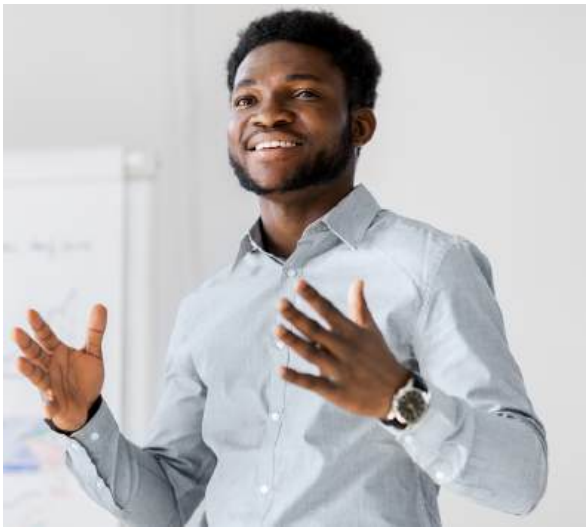
pmsquare
simplifying complexity

10 PROVEN STEPS TO IMPLEMENTING PLANNING SOLUTIONS THAT WORK

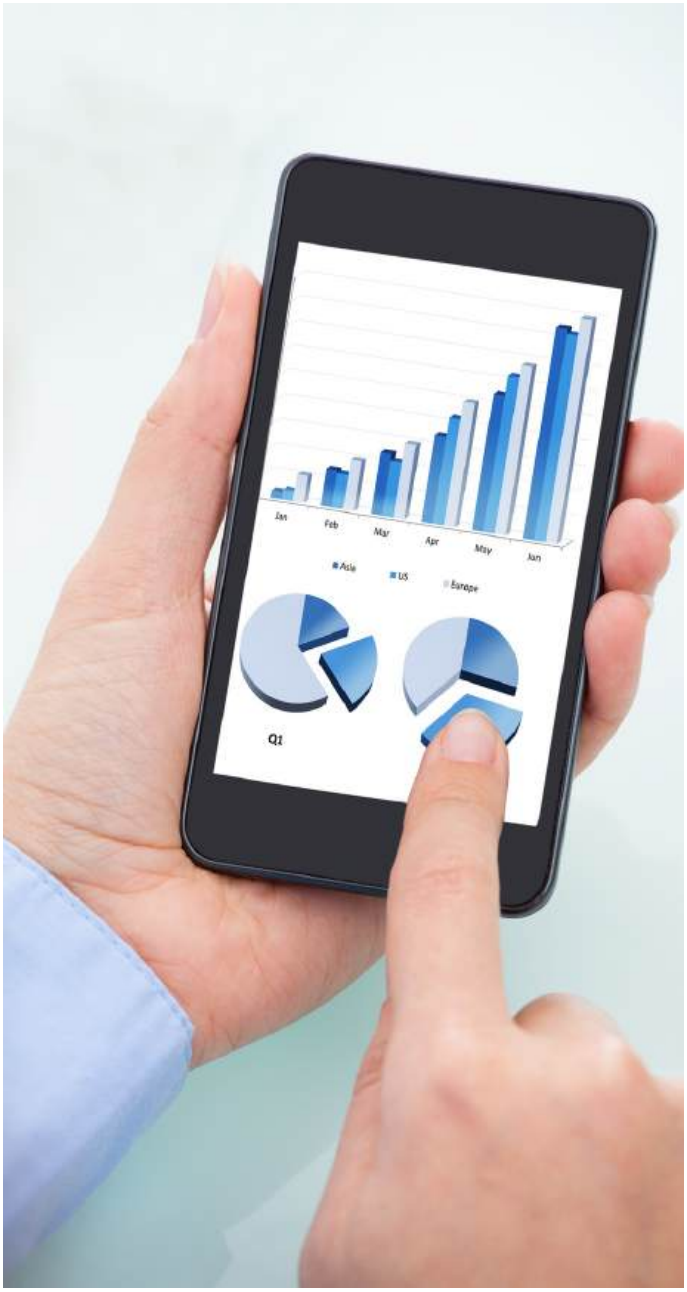
www.pmsquare.com/planning-ebook



In today's fast-paced business world, it is more important than ever for organizations to have a solid understanding of their data and to be able to analyze it effectively. As the amount of data available continues to grow exponentially, so does the need for planning solutions that can help businesses make informed decisions about their operations, finances, and overall strategy.

In this eBook, we'll be taking a deep dive into the world of planning solutions, exploring the essential steps that customers should be aware of before implementing such solutions. Our goal is to provide readers with a comprehensive understanding of planning process, equipping them with the knowledge they need to implement effective solutions in their own organizations. Whether you're a seasoned data analyst or just getting started with analytics, this eBook is designed to offer valuable insights and practical advice that you can put to use right away.

By the end, you'll have a solid grasp of what it takes to implement a successful solution in your own organization. Whether you're looking to optimize your supply chain, improve your financial forecasting, or gain insights into customer behavior, planning solutions can help you achieve your goals and drive business success.



Introduction

Companies have been leveraging spreadsheets to prepare budgets since shortly after the dawn of the computing age. Spreadsheets are a useful tool that are universally understood, easy to use, and provide a great amount of flexibility.

However, all growing businesses will eventually outgrow the capabilities of spreadsheets, and a larger, more automated and scalable solution will be required. But what does this foundational change even look like? Where does one start? Who should one call? What will it all cost? How many team members will be needed? This comprehensive 10 step guide has been prepared to address these questions and provide a full overview of what to expect when implementing a Planning Solution.

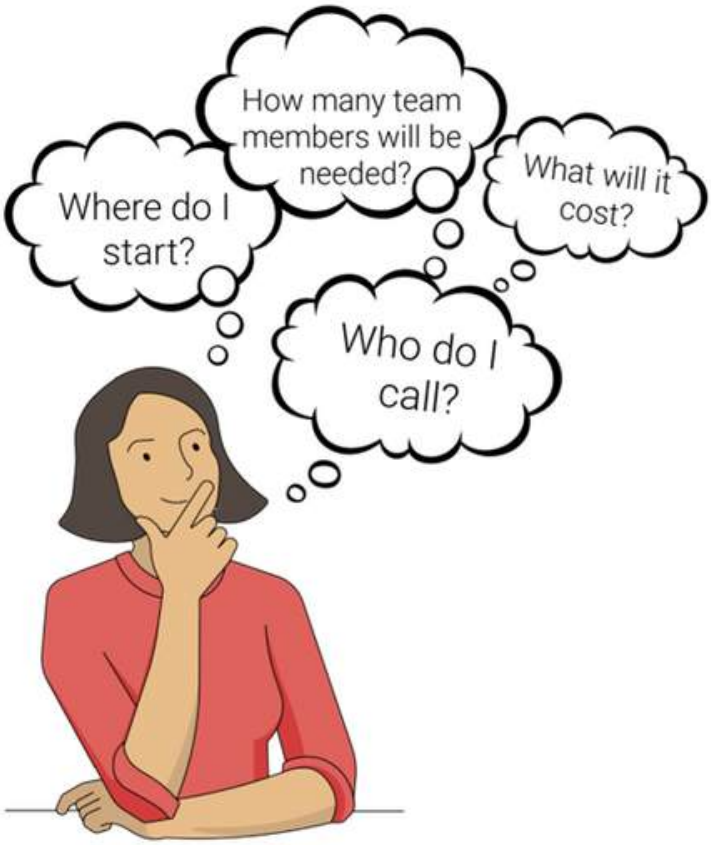
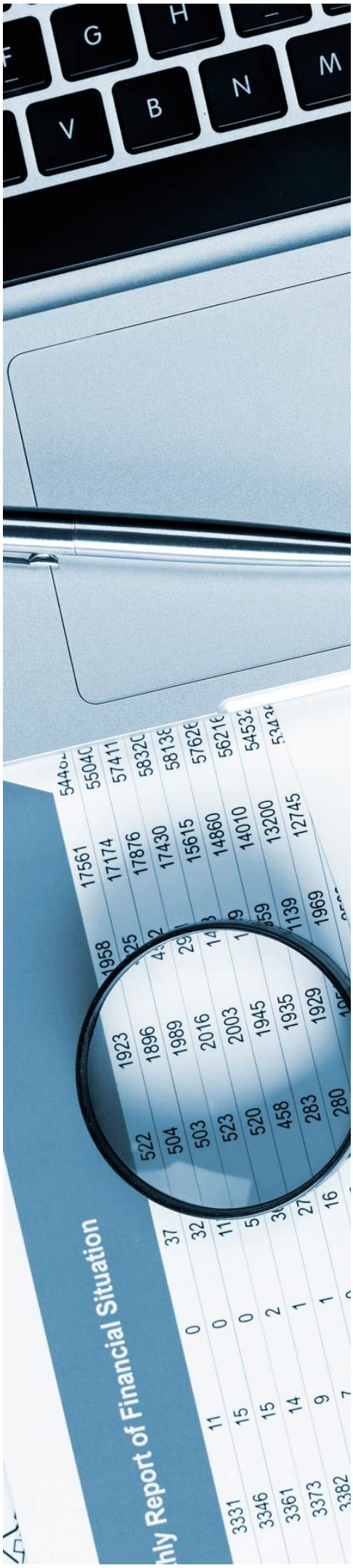


Table of Content



Requirements Analysis

In the first step of the journey towards improved FP&A, it is of utmost importance to look inward. What are the current business processes around Budgeting and Forecasting? What teams and what programs are in use? What's working and what isn't? Where are the pain points and inefficiencies? Who will be championing the cause towards a better solution? Who will serve on the project team?

As with most large endeavors, when implementing a planning solution, it is often best to start with the end in mind. Decide on the desired deliverables that the new software will provide. Review the current reporting book and budget templates to help organize some of the basics of the current processes.

Keep an eye out for areas of the planning cycle that are causing headaches and brainstorm some efficiencies that could be introduced. In addition to reproducing the existing infrastructure, it is also critical to include a discussion about 'blue sky' enhancements and the art of the possible.

> **When implementing a planning solution, it is often best to start with the end in mind.**

Items like automation, batched reporting, and predictive forecasting via statistical algorithms are common improvements to consider alongside the status quo business tasks.

Adopting a new planning solution can also provide a great opportunity for consolidation and integration of several systems. Now is the time to investigate all the different information systems currently in use and explore possibilities where a new tool could replace 2 or 3 of the incumbents. Less software applications will yield savings in licensing, staff training, administration, avoidance of compatibility/technical issues, and therefore lower total cost of ownership.

Another preliminary aspect to focus on is the project team; which members of the organization are interested, capable, and available to play a large role in the implementation project. Depending on the size of the company and the scope of the project, it may be necessary to have 1-3 resources from the office of finance committed 20-30 hours per week from project kickoff to Go-Live.

Some customers will elect to make a new hire(s) specifically to focus on implementing and administering the new planning application. Some additional roles on the project team include the sponsor/owner, project champion, IT liaison and change manager. The sponsor/ owner is ultimately in charge of making all final decisions (vendor selection, implementation team, project resources, financial approvals, change orders, etc.).

The champion is the advocate for implementing the new technology and motivating the organization to pursue this new initiative. The project champion will work alongside the change manager to socialize the project and upcoming changes to business processes and inspire confidence that the new application will improve the functions and operations of the business as a whole.

Finally, a representative from the IT department will be needed to set up the back-end infrastructure, perform the installation, assess 3rd party software requirements and compatibility, and administer security of the application.

Once a thorough review of the internal environment, resources, and processes has been completed, it is now time to look outward and review the product landscape.

Due Diligence

How does a company perform due diligence in selecting the right platform for a planning solution? Start by looking around the marketplace for products that meet the company's requirements.

Define what must be accomplished on this new platform. Does each product meet the company's baseline planning requirements?

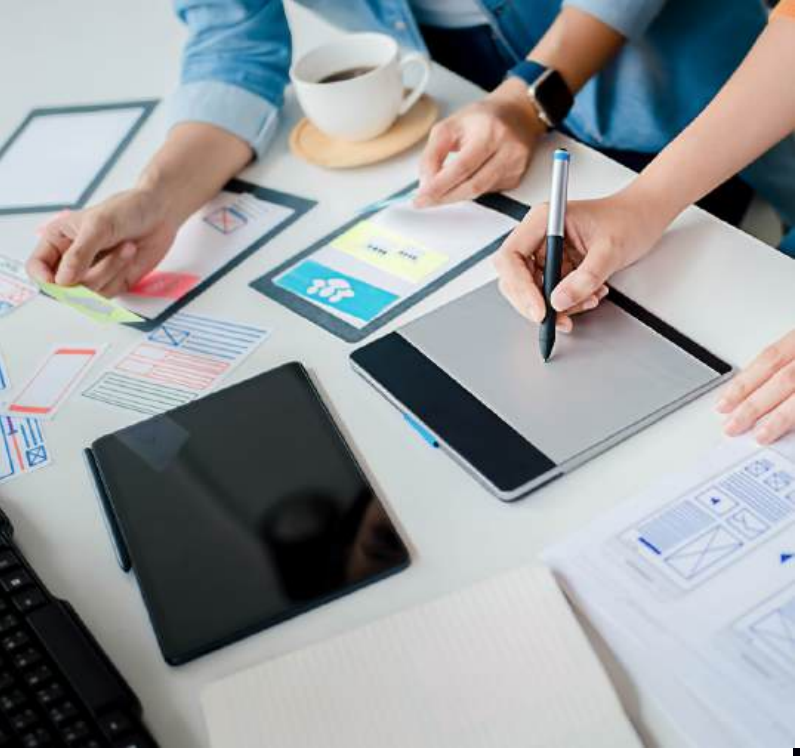
Can it create a place to make it easier for users to contribute to the plan? How can it decrease the time it takes to collect plans and consolidate them across the company? Can it automate any repetitive manual tasks that occur during the planning cycle? Is AI forecasting a capability the company might want to use?

Create a checklist to compare the features and benefits of each software platform to ensure that a particular product checks all the boxes for current and future needs. It is important to think beyond how the planning process is managed today. That will help ensure the solution can be extended down the road and that the product will support future requirements.

Reach out internally within your company and through professional networks for recommendations. Discuss the challenges that arose trying to implement a particular planning solution. Understand what went well and what challenges they faced. Were the issues due to limitations of the platform or some other reason?

Consider if the platform is to be deployed on-cloud or hosted on-premise. If hosted on-premise, does the team have the expertise to keep the product components updated to the latest versions and running with high availability and minimum downtime?

If hosted on-cloud, will the software vendor host your application on their cloud utilizing their Software as a Service (SaaS) offering, or will it be hosted on the company's preferred cloud provider? These are a few things to consider when deciding how and where to deploy the platform.



Deciding on a platform to develop a planning application is an expensive decision that has costs beyond the software or user licenses. The investment will include initial development, testing, and go-live.

Down the road, there will be ongoing maintenance that includes code fixes due to business changes and functionality enhancements to extend the model.

Choosing the right platform will reduce costs in the long term by standardizing a platform early in the process.

It is important to engage with a vendor who supports the product by providing new features along with bug and security fixes.

After conducting thorough due diligence on the product landscape, it is time to move forward with vendor selection.

Vendor Selection

Choosing the appropriate software vendor is a crucial step for any organization that seeks to implement a planning solution.

Vendors offer different levels of functionality, each with unique strengths and weaknesses, and selecting the wrong one can result in significant consequences, such as wasted time and money, and an ineffective solution. To prevent such outcomes, it is essential to conduct objective research and evaluate potential vendors.

When evaluating vendors, several criteria must be considered:

- Functionality
- Scalability
- Integration
- Support
- Cost
- Ease of Use
- Vendor Track
- Record
- Free Trial

You can read more about these criteria on the following page.

FUNCTIONALITY

Functionality should be the primary factor, and the software's capabilities must align with the identified business needs and requirements. This could include anything from data integration capabilities, infrastructure requirements (traditional on-premises installations vs SaaS offerings vs own third-party cloud), to advanced analytics and reporting. Consider features such as budgeting, forecasting, reporting, scenario analysis, and what-if analysis.

COST

Evaluate the cost of the software and compare it with other solutions on the market. Consider the total cost of ownership, including licensing fees, implementation costs, and ongoing support and maintenance costs. This will help narrow down options and focus on vendors' solutions that fit within a budget.

SCALABILITY

Consider the scalability of the vendor's solution. Make sure that the solution can grow and adapt to business needs in the future. Consider factors such as the number of users, the volume of data, and the complexity of business processes.

EASE OF USE

Ensure that the software is user-friendly and easy to navigate. This will ensure quick and efficient financial planning and analysis. Choosing an easy-to-use software also increases user adoption, reduces training costs and time, and ultimately leads to a more successful implementation of the software.

INTEGRATION

Evaluate the vendor's integration capabilities. Make sure that the solution can integrate with existing systems, such as ERP, CRM, or HR systems and other data sources such as text files and Excel spreadsheets. This will enable data to be transferred seamlessly between systems and ensure a complete view of the business.

VENDOR TRACK RECORD

Evaluate the vendor and vendor software track record compared to other vendors in the same market by looking at reports and reviews produced by impartial entities like Gartner.

SUPPORT

Evaluate the vendor's support offerings. Make sure that the vendor provides adequate support to ensure a smooth implementation and that the vendor provides ongoing support to address any issues that may arise.

FREE TRIAL

Sign-up for a free trial of the software, if available, and explore the software firsthand. By following these guidelines, organizations can select a vendor whose solution aligns with their business needs and requirements.

Vetting Implementation Partners

Sometimes, in combination with the vendor selection process, it is essential to find the right implementation partner. When searching for an implementation partner to guide your organization through the process of adopting new financial planning software, it is crucial to carefully consider several factors.

Begin by assessing the partner's track record in delivering successful implementations of the software under consideration. Look for online reviews, testimonials, and case studies that provide insights into the partner's reputation and ability to deliver in industries similar to your organization.

Investigate the partner's employees and assess their experience and qualifications, including certifications demonstrating their proficiency in software and related technologies.

Determine if the partner shares the value of your organization's values, goals, and approach to business, thus prioritizing long-term relationships and customer success.

To obtain a more in-depth understanding of the partner's capabilities, ask for contactable references and follow up on them.

Contacting references can provide valuable insights into the partner's strengths and weaknesses and ability to deliver on their promises within budget and on time.

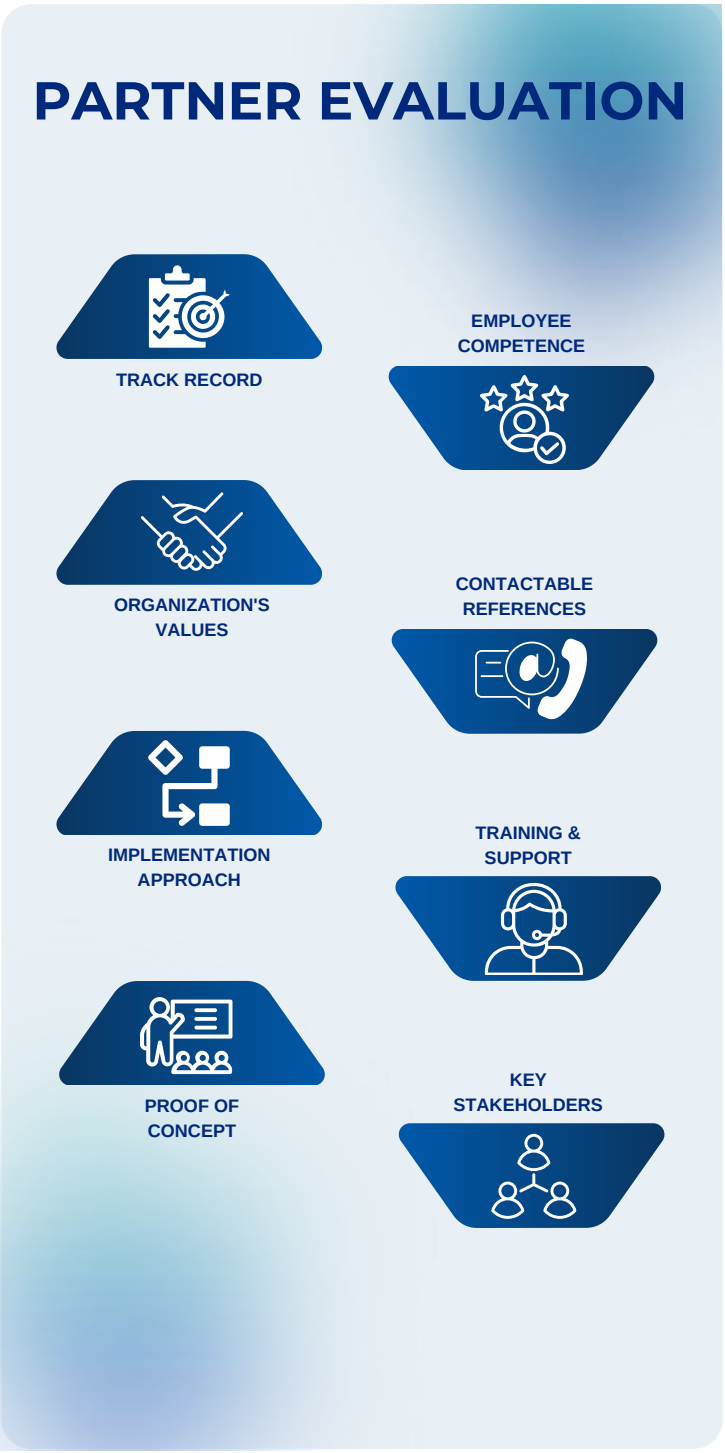
In addition, ask about the vendor's implementation and project management approach, including whether the partner followed a detailed plan with clear timelines, milestones, and responsibilities. Inquire about the partner's support and training offerings to ensure that your organization's employees receive the necessary training and support for effective use of the new technology.

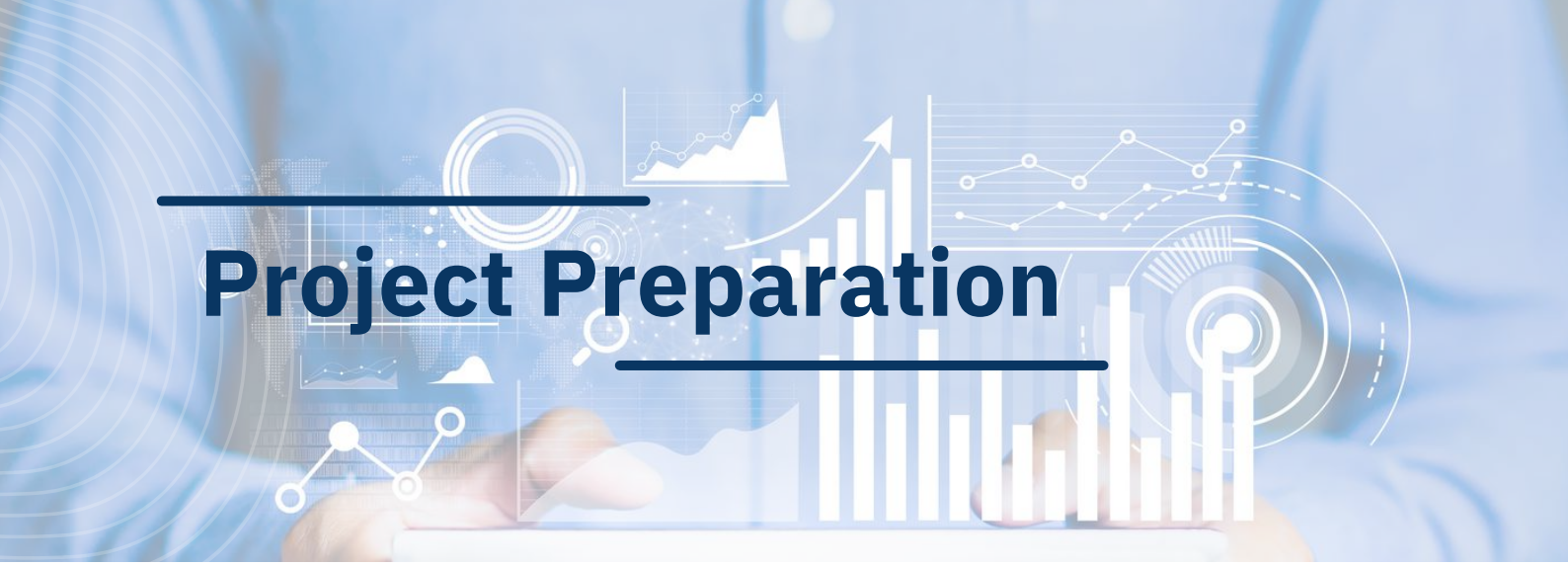
Another effective evaluation method is to request that potential partners perform a presentation of the software. Schedule presentations from several partners within a short time span, with the same selection committee and objective criteria to eliminate bias.

Request that the presentation be customized to your organization's unique circumstances, using a sample set of your organization's data. An interactive presentation allows you to ask questions and receive detailed answers that address specific concerns.

Finally, request that shortlisted partners provide a detailed proposal outlining their approach to implementing the software for your organization. Evaluate the proposals and clarify any issues related to the solution, implementation, support, or cost.

After you have selected the right partner who can help you achieve your financial planning goals, you are ready to start preparing for the project.





Project Preparation

Once the software and implementation partners have been selected, both internal and external teams will need to prepare for the project. This entails scoping activities, requirements gathering, software environment preparation, training, and most importantly data preparation.

In preparation for project kickoff, several meetings should take place with the chosen implementation partner. It is now time to flesh out a full project scope based on the initial requirements gathering exercise performed internally along with discoveries made during product demonstrations. Internal project stakeholders will meet with project managers and solution architects to assess all project deliverables, build out a project schedule and create a schematic diagram of the proposed solution.

With a project scope established, the project team can move on to selecting a methodology that will guide the project's development and implementation phases. The chosen methodology should align with the project's goals and objectives and ensure that the project is completed on time and within budget.

Agile, Waterfall, and Hybrid methodologies are common options and the team should choose the most suitable methodology based on the project's unique characteristics, such as complexity, size and customer requirements.

The project manager should also develop a communication plan that outlines how the project team and stakeholders will communicate throughout the project's lifecycle, including regular progress updates, issue resolution, and change management. The project plan should also include a risk management strategy that outlines potential risks and how they will be mitigated.

Next, it is advisable that internal team members undergo training on the new application so that everyone understands the basics and terminology of the newly acquired software.

This cursory level knowledge of the product helps guide scoping discussions so that software architects can effectively communicate the capabilities of the product, and determine what functionality is feasible versus what items are not advisable.

If the selected software will be installed on-premise (a traditional software installation utilizing the servers owned by the customer), team members from IT will need to connect with the software vendor to setup user accounts, receive credentials, and download the software.

Servers will need to be provisioned with proper operating systems, connectivity, and third-party software. Once these steps are completed, software installation can begin. This may require assistance from the software vendor and/or developers from the implementation partner.

If a Cloud solution is chosen, most of these steps **aren't needed** because the vendor's technology team will host and maintain the software, however it is important that the vendor provisions an instance on their Cloud for the client and confirms availability before project kickoff.

Lastly, the customer will need to conduct a full analysis of source data that will be used by the new application. Typically planning solutions incorporate actuals data sets from finance, sales, operations, and HR.

Corporate ERP systems help with the organization and availability of these datasets from one central location, and their presence can make the implementation of a planning solution much simpler.

In the absence of a mature ERP system, there are some instances in which a data warehouse should be delivered before the planning project can begin. Some implementation partners may want to push forward with the planning project in tandem with data architecture projects or ERP implementations, but this should be avoided.

While SpaceX has shown that hitting moving targets can be exhilarating and impressive, let's stick to FP&A and leave the rocket science to Elon and his crew.

Once the existence, location and availability of source data is confirmed, it is time to review the tables, fields, formats, and records in the data sources. Before project kickoff, it is important to identify and correct any source data issues because poor data quality leads to bottlenecks, project delays, and oftentimes change orders.

After requirements gathering, project scheduling, software environment preparation, training, and source data review has been completed, it is time to formally kick off the project. The next step will provide all the details of developing the solution.

Project Design and Build

The longest step in the process is developing the solution. This entails software installation, rapid prototyping of the solution, iterative meetings to review progress and discuss revisions, and working towards delivery of the project. Solutions can be developed within existing frameworks of the chosen application, using previous templates or 'blueprints' provided by an implementation partner, or from a clean slate.

The design/build phase can begin once the requirements have been identified and documented. A design is created that outlines the architecture, data model, business rules and user interface of the application.

The design phase also involves applying the requirements to begin building a prototype of the planning model. This will include model structure, calculations and employing the statistical or financial drivers that influence budget or forecast results. Often, it's possible to use an existing model or planning process to jump-start the build phase which can save an enormous amount of time.

Using the existing planning model, no matter how rudimentary, can inform the development team on existing practices and company policies.

Typically, a small, but representative model is created to start the iterative process of application development along with stakeholder reviews of the application. It is extremely important that stakeholders are regularly involved in reviewing the model as it's developed.

Regular evaluation of the model is imperative to remove any ambiguity in the requirements. By using the evolving application to demonstrate how the model captures, transforms, calculates and reports results, stakeholders will have the opportunity to guide development toward a successful implementation.

As the build process progresses, more details are considered, and some issues may appear for the first time. Expect to have your subject matter experts invest more time to handle new issues as the application development deepens.

In parallel with the core development, the user interface development will take place. The user interface will be another tool for developers and stakeholders to assess progress and to facilitate testing.

The user interface should conform with design requirements, which should include creating a visual design that reflects the company brand and the needs of the user, while also being functional. A great user experience will accelerate user adoption.

The user interface should conform with design requirements, which should include creating a visual design that reflects the company brand and the needs of the user, while also being functional. A great user experience will accelerate user adoption.

Another essential development component is providing data from source systems, whether ERP systems, operational systems, or from direct user input.

There are inevitably some challenges when sourcing data for the developing application including writing the code to extract timely, valid data. Planning application developers will rely heavily on internal subject matter experts to extract required data.

Data from source systems will be used to build and maintain the model structure by providing information for accounts, organizational units, products, etc.

Financial and statistical data from general ledger and operational systems can populate the application with historical data. As mentioned earlier, it is important to have clean, relevant data ready and accessible before development of the model begins.

When the initial development phase is nearing completion, implementing plans for testing should begin.

TAKEAWAYS

1. **DOCUMENT AND CLARIFY REQUIREMENTS BEFORE STARTING THE DESIGN/BUILD PHASE**
2. **CREATE A COMPREHENSIVE DESIGN THAT INCLUDES ARCHITECTURE, DATA MODEL, BUSINESS RULES, AND USER INTERFACE.**
3. **DURING THE DESIGN PHASE, A PROTOTYPE OF THE PLANNING SOLUTION IS BUILT**
4. **CONSIDER UTILIZING AN EXISTING MODEL OR PLANNING PROCESS TO EXPEDITE THE BUILD PHASE**
5. **REGULARLY ENGAGE STAKEHOLDERS IN THE MODEL DEVELOPMENT PROCESS**
6. **IT IS ESSENTIAL TO PROVIDE DATA FROM SOURCE SYSTEM – IT PLAYS A CRUCIAL ROLE IN THE MODEL STRUCTURE**
7. **IT IS IMPERATIVE TO HAVE RELEVANT AND CLEAN DATA FROM SOURCE SYSTEMS TO ENSURE A ROBUST MODEL STRUCTURE**
8. **THE USER INTERFACE DEVELOPMENT WILL HAPPEN IN PARALLEL**
9. **ENSURE THE USER INTERFACE CONFORMS TO DESIGN REQUIREMENTS, BOTH VISUALLY AND FUNCTIONALLY**
10. **PRIORITIZE CREATING A GREAT USER EXPERIENCE TO ACCELERATE USER ADOPTION**

Testing the Solution

It is imperative that all projects incorporate a thorough testing plan. There are four main testing phases that every implementation must go through before the planning solution is ready for Go Live: Unit Testing, Integration Testing, System Testing and User Acceptance Testing (UAT). The first three will take place within the development step solely among the project team, and testing can be iterative as development progresses.

As the developers are creating the planning solution, **Unit Testing** will be conducted to confirm that the code is performing correctly in isolation. This differs from **Integration Testing** which tests the functionality and interoperability of the modules. These testing cycles help developers work out the kinks as they progress towards a viable solution.

Towards the end of the development phase, **System Testing** is conducted to provide a comprehensive assessment to ensure that the solution handles the requirements and deliverables as originally prescribed during the planning and design process. Any bugs or malfunctions are remediated, and additional regression testing is performed to ensure other parts of the solution aren't impacted by the fixes. Successful System Testing marks the completion of the Project Development phase of the project.

WHY DO WE TEST?

- **CONFIRMS THAT THE SOLUTION WORKS FROM A TECHNICAL PERSEPCTIVE**
- **APPROVES THAT THE APPLICATION DELIVERS THE INTENDED PROJECT OBJECTIVES**
- **EXAMINES SYSTEM PERFORMANCE**
- **ENSURES THE PRODUCT IS READY FOR GO LIVE**

At this point, the solution is migrated or cloned to a new software environment for **User Acceptance Testing (UAT)**. The use of a separate server is necessary to provide users with a testing area separate from the Development environment.

Developers still require security and autonomy over their development environment so that they are able to remediate issues discovered during UAT and regression test the model. Testing scripts are developed by the quality assurance team so that a separate group of users can objectively test all requirements and the overall functionality of the solution.

Testers follow the scripts and document any defects which are then logged and sent back to the development team for resolution.

It is advisable to use a ticket system like Click-Up, Jira or Asana to track the remediation of issues and bugs, however on smaller projects MS Excel will suffice. UAT testing validates that the planning solution completely covers the intended business use case end-to-end.

To safeguard the success of the project, it is absolutely critical that the solution functions to the expectations of the user community, and thus the importance of UAT testing before the solution can Go Live.

TESTING TYPES

01 UNIT TESTING

02 INTEGRATION TESTING

03 SYSTEM TESTING

04 USER ACCEPTEANCE TESTING (UAT)

Go Live

Go-Live is the day the switch is flipped where users are granted access to the production environment of the new solution. Having a solid plan in place on how to handle change management before going live is necessary.

Up to this point, the solution would have been thoroughly tested during the beta and UAT phases where it was rolled out to a small subset of users. This testing period was to help identify and allow the team an opportunity to fix any defects in preparation for Go-Live.

The reality is that not all defects will be found during testing and the teams will have to handle these as they come up during Go-Live.

Depending on how large of a deployment and the number of users that will be onboarded to the application, the launch could be staggered and rolled out to individual departments over a few days or weeks. This can alleviate the burden on the implementation team of having so many new users using the solution at the same time.

Leadership should communicate the benefits of the new solution and there needs to be clearly defined steps users need to take before going live and during the launch. This communication should include multiple emails that hype the upcoming launch, reminders of the launch date, and documentation on how to access the new solution.

There should be training sessions on how to interact with the tool and a knowledge base (KB) that users will be able to access. The team should continue to update the KB as things evolve and change. There should be clearly defined ways for users to request help during and after the launch.

This requires a team to actively help users navigate the application, grant access to users as needed, and handle any other requests that may come in. This support is critical to having a successful Go-Live. It can be beneficial to train power users within functional areas to help others on their team and have them funnel all questions or requests up to the support team.

Defects will be uncovered during this process and the implementation team will need to be ready to handle them during this initial period. Some defects may not be handled during the launch and will be placed into the backlog to be resolved depending on priority and severity.

> **Leadership should communicate the benefits of the new solution and there needs to be clearly defined steps users need to take before going live and during the launch.**

Users may also request enhancements or new features that can be prioritized and placed into the backlog. This can be a stressful time for everyone involved. Issues will come up during Go-Live and it is prudent to have a plan in place on how they will be handled during Hypercare to make it easier for the developers and end users to minimize frustration during the launch.

Hypercare

Hypercare begins immediately after a successful Go-Live of the new solution. This is a time to provide an elevated level of support for the solution and help new users understand how to use the product so that they can realize the greatest benefit.

By providing the type of support post-launch, you can help drive user adoption and move the needle to provide a seamless launch. This support period can last for a few weeks to a few months depending on the complexity of the solution and the frequency of the cycle in which the solution is used.

If the users only utilize the solution for a few days each month during a financial close, it may take a couple of cycles before everything is running as anticipated. Depending on the size of the deployment, the team may want to set up a war room to be a centralized place for the developers and project managers to work together with end users as issues arise.

The Hypercare team should be a limited subset of the original implementation team that is kept around to address any issues that arise from the use of the solution deployed in production. This team could be comprised of people from the software vendor, the implementation partner, project managers, and internal subject matter experts (SMEs) involved with the project.

1. **SELECT A SMALLER GROUP FROM THE INITIAL DEVELOPMENT TEAM AND VENDORS.**
2. **ESTABLISH A DEDICATED WORKSPACE TO HANDLE SIMPLE HELP REQUESTS AND TROUBLESHOOTING.**
3. **ENCOURAGE USERS TO SUBMIT TICKETS FOR ANY DEFECTS OR BUGS DISCOVERED DURING THE LAUNCH PHASE.**

The team should be able to answer questions on the spot and put out fires as they come up. For larger issues, they should have an intake system in place to handle tickets that come from the users where they can document the issue and close the ticket once the issue has been resolved.

This feedback loop is important to the success and longevity of the solution. Once the fix has been identified, developed, and tested, they can document and deploy the code fixes necessary to resolve any defects in the production application. This is the time to prioritize any defects identified during Go-Live that were not deemed critical and start working through the backlog of defects and enhancement requests.

Some of these larger enhancement requests may be held off until the sustainment phase where they can be grouped and deployed as a new release of the solution at a later date.

Hypercare is all about partnering with business users to address any issues as timely as possible. The Hypercare team should be partnering with users to solve and document issues that arise in the solution post-launch.

Be sure to fill up the knowledge base with lessons learned during the launch. The initial focus should be to stabilize the production environment and provide user training. Once the production environment is stable, the focus should shift to ensuring users are satisfied with the user experience by addressing any small tweaks that could make using the solution easier. Follow this up by addressing any defective tickets in the backlog.

Once the solution is in a good place, you can move on to the next phase of longer-term ongoing support, maintenance, and enhancement requests.

CONTINUOUSLY OFFER SUPPORT TO YOUR USERS AS NEEDED

KEEP THE DOCUMENTATION UP TO DATE WITH ANY NEW DEVELOPMENTS OR IMPLEMENTED FEATURES

REGULARLY MAINTAIN THE SYSTEM BY APPLYING PATCHES AND PERFORMING UPGRADES

PROIRITIZE ADDRESSING THE BACKLOG OF DEFECTS AND ENHANCEMENT REQUESTS IN A TIMELY MANNER

Sustainment

A sustainment plan is a vital component of any organization's software implementation process. It involves establishing an ongoing administration, support, and maintenance plan for the software after the Go-Live phase.

The goal of a sustainment plan is to ensure that the software continues to operate effectively, efficiently, and securely over its lifetime. To ensure the plan is well-designed and effective, several key factors must be considered.

First, it should account for support and maintenance activities, such as software updates, security patches, bug fixes, system upgrades, hardware maintenance, and user support. Additionally, a change management process should be included to properly assess, test, and implement changes.

Second, the plan should outline the required resources, including personnel, equipment, tools, and funding. Furthermore, it should identify potential risks that may impact the system's ability to operate effectively and provide strategies to mitigate them.

A disaster recovery plan should also be included to ensure business continuity in the event of a disaster or unforeseen event.

Clear metrics for measuring the success of the sustainment effort and service level agreements (SLAs) for support response times and issue resolution should also be established.

Finally, the roles and responsibilities of the stakeholders, users, vendors, and support personnel involved in the sustainment effort should be defined.

KEY FACTORS:

1. **ACCOUNT FOR SUPPORT AND MAINTENANCE ACTIVITIES.**
2. **OUTLINE THE REQUIRED RESOURCES AND IDENTIFY POTENTIAL RISKS THAT MAY IMPACT THE SYSTEMS ABILITY TO OPERATE EFFECTIVELY AND PROVIDE STRATEGIES TO MITIGATE THEM.**
3. **DEFINE THE ROLES AND RESPONSIBILITIES OF THE STAKEHOLDERS, USERS, VENDORS, AND SUPPORT PERSONNEL INVOLVED.**

There are several options for implementing a sustainment plan, including in-house, outsourced, or hybrid sustainment.

In-house sustainment involves building an internal team to provide support and maintenance for the system, which may be more cost-effective but requires significant investment in training, resources, and infrastructure.

Outsourced sustainment, where a third-party vendor is hired to provide support and maintenance, may be more expensive but offers greater expertise and access to specialized resources.

Hybrid sustainment, a combination of in-house and outsourced support and maintenance, may offer a balance of cost-effectiveness and expertise.

Organizations should evaluate the advantages and disadvantages of each option and choose the one that aligns with their needs and capabilities. Regardless of the option chosen, involving internal resources throughout the implementation process is essential.

This ensures they have a clear understanding of the software after implementation and facilitates a smoother transition to the sustainment phase after Go-Live.

RISK IDENTIFICATION

- **CYBERSECURITY RISKS**
- **NATURAL DISASTERS**
- **EQUIPMENT FAILURE**
- **HUMAN ERROR**
- **COMPLIANCE AND REGULATORY RISKS**

RISK MANAGEMENT

- **FIREWALLS AND ANTIVIRUS SOFTWARE**
- **DISASTER RECOVERY PLANS**
- **TRAINING EMPLOYEES ON BEST PRACTICES**

Closing

Embarking on the path towards an advanced Planning Solution marks a big step in the evolution of an organization. With the right resources, dedication, preparation and foresight, this major undertaking can be greatly simplified to minimize time, cost and disruption to business processes.

This guide was meant to shed light on this whole journey and provide tactical details of all steps involved. Share it with the core team members that will be involved and keep it as a reference.

Change is often difficult, and the road is long, but implementing a Planning Solution will greatly benefit the organization long-term by creating efficiencies, streamlining business processes, and providing greater insights into financial and operational performance so the company can continue to grow and thrive for many years to come.