enterprise-software-technology/.

Jia, M. (2017, April 21). A Detailed Vendor Sourcing Framework For Enterprise Solutions. Retrieved January 1, 2020, from https://www.topbots.com/a-detailed-vendor-sourcing-framework-for-enterprise-solutions/

Shleyner, E. (2019, November 5). The 8 Key Factors When Buying Marketing Software: Databox Blog. Retrieved January 1, 2020, from https://databox.com/how-to-buy-software/

Mays, K. (2018, February 12). Top Factors to Consider When Buying a Software. Retrieved from http://www.ontwik.com/top-factors-to-consider-when-buying-a-software/

Parsons, J. (2019, June 10). How to Determine How Much to Charge for Your Software. Retrieved January 1, 2020, from https://www.entrepreneur.com/article/334977/

Saper, J. (2017, October 2). How to Price Your Software: 101. Retrieved January 6, 2020, from https://medium.com/@jakesaper/how-to-price-your-software-101-4762fb939dd/

Law, Ryan. “The Ultimate Guide to SaaS Pricing Models, Strategies & Psychological Hacks.” *Cobloom*, 6 May 2019, www.cobloom.com/blog/saas-pricing-models.

Staff, Rhumbix Editorial. “How to Choose Between a T&M Contract vs Fixed Price Contract.” *Rhumbix*, 8 Aug. 2019, www.rhumbix.com/how-to-choose-between-a-time-and-materials-contract-vs-fixed-price-contract/.

Team, Synopsys Editorial. “5 Types of Software Licenses You Need to Understand | Synopsys.” *Synopsys*, Google, 2017, www.synopsys.com/blogs/software-security/5-types-of-software-licenses-you-need-to-understand/amp/.

Agile Team Estimator. (n.d.). Retrieved July 19, 2020, from https://techfarhub.cio.gov/custom-tools/ate/

Rahmani, B. (2020, March 05). □ Mattermost □ Slack ☐ Which Way Should You Go? Retrieved September 11, 2020, from https://www.troopmessenger.com/blogs/mattermost-vs-slack

Slack vs Mattermost. (n.d.). Retrieved September 11, 2020, from https://www.capterra.com/team-communication-software/compare/170524-135003/Mattermost-vs-Slack

Skok, D. (2011, October 31). Multi-axis Pricing: A key tool for increasing SaaS revenue. Retrieved November 09, 2020, from http://assets.businessinsider.com/multi-axis-pricing-a-key-tool-for-increasing-saas-revenue-2011-10

Project-Management.com. (2020, October 23). Agile Development Methodology & Principles for 2020. Retrieved November 09, 2020, from https://project-management.com/10-key-principles-of-agile-software-development/

Kumar, A. (2020, June 08). What is DevSecOps? Benefits of Adopting DevSecOps. Retrieved November 09, 2020, from https://www.devopsschool.com/blog/what-is-devsecops-benefits-of-adopting-devsecops/

Barron, A. (2014, July 30). Pizza as a Service. Retrieved October 20, 2020, from https://www.linkedin.com/pulse/20140730172610-9679881-pizza-as-a-service