TIP Publication Date
June 11, 2014

Packaged Application Solutions

Best Practices for Vendor Selection & Implementation

Enterprises and institutions across a variety of industries depend on external vendors' solutions to support core business processes, generate revenue, and accelerate growth. More times than not, the vendor selection and implementation process is difficult and disjointed, leading to a painful first experience with future strategic partners.

Selection and implementation of a packaged application solution is often undertaken either in a silo'd business unit with little enterprise insight and leveraging an undocumented and haphazard process, or through a painstakingly long process which then fails to place sufficient effort in understanding the business and technology implications of a packaged solution.

Trexin has defined a series of best practices that span vendor solution selection and implementation, which place the focus on business issues and drive cross-functional support. These best practices include:

1. Implement processes, not tools
2. Leverage industry best practices
3. Compete with the market, not the software vendor
4. Drive a sustainable vendor choice
5. Manage shifting project roles and skills
6. Take advantage of early system availability
7. Address functional gaps and overlaps
8. Map application and data architecture ASAP

As presented in the following table, each best practice is associated with a set of time-tested considerations that have been derived directly from years of cross-functional implementation project experience at the user, manager, and executive levels:

Best Practice	Considerations
Implement processes, not tools	• Often major gaps and issues are blamed on legacy enterprise tools. However, it is most efficient to view problem areas from the process perspective first. • To realize tangible targeted benefits, cross-functional integration of policy, data, people, and tools are required.

Leverage industry best practices	• A critical examination of your Target Operating Model and enterprise alignment will foster more efficient uses of a packaged software solution. • Properly executed RFIs & RFPs, coupled with a focused definition of functional requirements, will uncover a number of great opportunities for the business.
Compete with the market, not the software vendor	• Packaged software is intended to increase efficiency and capability in how you do business, but too often this can turn into a long-term, costly customizable project where functionality and utility from the vendor and their corporate roadmap is sacrificed. Assess your ability to tweak existing process and not the tool. • A critical realization lays in understanding that a vendor's solution capabilities have matured through a large amount of investment. As you start to customize a package, you miss out on market insight and planned releases and enhancements.
Drive a sustainable vendor choice	• The vendor selection process sets the tone for a long-term partnership and often becomes critical to the interworking and culture of your business. From the start, you must establish a "fair process" for selection and generate up-front "all in" commitment from the actual users, their managers, and the executives. • You should continually fact check the vendor and tool during selection and work to ignore shiny functionality. This can be accomplished through meeting the vendor's technical and implementation talent face-to-face, reviewing project methodologies and functionality roadmaps, researching vendor corporate viability and technology stacks, and most importantly reference checking vendor's existing clients and having the actual users attend live demonstrations.
Manage shifting project roles and skills	• In a packaged solution implementation an application's functionality and technology should be treated as "one thing". After purchasing the package it is best to leverage out-of-the-box functionality to every extent possible. • Packaged software affords for very user-centric applications and therefore data and integration work are critical to get correct.
Take advantage of early system availability	• During a packaged solution implementation, depending on data and integration work, the tool is able to be turned on quickly. Use this as an opportunity to learn the system and to drive final design as the remaining tool is configured. • Since packaged tools are intended to be deployed quickly, you should have the mind-set to begin prototyping immediately.
Address functional gaps and overlaps	• Functionality is mostly pre-canned and is not easily turned on or off, it is in effect a piece of the entire package. Address these areas through prioritization of existing functionality vs. any overlaps and create process where needed. • Similarly, you may find yourself in a situation where you have turned to packaged software to replace an existing area. However, when unplugging the legacy tool and process, gaps will be discovered. In this scenario, a cost benefit analysis is critical to assess creating new process, bringing in new people, or lastly customizing the tool.

Map application and data architecture ASAP	• A packaged solution is meant to be pre-defined and therefore will almost never be a perfect fit with existing interfaces and data standards in your environment. • During the selection's RFI and RFP process, work diligently to understand how and where the new package will fit into your existing architecture from an interface and data model perspective.

Once you have internalized packaged tool best practices and their associated considerations, these best practices will allow you to dig deeper into actual tactical impacts to vendor solution selection and implementation workstreams, as described in the following table:

Impact to Selection	Best Practice	Impact to Implementation
• Establish clear executive ownership, and brand the tool as a business solution. • Develop a future-state roadmap for how you envision the tool will enable your desired process.	**Implement processes, not tools**	• Do not name the project after the vendor, make it your own. • Keep the entire team focused on business goals and benefits, and organize implementation work and design by business process.
• Spend the time upfront to learn vendor and client PM approaches (if the people don't fit, neither will the tool). • Leverage demos and reference checks with actual users wherever possible, and treat it as a learning opportunity.	**Leverage industry best practices**	• Acknowledge and embrace the vendor's industry and process expertise. • Adopt the vendor's published and documented best practices to leverage their capabilities.
• Identify key design issues for early discussion and alignment. • Request the vendor's product roadmap and industry research up-front to validate their thought leadership.	**Compete with the market, not the software vendor**	• Avoid customizing tools! • If the solution will not fit, seriously consider moving on to a new process model or a custom build for your business. More times than not, customization of packaged software is extremely costly.
• Develop a detailed and cross-functional list of selection requirements, and force the vendor to address each requirement – ensure that you have a logical and "fair process" for selection, one that all stakeholders can buy into. • Identify your long-term experts and power users, and involve them right away.	**Drive a sustainable vendor choice**	• Avoid finalizing your design until you have learned the new system through and through. • Be intentional with knowledge transfer plans.
• Ensure that functional and technical teams feel joint ownership of the selection.	**Manage shifting project roles and skills**	• Focus on early communication and change management with project and user teams.

• Keep the selection process a very user-driven effort.		• Plan for data and integration to be the hot topics, and focus work areas for your technical team.
• Move quickly to a short list of vendors during the RFI & RFP process. Discredit show-stoppers immediately. • Take advantage of in-depth demonstrations with actual users and their managers present to build ownership and comfort levels.	**Take advantage of early system availability**	• As needed, conduct a proof of concept to build confidence in the solution. • Practice go-lives and demonstrate / share new capabilities with stakeholders as you implement.
• Ensure that the vendor offers rich integration experience via case studies and references, actual project plans, and the necessary tools.	**Address functional gaps and overlaps**	• Develop an identification and prioritization plan to address gaps and overlaps as part of the detailed implementation plan.
• Develop a detailed view of current data and integration issues and share those issues with the vendor immediately during the selection process as part of your assessment criteria.	**Map application and data architecture ASAP**	• Deliver on a quick and efficient timeline. Entering the implementation with control of the application portfolio and an understanding of the quality and location of your data will save vast amounts of time in multiple areas of the project.

These best practices around vendor solution selection and implementation are designed to make the evaluation process learning intensive and the implementation as streamlined as possible. Software projects are without a doubt a difficult task and selection and subsequent implementation of a vendor's packaged application solution can easily be derailed, especially on a global, enterprise-wide project. Focus on process, business issues, and efficiency is guaranteed to create cross-functional consensus and management buy-in, leaving you confident throughout the project.

This TIP was written by Dale Anderson, who specializes in Customer Centricity, Aligned Enterprise, and Delivery Leadership. Dale welcomes comments and discussion on this topic and can be reached at dale.anderson@trexin.com.