HOW TO BUY THE RIGHT SOFTWARE

arena

PRACTICAL HELP FOR COMPANIES OF EVERY SIZE

ptc

Contents

arena

INTRODUCTION

Welcome to the step-by-step guide on how to buy the best cloud software for your business. This series provides the collective wisdom of many people who over decades of work life have been involved in every aspect of software for business needs—from identifying requirements, selecting, purchasing, implementing, using, and expanding software installations, to selling software, and even killing failed software projects. We've written the guide we wish someone had given us for our early software search projects.

This guide does use specific business needs or software solutions as examples because we do sell software for specific needs. However, the guide is designed to help you buy the right software for any business need—from product development and quality processes to manufacturing operations, finance, design, and digital marketing.

We recommend you use the guide sequentially from start to finish; your search process will go smoother and your degree of success higher. However, you can also jump to any chapter for immediate help at your particular software search step.

Examples

Throughout this guide, you'll find examples based on what we've learned in our own software space. These composite examples represent our collective experiences as both the teams and leadership selecting software and as the vendor participating in selection processes, as well as input from our customers through interviews and surveys.

Jump to the Resources

This guide has many templates. If you want to jump to the resources, you'll find all of them within the master software search project template. This is a Trello template board you can use to create a board for your team. It has links to all the other resources we mention in the guide.

CHAPTER 1: BUSINESS NEEDS DRIVE EVERYTHING

Why are you shopping for software? This isn't a glib question and it will be harder to answer than you expect. Let's consider common reasons companies look for new software solutions.

Reasons for New Software

- Current software: not supported, problematic, expensive, untrustworthy, and/or hated by users
- No software: paper processes, so lots of chaos, pain, and resource sink

Possible Problem Areas That Cost $$

With this in mind, it's good to consider what the impact of the current situation is—what are the relevant challenges or problems affecting your business?

- Team costs: inefficiencies, inaccuracies, miscommunications, fumbles, lack of trust, onboarding new team
- Failure and recovery costs: poor quality, business delivery problems, service and support issues
- Opportunity-lost costs: what you could have done with better/faster/more accurate processes
- Brand/customer relationship costs: negative impact to your brand or customer issues that cost contracts or future business

INSIDER TIP

Now, any #FOMO reasons for software shopping need to be ignored. Software vendors do a great job selling solutions based on beautiful videos, websites, and stories, but you may or may not actually need the solutions they are selling. Don't give in to impulse shopping—it has no place in enterprise software evaluations.

So, why are you shopping for software in the first place? We ask because we've seen companies fail at software selection. Failing to get the right solution to solve business challenges often stems from undefined business requirements. Your evaluation process should be based on a solid understanding of your critical and nice-to-have business needs. These business needs should be the guiding compass you use for everything else from requirements definition to vendor research to purchase. Business needs will drive everything starting now through implementation and future upgrades.

Identifying the Business Needs

You might already know your business needs, but if you don't, this short exercise will help. If you personally know the business need(s), consider doing this exercise with your team, the other stakeholders, and your executive sponsor. Doing this will promote buy-in to the entire process from selection to implementation to go-live.

Consider and answer these fundamental questions:

1. Why did you start looking for software?
   □ Current software challenges (and which software) ________________________
   □ No software
2. What are the business processes that need improvement?
3. Who participates in these processes? Think beyond your team to others impacted as you follow the process from start to finish.
4. Who owns these processes? Which teams have revenue or cost responsibilities?
5. What are these problems costing the business right now in these areas? You'll look more closely at ROI later, but this calculator will help you assess costs of mistakes for product and quality management work based on your inputs.
   □ Team costs
   □ Failure and recovery costs
   □ Opportunity-lost costs
   □ Brand/customer relationship costs
   □ Compliance costs

This exercise will allow you to clearly state the business objectives. You want to take this information and get to a business needs statement in the form of:

We must improve W and have better X to accomplish Y and meet Z.

Circulate this "Business Needs" statement until you have input and agreement from all stakeholders.

EXAMPLE

Let's apply this exercise to an example company that has product development and quality challenges.

High Tech Electronics Company With 1 Product Launched and 3 More in Development

Why did you start looking for software?
Current software challenges (and what software is used): We started because engineering change processes were delayed or incomplete. We use spreadsheets, email, shared drives, etc. for this process.

What business processes need improvement? And what are the problems?
Change control and approval process: by email with multiple teams including factory team in China and design in U.S. Problems include difficult approval feedback process, no historical record of approvals, inability to know which revision is last released.

Who participates in these processes? Think beyond your team as you follow the process from start to finish.

- Product management teams: software, hardware, electrical
- Manufacturing ops team including production engineering
- Quality
- Supplier chain managers
- Suppliers and contract manufacturers

Who owns these processes? Which teams have revenue or cost responsibilities?

- Quality: shared cost responsibility
- Operations: shared cost responsibility
- Product management: design delivery responsibility

What are these problems costing the business right now?

- Team costs: poor change management, supplier confusion, and prototype failure negatively impacted new product development by 12 days
- Failure and recovery costs: field failure last quarter resulted in rework charges
- Opportunity-lost costs: delayed product launch resulted in estimated 5% miss in yearly revenue goal
- Brand/customer relationship costs: field failure resulted in relationship issues with two significant customers; unknown level of impact at this time

Example Business Needs Statement

What are the business needs for this example?

> Your first answer might be something like this:
> We need a way to control our product information, track change history, and know the latest approved build and the current manufactured build at any time.

Correct, but not at the level of business need yet. If you find your first business needs statement is more about processes and operations, push against that statement to articulate what happens when you do or do not have these needs met.

> We must improve the quality of our products and have better product development control to set accurate product launch dates and then meet those commitments.

Gaining Executive Sponsorship

Once you have a business needs statement, your company should prioritize solving these needs. Get executive-level agreement that the business needs statement you are crafting is a priority. We have seen committees select the perfect software, but the buying process was derailed for another project. And, some committees get the money but lack the executive support needed across the teams to make the implementation successful. A clear, executive-backed business reason for the project will help overcome naturally negative feelings about change during implementation and go-live.

Go back to your exercise above and look at the process owners you've identified. You may need executive sponsorship from one or more teams that have process ownership and/or revenue responsibility. Most successful software implementations have sponsorship across ALL identified teams that take part in these key processes.

Set your efforts up for success early by socializing the business needs to any executives that are not already involved. While you may not yet get full buy-in, you can set context and update your business needs statement after the stakeholders have supplied input. You may need to quantify the costs identified in the business exercise. Coming up with actual dollar values will require some investigation and calculations. If your business needs are product and quality management related, you can use this tool to help.

Agreement to Shop for Software

Clear business needs drive successful software selection. Your business needs statement should articulate the current costs of not fixing the problem to gain executive sponsorship. Do this and you can not only begin looking for software that will help—but actually buy and implement the software you've selected.

CHAPTER ACTIVITIES SUMMARY

- ☑ Complete business needs exercise
- ☑ Draft a business needs statement
- ☑ Calculate current costs of business needs
- ☑ Gain executive sponsorship

CHAPTER 2: PLANNING YOUR SOFTWARE SELECTION PROCESS

We laugh at Bernstein's words. In order to get through a software selection successfully, you'll need a plan. You'll also need urgency that prioritizes the plan above all the other competing projects clamoring for resources. If you started at the beginning of this guide, you have identified your business needs, calculated the costs to continue as you are, and lined up some executive sponsorship. Now is a good time to get a plan on how you'll run this selection process.

What's in the Plan?

Teams

In a software selection plan, you should identify different types of team members. Change is hard, even when people say they want something new. Knowing who is impacted by this software selection and how they are impacted is critical to the success of the selection, implementation, and long-term use of the software.

Users. You'll want to identify the participants of the business processes in question as they'll become the users of the software you buy. The user list is not static. As you define requirements and learn more about every team touched by each process, you'll make changes. Classify users in three categories based on how they would use a solution: power participants, occasional participants, and readers. For the initial plan, you may not identify all users if your company is large but try to identify some users in every category.

> "
>
> To achieve great things, two things are needed: a plan, and not quite enough time.
>
> – Leonard Bernstein

Stakeholders. Note who owns the business processes the software will support. They have an interest because this project will affect them and their teams directly.

Influencers. We're going to borrow from social media marketing here to identify people who can sway others for or against the project or eventual software selected. Influencers can be any level in the company and are respected for their knowledge and skills.

Sponsors. Continue to seek sponsors—executives who can help make the project a priority and/or provide budget and resources. Sponsors ideally take some level of responsibility for this project and will build support with others.

Here's a helpful template that shows one way you can put the team identification together.

Requirements

The next chapter in this guide covers requirements capture in detail. For your plan, you want a requirements-gathering phase. You'll use the active inquiry model we provide to ask each team member. The goal is to pinpoint the actual requirements you have for a solution to your business need.

Research

You have powerful tools today for solution research. Gone are the days when you had to rely solely on what vendors and a few questionably agnostic industry analysts told you about software options. In your plan, you'll want to include several different research tasks including digital market surveying, digital and direct vendor engagement, and solution verification activities.

Urgency

With your business needs and rough cost calculations, you've started to build an argument to prioritize this project. During the selection process, you'll build a financial case based on the business needs, costs of not solving the challenges, and the costs of the solution you select.

Decision-Making

Here at Arena, we've found that 78% of our buyers select software by committee, but one member of the committee often has an outsized impact on the decision. In articulating your business needs and identifying the team impacted, you'll see how the decision-making is going to happen. Later in this guide, we'll talk more about managing committees. Right now, you can consider the dynamics of stakeholders, influencers, and sponsors along with how decisions are typically made at your company.

Project Plan

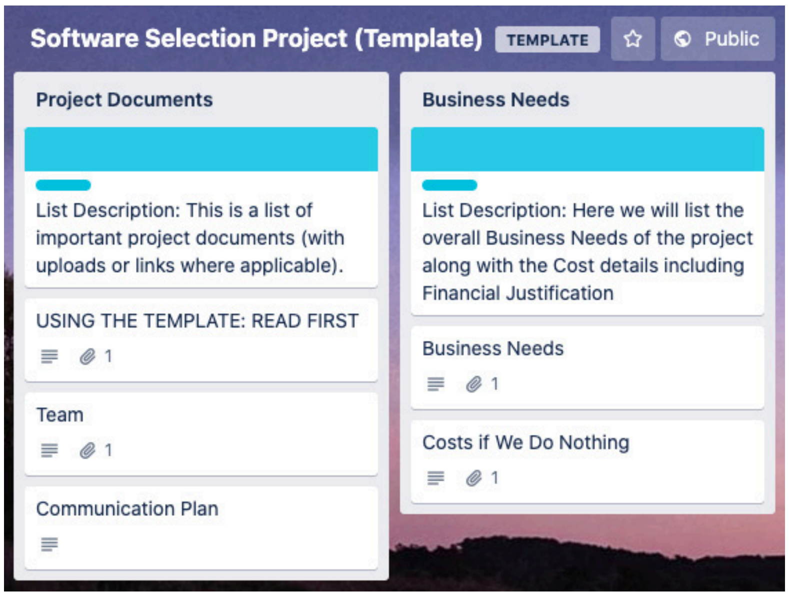
You should have a documented project plan for your software selection process. A documented plan can be shared across the team, keep you on track with tasks, and be used to communicate findings and decisions. You can use any variety of project plan tools—from Excel or Google Sheets to any number of newer project and task tools.

No matter how you manage the project plan, it should:

- Be accessible
- Identify major tasks
- Assign task owners
- Use due dates even if you need to adjust as you progress

Once you have the plan, you need to get buy-in from all stakeholders and sponsors. Have everyone review and provide comments. You can do this asynchronously or in a team meeting. It is important to take the time for everyone to agree to the overall plan you will follow.

Free Resource: Use this Trello template to get started. It includes template sheets for teams, requirements, and more for a PLM/QMS software selection project.



Software Selection Project (Template) TEMPLATE ☆ Public

Project Documents

List Description: This is a list of important project documents (with uploads or links where applicable).

USING THE TEMPLATE: READ FIRST 1

Team 1

Communication Plan

Business Needs

List Description: Here we will list the overall Business Needs of the project along with the Cost details including Financial Justification

Business Needs 1

Costs if We Do Nothing 1

Communication Plan

A project plan is like a shopping list until you share it—then it becomes a dinner party. You need to have a communication plan. It doesn't have to be complex, but it does need to be **consistent**. Your communication plan should include what information you will communicate:

- To whom
- At what frequency (e.g., every Monday, every other Friday, once a month)
- By which method (e.g., email, wiki, chat space, updated shared document)

To craft this communication plan, it is good to go back to that team list and consider who needs what level of information. It's common to provide two levels of communication regularly—a high-level summary for sponsors, influencers, and stakeholders, particularly those in executive management, and a more detailed communication for active selection team members (note that some people may receive both communications).

Finally, consider how to make your communications as easy to read or consume as possible. A good rule is to make each communication as short as possible while still providing the information needed.

CHAPTER ACTIVITIES SUMMARY

☑ Draft a software selection plan outlining the five components, including a first pass at team identification

☑ Get buy-in on the plan from stakeholders and sponsors

☑ Set a communication plan

CHAPTER 3: CAPTURING REQUIREMENTS

Shopping for almost anything, including software, is as easy as a Google search today. With a few phrases and clicks, we can quickly see what's available in any category from phone apps for meditation or financial planning to enterprise software for product lifecycle or quality management. Amazing and convenient, right? We also can quickly go down rabbit holes of possible solutions without the guidance of requirements. We do cover the research and evaluation stage later in this guide, but before you start Googling or checking out G2 reviews, you need to articulate and prioritize your requirements.

What Are Solution Requirements?

You may be familiar with product requirements, which range from market, customer/user, and business to functional, technical, operational, quality, and more. While you won't be doing full requirements management activities as you do with your own products, you need to identify requirements for the solution that will meet your business needs.

Solution requirements are capabilities or conditions the software you select must have to meet the business needs and provide confidence in continued success.

Requirement Categories

You can name and sort requirements into categories. This will help you as you work through requirements prioritization or weighting discussions as well as provide some organization to your list. You may also further group requirements into areas under the main categories.

Business Needs

Explanation:
The specific information, behaviors, rules, and operations a solution should have to ensure desired outcomes and meet the business needs.

Possible Requirements Areas:
Business-need requirement areas will map to the processes that support your business needs—so this will vary.

Get a PLM/QMS requirements template in the project plan template for examples of areas for product companies shopping for PLM and QMS solutions.

> “
>
> The perfect is the enemy of the good.
>
> – Voltaire

I

INSIDER TIP

Two Dangers to Doing Requirements Well

1. *Confusing preferences and requirements*
2. *Finding new problems to solve*

Functional

Explanation:
The manner or environment in which the solution needs to operate, including additional expectations you have for the solution or qualities it should include across all business need(s).

Possible Requirements Areas:

- Security and access
- Administration capabilities
- History and audit
- Configuration and extensibility
- Usability and acceptance
- Data flows and integrations
- Search, reporting, analytics
- Availability, performance, and system requirements

Non-Functional

Explanation:
The attributes of the solution as well as the vendor's strength (financial, historical, customer success, customer support) to support your projected growth and future needs.

Possible Requirements Areas:

- Vendor offerings and customer base overlap with your business
- Vendor product development philosophy, release history, and vision/roadmap
- Vendor customer success
- Vendor customer support and care
- Vendor financial health

Identifying Requirements

Creating a full requirements list from scratch is work and can be challenging if you have not defined solution requirements before. Instead, upcycle! Gather a few templates for requirements in the solution space you need and use them as the basis of your own.

Where to find templates?

- Vendors (should be free!)
- Teams that have done this before (free!)
- Peers at other companies with similar needs (free!) —reach out to them via LinkedIn and other networks
- Software consulting companies (usually as part of a fee-based service)
- Industry analysts (often fee-based)

Prioritizing Requirements

There is **no perfect solution**. You won't meet all stakeholder needs and you can't make all users happy.
You can find the **best solution**—the solution that best:

1. Meets business needs based on prioritized requirements
2. Will be adopted by users
3. Fits within your overall cost structures and resource plan
4. Will grow with you in the future

To get to the best solution, you will need to prioritize requirements.
You can use a simple prioritization, such as
1 = must have, 2 = nice to have, and 3 = do not need.

As you evaluate solutions, you'll use this prioritization value to determine weighted final scores, so use numbers.

Don't overcomplicate the priorities. The simpler the scheme, the more valuable your conversations about what is needed, preferred, or just technology-induced dreams.

Adjusting Requirements

- As you engage with vendors and see the functionality they offer, you may realize that what the vendors are providing differs in scope more than you expected. Indeed, sometimes no vendor does all you need in one package. In these cases, you want to talk with vendors (and some of their customers) about how they help with your business needs and how they don't. With structured data and standards, many companies use data flows and integrations across systems as needed. This removes the pressure to settle for a cumbersome, expensive, or unfriendly "all-in-one" solution.

CHAPTER ACTIVITIES SUMMARY

☑ Find one or more solid requirements templates to jump-start your identification efforts—we provide a PLM/QMS requirements template for you in our master software search project template

☑ Get the team together (virtually or F2F) and work on completing requirements identification and prioritization

☑ Put the requirements document somewhere accessible to the team, preferably attached to the project plan (this project plan template includes the requirements template and more)

CHAPTER 4: ENGAGING AND EVALUATING VENDORS

Where the Rubber Meets the Road

Now that you've considered your business needs, created a project plan, and captured your specific requirements, you're ready to start evaluating vendor solutions. Your team should focus on your needs and not necessarily the capabilities that each vendor may want to emphasize. It's important not to get enamored with bells and whistles that don't directly help your team and company meet the stated requirements.

More importantly, remember that you're reviewing more than a software solution. You're looking at each vendor's complete set of services and approach to not only implement software but to help your company fuel adoption, gain optimal benefits, and continue to be successful for years.

Look for Transparency

This is worth repeating: There's no perfect software There's no single system that will meet every one of requirements in exactly the way you want. Your goal is to for the best fit: the vendor solution and that comes closest to meeting your requirements.

The best vendors understand the power and benefits that come from having more transparent dealings and are not afraid to "just say no" when they can't meet a particular need.

Warning: If any vendor says "yes" too often or to almost every question—consider it a red flag. They're either misleading you, or they're setting you up for a costly custom-coding project that may lead to a long implementation or, worse yet, failed deployment.

INSIDER TIP

Let's Talk Demo

You might want to rush to demo. Our brains are wired more and more toward kinetic, multisensory learning. For many enterprise solutions though, it's common practice for vendors to engage with you first in a live discovery meeting to ensure they understand your business needs. This is a good idea and in your best interest. Why?

If the vendor understands your specific needs before providing a demo, they can confirm they fit your needs and can provide a more tailored demo to address your specific requirements.

Software Evaluation Is Easier Than Ever

Before Software as a Solution (SaaS) became popular, it was more difficult to evaluate vendor software solutions. Buyers often had to check out long (and infrequently updated) analysts' reports and weed out bias based on which vendors paid for good reviews. And, gone are the days when you must sit through multiple, long, and tedious sales engagements with onsite meetings and demos.

Not only are cloud-based SaaS solutions easy to use and deploy—they're also easier to evaluate. For this guide, we assume you are looking at SaaS solutions but almost all outlined here applies to on-premises solutions as well.

Here are six simple steps you should take to evaluate software solutions.

1. **Divide and conquer**
   As project manager, you don't need to do all the heavy lifting. You can delegate to committee team members to help conduct the vendor evaluations. Using your requirements document, each assigned person can review and rank vendor capabilities.

2. **Start with customer reviews**
   There may be dozens of vendors that provide solutions to your challenges. You can do web searches for top solutions, but we recommend going directly to independent customer review sites to search for solutions by product category. Just like Yelp can help you quickly zero in on the best restaurants or Amazon can help you find the best consumer product, customer review sites like G2 and TrustRadius are designed to help you quickly compare business-to-business software offerings.

   When referring to customer reviews, consider how many reviews are positive and how many are negative. If there are only a few positive or negative, then you may not place much weight on those reviews. However, a lot of positive or negative reviews is a good indicator regarding the vendor's solution and how it performs.

   To make the reviews easy and consistent, include additional high-level criteria for quicker assessment as well (see Vendor Customer Reviews template, also available in the software search project template).

   Once you've narrowed the field to the top three or four solutions, you're ready to do a deeper dive.

3. **Research the vendor's website**
   Before you invest time talking to vendors, make sure there are no deal-breakers that disqualify a given vendor from consideration based on your online evaluation. As an example, look for industries served or customer success stories that show the vendor has experience solving issues for companies like yours. For instance, Arena is designed to help companies with physical, discreet products comprised of hardware, software, and electronics. We're not a good fit for, say, an oil refinery.

   Armed with the feedback from customer reviews and overall vendor rankings, you need to get firsthand information from each vendor. You should start with a review of each vendor's website material, product videos, and other resources provided.

4. **Engage the vendor**
   There's only so much information you can gather online, so now it's time to engage the vendor's sales team to get an in-depth understanding of each solution. You can contact them via phone, but it's often easiest to make the request via online chat or "get demo" directly on the vendor website.

   To make discovery and demo calls the most efficient and effective, you want to:

   - Include your entire software evaluation team to ensure each team member can review the solution capabilities that affect their team
   - Invite your executive sponsor
   - Provide vendors in advance with a general statement of your business needs and environment as well as specifically what you would like to accomplish in the first meeting (and any subsequent). While vendors will probably confirm this information, it saves you time because the vendors are going to be more focused.
   - Have the vendor clearly articulate what will and won't happen in a session so everyone attends with the correct expectations and a known agenda. Will there be a demo at this meeting or next? Is this meeting the right place for a deep dive on features, technical details, or pricing?
   - Record the meeting for anyone who cannot participate from your team. If the vendor set up the call, ask for the recording—you won't have to relay information and possibly lose some of the nuanced details in the process.

As you meet with vendors, work through your requirements to score each solution. Remember that you may need to update your requirements or the weighting as you learn about what is possible and how solutions meet needs. Go back to your business needs when it gets confusing to ensure your requirements support the business needs.

5. **Conduct vendor reference checks**

   Ask the vendor to provide two or three customer references that share a similar business model to yours (e.g., similar size, industry, product offerings). Interviewing references will help you determine whether the solution is truly the right fit for your organization.

   It is important to consider the role of the references in implementing and using the technology. Ideally, the contacts should be power users who were part of the software selection committee, use the system as part of their daily job function, and understand the processes that the software should support.

   Key questions to ask the vendor references:

   - *Sales process*
     - □ Were you part of the software selection committee?
     - □ What drove your company to invest in the solution?
     - □ Why did you choose this solution over the others?
     - □ Did you have to acquire additional solutions, services, and/or training after the initial purchase?
     - □ Did you have to justify the purchase to stakeholders? If so, how did you build a financial case?
   - *Implementation process*
     - □ Did the vendor provide implementation services or was it handled by a third party?
     - □ Was the implementation a fixed price or was it billed according to time and material?
     - □ What were the resource and time investments needed to implement the solution?
     - □ How did the software vendor handle any hiccups or surprises that occurred along the way?
     - □ Did you use any of the vendor's training courses or materials as part of the implementation? If so, how would you rate these offerings?
     - □ Did the implementation live up to expectations set in the sales process?
   - *Solution capabilities*
     - □ How often do you personally use the solution? Have you used similar systems in the past?
     - □ What do you like most about the solution?
     - □ Describe user adoption at your company.
     - □ How does the system's security model support your needs?
     - □ Do you feel confident the system will easily scale with your business as it evolves?
   - *Customer education and support*
     - □ What has been your experience with the vendor's handling of software issues, questions, or concerns?
     - □ How are upgrades performed? What IT resources (if any) are needed?
     - □ What level of communication do you have with the vendor regarding software enhancements or upcoming releases?
     - □ Do you have access to training materials or educational resources (best practices, tech sessions, webinars, etc.)?
   - *Results*
     - □ How many employees and business partners are currently using the system? What was the level of effort to onboard them?
     - □ Have you observed any key performance indicator (KPI) improvements since implementing the system?
     - □ Does the solution meet or exceed your expectations?

6. **Determine cost to configure and deploy**

   Once you've determined the vendor's solution is a good fit, you'll want to gather specific pricing information to help you determine not just the cost of the software, but the cost of the services and ongoing support and maintenance. Knowing the total cost of ownership (TCO) is important, so ask pointed questions to make sure you're comparing apples to apples across vendors.

   - *Software*
     - □ Find out the cost of the software: Is it based on users and, if so, are there different user roles that are priced accordingly?
     - □ Based on your demo and requirements, make sure you understand what software modules are included in the quote.
     - □ What are the options for payment? Most SaaS offerings provide monthly costs, but bill annually. See if there's a discount for entering into a two- or three-year agreement. Most vendors will offer a compelling discount to do so and you're not going to implement an enterprise system for only a year, so take advantage of a multiyear price agreement.
     - □ Are there any costs to upgrade to new versions or releases of the software? How are those deployed and at what frequency?
   - *Implementation*
     - □ What resources are needed from the vendor and from your team? How much time is necessary to configure, test, and go live?
     - □ How are these services quoted—fixed-price packages, custom statement of work, or time and materials?
   - *Training*
     - □ What is the training approach? Is it a train-the-trainer? Is it live or recorded? How many people should go through training and at what cost? (You should also consider how much training is needed, which relates to overall usability of solutions.)
   - *Adoption and scalability*
     Consider what happens AFTER you go live. Many companies implement software solutions based on current best practices, specific modules purchased, and existing business processes. However, no company stands still. To compete, you're constantly changing your business processes to shrink overall development and time to market.

- ☐ Ask each vendor whether they have a plan and process to help your company evolve and get the most from your application investment. As an example, check out how Arena looks at the entire customer lifecycle.
- *Customer support*
  - ☐ SaaS offerings normally include customer support in the cost of the software subscription itself, but you should confirm that is the case. Does the vendor have tiers of support or limitations to support provided as part of the SaaS subscription fee? Make sure you understand the normal business hours for engaging support whether this is done via email, online portals, and/or with live calls.
  - ☐ If the software is not SaaS, you may have additional annual maintenance fees and upgrade services to consider.

CHAPTER ACTIVITIES SUMMARY

☑ Identify the top vendors through customer reviews and other online research

☑ Review top vendors' websites and resources

☑ Meet with the vendors

☑ Interview vendor references

☑ Collect full cost information

Justification

Once you've met with all the vendors and received quotes and service details, you can summarize this information to compare costs and implementation details. The next chapter offers some best practices to help you make a compelling case to senior leadership based on the value and benefits you expect to gain by using the software.

CHAPTER 5: BUILDING A FINANCIAL CASE

At this point in your journey to buy a new software solution, you should have:

- Agreement on your company's business needs and requirements
- Understanding of the capabilities of shortlisted vendors' solutions
- A rough idea of total cost of ownership for each solution

You are now ready to quantify the expected benefits into a monetary amount to justify the software purchase, a critical and often required step.

Work With Your Executive Sponsor

Meet with your executive sponsor to find out the justification needed for purchase approval. Your executive sponsor will know what is required and may also be able to provide examples of similar justifications successful in the past. Confirm that the justification demonstrates how the software will address the company's goals and the executives' needs and motivations.

Two Key Parts of Your Financial Case

Quantifying is challenging but necessary and doable. If your team has done a good job at business needs identification, then you should have agreement that you need a solution and it is "worth it." But gut feel, even based on past experiences, isn't what executive sponsors and CFOs want.

The financial case should include two components:

1. Quantification of impact
   This is a believable calculation of how the company will benefit from meeting the business needs with the proposed solution. You will be making assumptions and estimating the impact based on changes to your business processes. Assumptions and estimations, though, should be defensible.
2. Business justification
   This is a succinct statement of the business needs and perceived business improvements that will be used to measure the impact of the solution. A brief and well-thought-out statement will help gain approval. Once you have the quantification of impact, you can combine that with your business needs to construct a short justification. Looking ahead, your business justification can be used to message with stakeholders and users during implementation and after for continued support.

Quantification: Increased Revenue and Lower Costs

Think about the financial-benefit factors of this project as a sum of two components:

Increased Revenue + Reduced Costs

Increased revenue for most companies comes from selling more products, improving customer retention, accelerating fulfillment, providing more predictable cycle times, and improving collections.

Reduced costs impact a company's bottom line. "A dollar saved is a dollar earned." As discussed in the first chapter, here are some possible areas where costs can be reduced with better information sharing and automation.

- Team effectiveness costs: inefficiencies, inaccuracies, miscommunications, lack of trust, new-user training
- Failure and recovery costs: poor product quality, business delivery problems, service and support issues
- Opportunity-lost costs: more inventory turns, what could you do with faster and more accurate processes?
- Brand/customer relationship costs: what impact have product launch delays or quality problems had on your brand or customer satisfaction?
- Installed system costs: if you replace older client/server software, the savings yielded by eliminating software support and maintenance, hardware and infrastructure investments, and personnel required to keep the software system working

Identify the Processes You Expect to Improve

Revisit your business needs and requirements by listing all that can be improved with the proposed software investment. better information sharing and more automation speeds and reduces errors. This frees up your teams to do more added work.

For each process, you want to measure the cycle time and error rates. And any manual or duplicate data entry processes that can be replaced with an integration will help eliminate errors.

At this point, you should assign a dollar value to expected improvements. Remember both factors: increased revenue and reduced costs.

INSIDER TIP

To make quantification believable, you need improvement metrics. Ask vendors for case studies or benchmarks from past implementations to give you an idea of the potential for improvements. You can also look to third-party research and analysis for case studies on costs of bad processes or expected improvements

Target Improvement Metrics

Estimate the expected improvements by determining a range based on best, expected, and worst-case improvements.

Translate Into Financial Benefits

To do this, talk with people in different roles who understand how each process contributes to the business. You will also need some basic company financials such as company revenue, cost of goods sold, gross margin, and daily team expense (employees). Your formulas will vary depending on both your business and what business needs you are impacting. Our examples are for product and quality business needs for product companies.

To calculate possible financial benefits based on your particulars, use this easy calculator for product and quality business needs.

Consider Existing Solution Costs

If you are replacing an existing solution, you need to add those related cost reductions to the calculations. For example, if you have on-premises client/server software, calculate the total cost annually for software, hardware, and upkeep, including FTE resources, particularly for homegrown solutions that require continued oversight. If you have a mismatch of off-the-shelf tools and manual, the actual software used may cost you little today in hard dollars.

Intangible Benefits

As well as hard dollar improvements, consider these benefits of having the solution you need: ability to respond to changes, be more competitive, and have better decision-making inputs.

- Ability to respond: The right software for your business needs should scale up (or down) as your company's needs evolve. Your company may have plans for continued and sustained growth (e.g., more products, higher volume, acquisitions, expanded services). And you may need to adapt quickly to external threats and opportunities. How do your processes scale with existing technology, personnel, and other resources? What parts of these processes are manual and not repeatable? What could be automated to help your teams address higher-level issues instead of routine administrative work? Contrast this with the new proposed solution. In general, your goal is to increase resiliency: the ability to handle the ups and downs in business cycles.
- Competitive position: Do your competitors use newer or more automated technologies? Are you using emails, spreadsheets, or on-premises software (client/server)? To stay ahead of your competition, you must continually explore new ways to gain a competitive edge in the marketplace.
- Decision-making inputs: Are decisions made using the most accurate and current information? Cloud software solutions eliminate silos and confusion. They help you measure performance in real time, providing decision-makers with better insights to improve business processes. For instance, you might discover process bottlenecks or learn about extra production capacity. Better information means better decisions.

It's unrealistic to place an exact dollar value on any software solution's ability to improve these intangible areas.
Yet, you should include these benefits in your justification.

Return on Investment (ROI)

Return on investment describes the financial benefit relative to the investment cost. At its simplest:

ROI = (Gain from Investment – Cost of Investment) / Cost of Investment

ROI looks at:

- Initial investment costs for solution: for SaaS, first-year subscription costs, services, training
- Yearly costs: for solution after first year
- Benefits: initial and continued
- Payback period: time needed to recoup initial investment costs
- Cumulative ROI: for enterprise solutions, the time period is often 3-5 years

EXAMPLE

Real-World Example From a Medical Device Company

Accuryn Medical reported their ECO (engineering change order) cycle times were shortened by 30% using Arena's cloud-based quality management system. For product and quality management processes, you can see more benchmarks in these customer stories. Let's look at how you might build a financial case for this business need.

Identify Processes to Improve

Automating ECOs, like Accuryn Medical did, impacts business in three ways.

1. Increased revenue: faster ECOs means faster new products and features to market. The better the product—the more products you can expect to sell.
2. Decreased costs: faster ECO processes means correcting product or quality problems faster. Often, products built with errors or other issues cannot be sold, which often results in recalls, mandatory rework, or scrapped product. This expense is a direct bottom-line hit. So, finding and fixing product problems faster reduces that cost.
3. Ability to scale: more product types or increased production volume often requires more employees and partners to collaborate throughout ECO review processes. The ability to get everyone's feedback and input into the change without slowing down approvals is key to meet demand.

Target Improvement Metrics

- Expected: looking at the vendors' case studies (such as Arena's), improving ECO cycle time by 30% is the average improvement.
- Best case: given that the team uses email and spreadsheets now, with a lot of manual inputs and serial signoffs, we think best case is 45% improvement.
- Worst case: we recently re-engineered our ECO process by redesigning the form and using ad hoc tools that we already have in-house. That improved our ECO cycle time by 12%. So, we may only get 15% improvement.

Translate Into Financial Benefits

Ideally, you would model expected, best, and worst case. For the purposes of this example, we'll use worst case for target improvement metrics, which is 12% ECO cycle time improvement.

- Sales reports that they could sell 0.2% more products with 12% faster new feature releases. And, they added that a more predictable release schedule would improve that even more.
- Manufacturing reports that they could reduce scrap and rework costs by 7% for 12% faster ECOs.
- Engineering can speed approvals of cost-reduction ECOs in their backlog that represent $80,000 in cost of goods sold.
- Both engineering and manufacturing agree that an online, connected system for ECOs would better support the planned growth to develop, test, and ramp production of new and improved products.

Return on Investment (ROI)

For our scenario, assume the favored solution costs $95K annually, plus $15K startup costs in the first year.

ROI = (Gain from Investment – Cost of Investment) / Cost of Investment

3.14 months = (456,000 – 110,000) / 110,000

If the numbers above are valid, you will easily make that investment back within a year—in fact, in less than four months, and your cumulative ROI over five years will be 444%. You can calculate out your yearly costs against continued investment gains and build a sophisticated ROI model if necessary with this template.

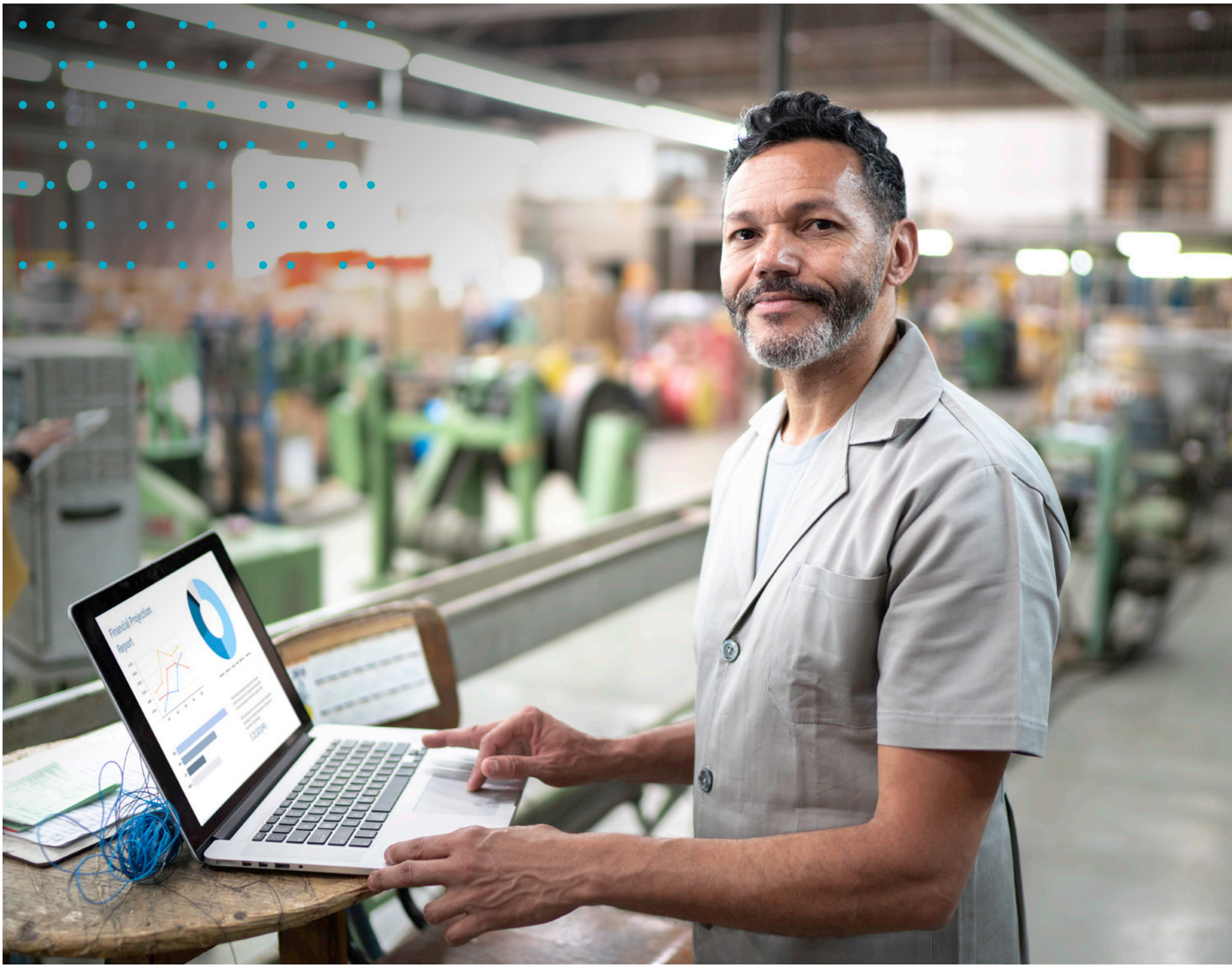
Meet With Finance

You'll want to validate your findings with the finance team. Share your assumptions, perceived benefits, and justification. Finance should be able to check the company metrics as well as provide any other details you missed. In addition, this discussion provides an opportunity to prepare finance for a possible purchase and for you to verify any additional information finance will need for approvals.

CHAPTER ACTIVITIES SUMMARY

☑ Calculate ROI for a solution

☑ Review ROI with finance



CHAPTER 6: SELECTING THE WINNING SOLUTION

So, Where Are We Now?

At this point of the software evaluation and selection process, you've:

- Established the compelling business need to search for a new solution and received executive agreement
- Come up with an evaluation plan that includes the team, requirements, research, urgency, and decision-making approaches
- Captured, classified, and prioritized the detailed requirements that meet the all the teams' needs
- Engaged and evaluated several vendor offerings
- Built and presented a financial case for a software solution

Now you're ready to select the desired solution!

What You Need to Know

It's likely that you and your entire project team are swimming in information and trying to keep everything straight. It's common to see that some team members will remember things differently or maybe remember more about the last vendor(s) they evaluated than the earlier ones.

However, if you've stuck to the plan, you've been able to capture the requirements and how each vendor meets those. It's important to keep team members moving forward throughout the process to prevent memory loss regarding how each vendor meets your overall needs.

Before selecting the winning solution, the team should keep the following in mind:

- **Nirvana doesn't exist.** There is no perfect decision or software solution. You should consider finding the solution that meets 70-80% of your needs and offers shorter time to value (TTV) and the lowest total cost of ownership (TCO
- **The bigger your decision-making committee—the harder it is to come to a quick consensus**. Not all committees are the same, nor do they act as a perfect democracy. Some members may have elevated titles or power and push their opinions forcefully. Experienced salespeople know this and will do their best to cater to and ensure the decision-makers and influencers get their questions met and needs taken care of during the sales process. So, be sure to get any key issues, questions, or requirements addressed—even if they come from someone with less influence or power.

> **70% of the time, salespeople only need to convince one person in a buying committee: the dominant influencer.**

- **Understand the cost of doing nothing.** If your team is having a hard time making a decision, consider the impact of continuing with existing systems and processes. What is the cost of doing things more manually, using tribal knowledge, or using disconnected systems between distributed teams? Be sure to capture and save all of the project information in case your team or the final decision-maker decides not to select a solution at this point in time.

  We have created a simple way to consider the cost of common mistakes for manufacturing companies to help your team understand the ongoing cost should you elect not to buy any software. See our "What Does It Cost?" calculator and plug in your company's unique dynamics and select from a variety of challenges to gain instant feedback on the cost associated with common mistakes in your industry.
- **The need to mitigate decision-making risk.** The risk tolerance of your team will vary by role and department. Each team member is likely to have preconceived ideas about working with salespeople and vendors. Some will have a favorable view and be more trusting of sales teams. Others will be skeptical or question a lot more to poke holes in vendor solutions or claims. This will make it harder to come to an objective decision, so stick to your guns and follow the plan established in Chapter 2. Don't overthink the selection process and create an overly complex justification document because the committee is worried about fear, uncertainty, and risk.

Time to Evaluate the Vendors

You have conducted an organized, systematic, and well-thought-out plan to select the best solution for your company. Using the requirements guidance in Chapter 3, you can compare and contrast what you captured for each vendor. We recommend using a simple requirement rating scale of 1-7, with 1 being best. Likewise, we recommend that 1-7 weighting be applied to all the requirements since not all requirements are equal nor should they be given equal weight.

In addition to rating and comparing your specific business requirements, we recommend comparing more areas:

- Does the vendor offer a true cloud-based software-as-a-service (SaaS) solution (or is it hosting of a client/server solution)?
- Is there a significant learning curve or custom coding required to meet your requirements?
- Did the vendor's customer references offer both pros and cons or was it simply one-sided and everything was great? Credible references will share the good and areas of improvement as well.
- Are you convinced the software solution can scale as your company grows and your needs evolve?
- How long has the vendor been in business?
- How many customers do they have and how many in your specific industry?

INSIDER TIP

HAVE OPTIONS READY

While you should present a single vendor choice, we don't recommend notifying the other vendors of any decision until your decision-makers have approved your selection. It's always possible that the decision-makers may either not approve your team's selection or they may ask for a consideration of other vendors based on pricing or other factors.

Presenting the Recommended Software Solution

With all the hard work behind you, you want to make sure the information you present is clearly articulated and compelling. Depending on your executive sponsor's direct involvement throughout the process, h/she may be ready to make the final presentation to the executive team. If not, your pitch may be to

an executive sponsor before you have the green light to present to the ultimate decision-makers. When the final pitch is made, be sure to have the most compelling and influential member(s) of your team present to increase the likelihood your selection will be approved.

Present Your Case for Executive Approval

Here is your "consultative" sales pitch: the identified problem and its impact, then a solution along with its results. The presentation will address hard dollar as well as intangible, even emotional, benefits.

In the project plan template, you'll find a template presentation based on our example scenario.

When presenting, remember:

- ☐ Don't overwhelm with eye charts and a ton of slides/documents
- ☐ Your job is to assimilate the software evaluation simply and explain your team's selection
- ☐ Don't sugar coat anything you have concerns about if there are some minor risk points—it's best to be transparent should issues arise later (then you've covered yourselves)
- ☐ Be prepared to dive into details and documents that support your decision during or after the presentation
- ☐ Ask for input, thoughts, and timeframe to make a final decision. Your decision-makers will likely not be shy about asking hard questions, but close by asking if there is anything else they need to know before making a decision.
- ☐ Ask for approval. Close by thanking the entire evaluation team and decision-makers for their time. Ask when they believe they can provide a final decision.

Here is an outline of presentation components with a suggested order.

EXECUTIVE PRESENTATION OUTLINE

Slide #	Topic
1	Introduction
2	Business Needs
3-6	Critical Business Processes
7	Financial Impact and Supporting Information
8	Other Cost Savings
9	Intangible Benefits
10	Cost Range

What's Next?

1. Once you have a final decision—hopefully to move forward with your vendor of choice—you should notify each vendor. Each vendor is likely to have invested tens or even hundreds of hours during the sales process and it is a sign of respect to let them know as soon as a decision is made.
2. You don't have to be very specific about the rationale for your team's decision but be prepared for the vendor to want as much specific feedback as possible, so they can share with their bosses and executives why they didn't win your business and how they might improve their sales approach, solution, or pricing in the future.
3. Share the decision internally. It's a good idea to share the news about vendor selection and allow various impacted departments to understand why the vendor was selected and the goals for the new solution. Be sure to leverage your committee members appropriately to help gain acceptance and gather support for the decision.
4. Finally, it's time to begin the transition to implementation phase. You'll want to work with the selection committee to establish the implementation team players (see next chapter). Those players should be well-informed about the decision-making process so they can head into the implementation project with a positive approach and not feel like a decision, or solution, was handed to them without a clear understanding of the reasons and benefits.

CHAPTER ACTIVITIES SUMMARY

☑ What you need to know (about software selection)

☑ Evaluating vendors

☑ Presenting the recommended solution

☑ Setting the stage for successful deployment

CHAPTER 7: PREPARING FOR DEPLOYMENT SUCCESS

Deployment Approach

1. **All at once**
   For less expansive tools that touch only one team or process, you might implement all/most of the solution at one time. If you are implementing an enterprise system like Arena for PLM/QMS processes, you may not want to take this approach—even if you are a younger company—particularly if you are growing rapidly or tackling important milestones such as new product launches, market expansions, or regulatory approvals, you should be sensitive to trying to do too much too soon.
2. **Phases—the how-to-run-a-marathon-successfully plan**
   For enterprise platforms that many teams will use for multiple processes, you will most likely use a phased approach to implementation.

INSIDER TIP

WALK BEFORE YOU RUN

You want early and continuous wins. If we take the marathon metaphor, training for a marathon means building a base of smaller successful runs that condition you for the longer or harder parts of the plan. In enterprise software deployments, a sensible phased plan allows you to demonstrate success for the teams (particularly if you have any detractors) and build momentum. This may mean not selecting the biggest business need to solve first with the solution. Often the critical business needs are important because they are complex and can involve many people and more data.

Phased Deployment Approach Options

If you take a phased approach to implementation, you'll be deciding which tasks to accomplish in each phase. There is no one right way. Much like training for a marathon, you want each phase to have visible wins for your team—both to ease the process pains as well as demonstrate the solution is working. In determining what each phase will include, you want to think about the uniqueness of your company, culture, organizational structure, and other corporate initiatives.

In deciding on phases, consider the following:

- Business needs: the problem or opportunity areas you've identified that are driving the purchase
- Data flows: if the software of choice integrates or shares data with other systems, you should include these data flows in your phase planning. Some data flows may be necessary to support business processes while others can be done in later phases for additive value.

Team

- Start with the selection team but re-evaluate/expand. Don't be afraid to re-assess and ask team leads for the right people once you have planned your deployment.
- Cover all the team roles and responsibilities. You'll typically need people on the implementation team to cover the following aspects: processes, data, administration. You may also need people for integrations and technical work depending upon your solution (anything on-premises has a much higher technical lift and ongoing maintenance).

Get the details: For roles you'll need to cover with Arena, see the project plan template.

Communication Plan—5 Cs

Change is hard. You are deploying a software that will require change. Following these basics of communication will reduce confusion and risks.

1. Consistent: set up a regular method and timing for project communications to help everyone stay informed.
2. Clear (and concise): create a format for project communications and re-use (another win for consistency). The format should encourage quick reading and easy writing. Consider clear sections for progress, next actions, and risks that need to be addressed.
3. Congruent (conform): fit the message to the audience. Remember to include executive sponsors and vendor representatives on communications at the appropriate level.
4. Comment: communication is two-way. Give people easy ways to provide feedback or updates, whether through meetings or asynchronous communication, preferably in a self-documenting project space such as a Trello board.
5. Chronicle the communications: archive all project communications in a central location that is accessible to the team. People can catch up if they miss communications due to absences and you also create a running history of the project. The solution you are implementing may itself be a place to store such information (indeed, Arena runs implementations through the Arena project functionality in the platform).

Risk Mitigation

An often-overlooked deployment planning activity is risk assessment. The purpose of risk assessment is to mitigate risks as much as possible. The team should proactively discuss risks to project success overall as well as to individual tasks or milestones and then consider what steps should be taken to reduce or remove risks. Risk assessment steps should include:

- Risk identification: articulate the risks. Think of the "what ifs" that could derail the project. For software deployments, common risks include a loss of team resources for other projects, lack of adoption, a technical challenge, and unexpected platform limitations.
- Threat level of each risk: once you have each risk identified, determine the threat level to the project. This can be a simple three-level risk system such as low, medium, high. You aren't going to completely ignore low risks, but you will have a different response to low-level risks vs high-level risks.
- Possible mitigation of risks: for each risk, determine how you may mitigate it—steps to completely remove the risk or reduce its impact. Mitigation might include proactive steps (such as additional communication sessions with the extended team to ensure adoption) for a high risk or simply awareness of a risk and recognition that if the risk occurs, the team will then assess for response.

Once your risk assessment is complete, include risk mitigation in project planning and communication as appropriate. Also be prepared to revisit risks as the project progresses.

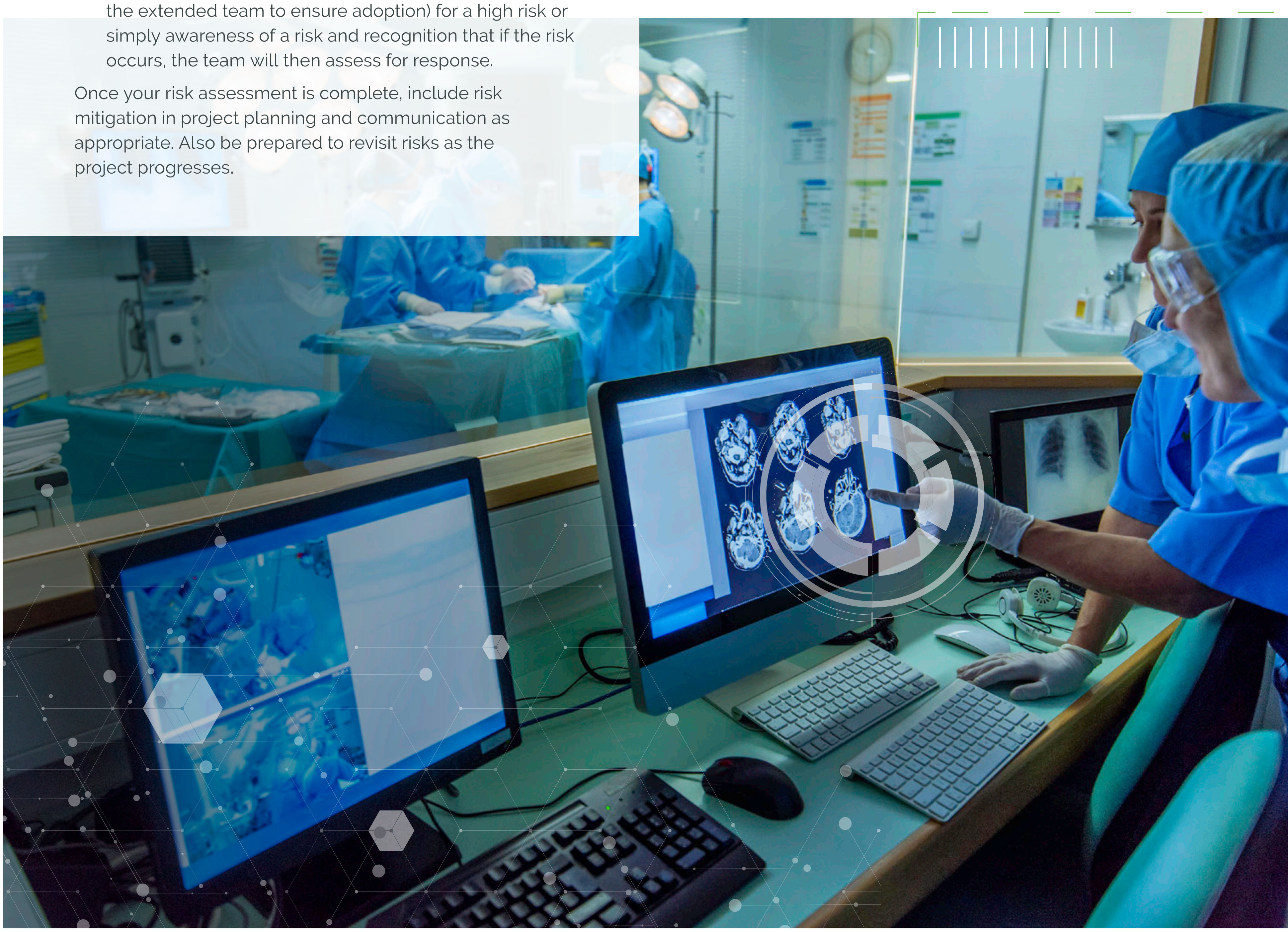
CHAPTER ACTIVITIES SUMMARY

☑ Determine your deployment approach

☑ Assemble the team and get educated

☑ Set up your communication plan

☑ Complete initial risk assessment



CHAPTER 8: MEASURING SUCCESS AND ON TO THE NEXT

You've come a long way. But you're not there yet.

Now that you're using your new solution, understand how your project results compare to expectations. Measure now and continually to know outcomes and support continuous improvement. Here are four reasons to measure:

1. Inform your decisions to continue or change implementation. If the results are good and in line with expectations, that's great. But, even if the results are not what you expected, you are armed with this knowledge, so you can find out why and correct that issue.
2. Improve user buy-in. New software and redesigned processes inevitably change team members' responsibilities. Users are more willing to make the changes when they understand the roles they are playing to get better results.
3. Secure ongoing support for the project. Software has annual subscriptions or support fees. Stakeholders who understand why they are using the software approve ongoing costs.
4. Remind the executives who approved this investment how the software has helped meet corporate goals and their own needs.

Being able to measure and communicate the success of the project is good for your career, too.

When Should You Measure?

Take these measurements at regular intervals to review performance. This gives you a realistic snapshot and, over time, a trend for basing projections into the future. Today, you can use dashboards and analytics to understand cycle times, throughput, and other measurements. If these are not available, you can schedule reports and data extracts so you can analyze performance.

In the early weeks of go-live, things can be unstable or get worse before they get better. Measure and stay in touch with users as this system is becoming the norm, so you can address issues during this transition time. But it's too early to make any conclusions about whether the system is a success or not.

Process Improvements

The financial benefits are usually a result of improved processes. Use the process metrics (like faster cycle time, expanded throughput, or smaller backlog) you identified in the business case before implementation as a performance baseline. If for some reason they are not accurate or current, measure the processes you intend to improve.

Revisit Your Financial Case

Earlier in this process, a financial case helped you get approval for this software. The financial impacts you predicted are those that resonated with the executive sponsor and other stakeholders. The people who are part of the approval process, as well as new executives, will be interested in how the solution is delivering on those goals.

- Check and refine the assumptions and estimations you used for the financial case
- Work with finance to understand the business results that your project impacted
- Include improvements that you omitted from the original financial case, when appropriate

Get User Input

Interview and survey your users. Users range from power users to those that use the output from the system. Also consider whether partners and customers apply.

These exercises can reveal positive results as well as weaknesses in the implementation. You might find, for example, that people need a refresher training class or a "tip sheet" after go-live.

Our experience has been that user input can help in two ways:

- Suggestions are usually concrete steps you can take, benefiting everyone
- Users feel empowered when they contribute to a system's success

Proactively schedule interviews with individuals and small groups about the software. Seek out the influencers in your user group as well as other users. Find out what is working and not. Ask both open and closed questions. Listen and ask follow-up questions. Recording the conversation takes some pressure off you so you can be present with the interviewee.

Surveys are easy these days. If you don't have a corporate standard survey tool (check with HR), you can use one of many with free versions, including SurveyMonkey, Qualtrics, and SoGoSurvey. You can ask general questions, like "How satisfied are you with ______?" or specifics about the functionality, their confidence, and results. Include text, yes/no, and numerical answer types. Like the financial measurements, you can do these throughout the year and derive trends.

External Factors

Note things outside your control that may influence your results. Mergers and acquisitions, changes in priorities, partner issues, even pandemics and climate crises can impact performance.

Note how your software has helped you lower business risk by adapting to new situations.

Executive Presentation or Report

As you did with getting approval for the software, you now need to present the results to your executive sponsors and stakeholders. The Measuring Success presentation in the project template will help get you started.

Maintaining Momentum

Armed with your success measurements, you may be ready to look for ways to get more value from the software. Work with your customer coach, as well, to find out how you can expand beyond your initial implementation. You may decide to add more business units, processes, users, or modules in the next phase of your implementation.

CHAPTER ACTIVITIES SUMMARY

☑ Determine your metrics, frequency of measurements, and tools used to measure

☑ Measure!

CHAPTER 9: SUCCESSION PLANNING

If you have made it this far in the guide, you might wonder what else you need to know. If you've successfully selected and implemented a software solution to meet your business needs as well as made a plan for continuous improvement and ongoing success, you and your team have done a lot. There is only one important thing left to do: Plan for the inevitable change in ownership of the business processes and, therefore, supporting software. You and others who have been involved up to this point will not be in these same roles forever. Succession planning is a pay-it-forward move you can do to make the team and company stronger in the future.

The good news is that if you have followed the process outlined in this guide, you've documented much of the background information that would be part of a transition package.

Here is an outline of recommended sections:

Project Summary and History

- Business Needs Statement
- Executive Presentation for Decision
- Deployment Plan

Current State

- Current contracts, any vendor commitments, your commitments for any marketing or customer advocacy, and vendor team contact information including roles
- Use cases and SOPs related to the solution
- Team list of business owners, stakeholders, administrators, and user groups (both internal and any external)
- Data flows and integrations in place

Unfinished and Future

- In-progress phases
- Unresolved challenges
- Future expansion promises or plans for the software

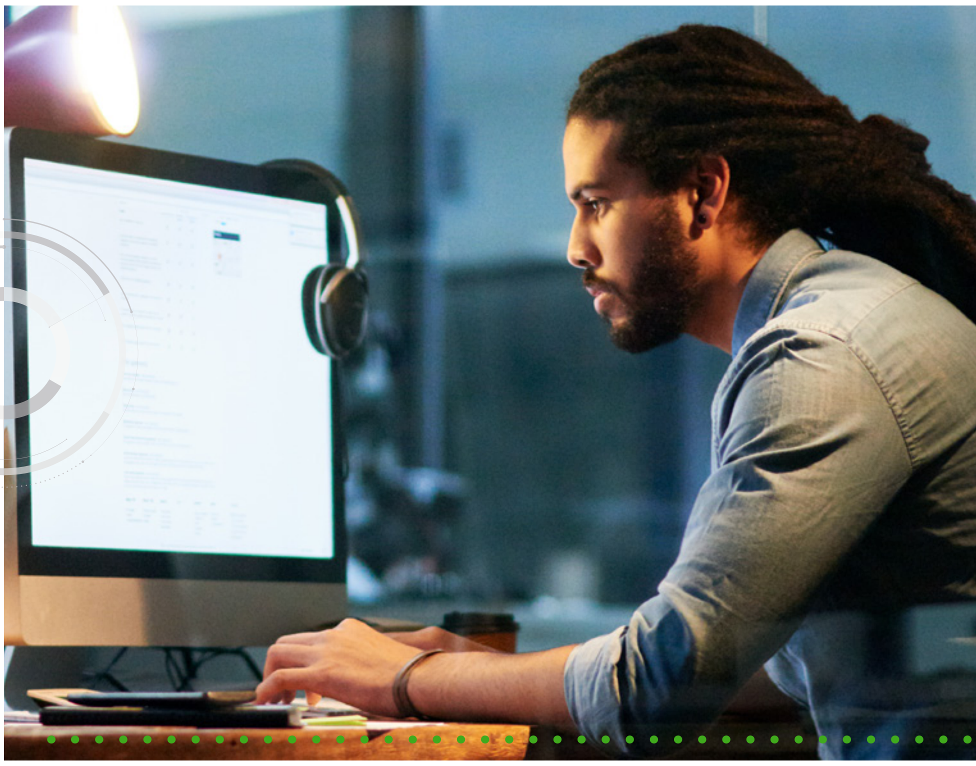
With this transition package, you can onboard any new leaders to the project, regardless of timing or role. If you involved others in putting together and maintaining this information, succession planning is easy and lets everyone know that the company is committed to the software critical to your teams doing better work.

CHAPTER ACTIVITIES SUMMARY

☑ Create a transition package—
make sure others in the team participate

CHAPTER 10: GET STARTED NOW

This is the non-summary summary to the guide. You don't need a few paragraphs on what we've covered, you need an action plan. Or, if you skipped all the details, you still need an action plan. Here is the plan to get you going in your hunt for a software solution that solves your business needs.



1 Identify Your Business Needs

- Calculate the costs of these pains
- Agree now is the time to do this work better
- Create a plan for how you'll evaluate—and have at least one executive sponsor

2 Find a Scalable Solution

- Ensure inclusive participation of stakeholders and collect requirements
- Do proper due diligence on vendors
- Do the value assessment
- Present findings and choose the best solution

3 Deploy the Solution

- Get participation from all teams significantly involved as users
- Determine your deployment approach
- Educate users

4 Ensure Adoption and Success

- Analyze and optimize the software
- Cultivate a culture of continuous improvement to scale solution with business and new functionality provided by the vendor

5 Create a Practice of Continuous Improvement

- Ensure the project endures beyond any individuals on the team by documenting where you are, what has been accomplished, and future plans

SEE ARENA IN ACTION

❯ EVALUATE ARENA

Additional Resources for PLM/QMS Software Projects

- The Software Buy Project Plan: the template project plan with associated resources that match this buy guide. Built in Trello, it's easy to make a copy for your project and share with your team.
- What Does It Cost?: interactive calculator to help you see how much process mistakes cost product companies
- 7 Principles of Product-Centric Quality Management
- Intelligent BOM Management for Product Innovators

ptc arena

121 Seaport Blvd, Boston, MA 02210 : ptc.com