TOP 10 Mistakes Made Implementing IT Systems and How to Avoid Them

By Ted Farrington
With over 30 years of experience working in research and development (R&D) for a variety of companies and industries, I’ve seen my fair share of information technology (IT) systems implemented to support various global processes such as product lifecycle management (PLM), phase-gate, portfolio management, legal and finance.

I’ve observed these projects from all angles; watching from a distance, participating in and leading numerous efforts throughout my career. And the success rates varied – from failures that were abandoned within a year, to systems that never realized their potential, to sustained successes that added value for many years.

Over time, I began to notice a pattern behind some of the most common mistakes being made during the implementation of large global IT systems in support of R&D or other business processes.

Read on to discover leading practices for avoiding these ten common mistakes.

Being mindful of the challenges and solutions discussed here will greatly increase the chances of your next project becoming a sustainable success. Enjoy!
MISTAKE 1

The “Global” Roll Out ................................................................. 5
Invest the time to reach agreement on a global process from the start

MISTAKE 2

Playing the Shell Game .............................................................. 8
Make sure your project actually improves global productivity

MISTAKE 3

The “Fix All” Solution .................................................................. 11
View the IT system as a tool to lock-in a process; it cannot, on its own, fix process problems

MISTAKE 4

Ignoring Inconsistencies ............................................................ 14
Start off on the right foot by aligning on terminology and processes

MISTAKE 5

Missing the Point ......................................................................... 17
Clearly define the burning platform driving a global implementation

MISTAKE 6

Jumping the Gun ......................................................................... 20
Perform readiness assessments to improve ROI throughout the implementation
Skipping Stakeholders
Ensure that—at worst—users see no increase in workload, while most see a net decrease

Skipping the Dress Rehearsal
Perform real-world testing to ensure smooth sailing

Self-Gathering Data
Don’t ask the user community to migrate data to a new system

DIY Projects
Find a trusted advisor to manage expectations and deal with vendors
MISTAKE 1

The “Global” Roll Out
Invest the time to reach agreement on a global process from the start

I was once at a kick-off meeting for a global IT system project aiming for implementation across seven campuses around the world. Only one campus—headquarters—was represented at the meeting. When I asked who was representing the other sites, I was told,

“We’ll design and launch the system here first, then roll it out across the other campuses.”

You should never start a global IT project in the R&D process space with that mindset. It may take extra effort to include all campuses in the early phases of a project, but the dividends are paid back in the end because the project has buy-in from everyone when it goes live. You can either invest this time up front and have ownership from everyone at the start, or you can wait and pay the price on the back end when other locations resist implementation.

Here are some leading practices for avoiding the global roll out:

Keep processes simple

We all know that in large, multi-campus organizations, headquarters are often the most bureaucratic locations. Processes are overly complex with everyone needing to approve everything for many different reasons, including politics. The owners of all approving functions such as regulatory, safety, legal, etc. are all on site; probably with their own individual complex processes. Why model a system after the most complex location and then use an IT system to spread this complexity? Simplicity increases with distance from HQ for a variety of reasons. Isn’t simplicity what we want?
Tap your talent

There are many smart people spread across any large, multi-campus organization, so best practices can be found anywhere and everywhere. Why limit the project to the best practices of a single location? Unique perspectives and insights should be sought out at the start of the project to design the best IT tool possible.

Check for alignment

The culture and process richness of different campuses in a global organization are driven more by the behavior of their leaders than their countries of residence. So the strengths, weaknesses and challenges faced on these campuses may be very different regardless of location. You would be surprised at the number of times I’ve seen a single campus develop a new process with IT support to address a specific problem, then try to roll it out globally, only to learn no one else had the same problem!

Bottom line: if you’re leading a global IT system implementation for PLM, phase-gate, project portfolio management (PPM), or any other process, bring all locations and stakeholders to the table right from the start! You’ll have a much more successful and sustainable project in the end with a much higher return on your investment.
Playing the Shell Game
Make sure your project actually improves global productivity

Financial benefits from global IT system implementations, especially those seen as productivity improvements, often take the form of headcount reductions. But, these reductions cannot only be accounted for by the team sponsoring the initiative. They need to be calculated globally across the entire enterprise.

I once saw a global system that was justified by the elimination of 20 positions in one location who performed the job manually. However, the system was so time consuming and hard to use that other parts of the enterprise needed to add at least 20 temporary full time equivalents (FTEs) to keep the system running and not consume all their time. When asked for help in covering these costs, the project management office (PMO) replied,

“We didn’t budget for that; you’re on your own.” So even though the sponsoring team was able to claim a local cost savings, the system actually hurt productivity.

Similarly, at one time, all employees in my company received the following email from one functional team that had just installed a new global system:

“To allow our team to focus on higher order activities, we’re launching a new software tool for your use. Instead of having us enter and own your data, this tool empowers you to enter and take responsibility for your own information. Mandatory online training starts next week.”
In other words, it’s not a good use of our time to do data entry, but it is a good use of yours… These “offloading of work” projects are just another shell game, with the offloading organization potentially saving the time of a junior clerk at the expense of more senior personnel around the organization.

One of the most telling statements I’ve ever heard after a new system launch is, “Now that we’ve gone live, we need to collect end-user requirements.” Unfortunately, this is the norm more often than folks want to admit.

Obviously, none of these projects engaged all their stakeholders at the beginning of the process. To avoid making the same mistakes for your next project, follow this simple but effective principle:

Anyone who touches the new system will see—at worst—no net increase in workload, and usually a net decrease in their workload.
MISTAKE 3

The “Fix All” Solution
View the IT system as a tool to lock-in a process; it cannot, on its own, fix process problems.

It’s amazing how many projects make the mistake of jumping to system implementation without taking a hard look at what they’re automating first. Projects need to put the process first. In other words, take the time up front to ensure any processes involved—new product development (NPD), new product approvals, financial processes, phase-gate, etc.—are as efficient and effective as possible before implementing an IT system to support them.

View the IT system as a tool to lock-in a process; it cannot, on its own, fix process problems.

To avoid this “fix all” approach, there are a few things you can do to ensure your IT system serves its purpose.

First – Never automate a bad process

I once inherited a PLM system project where the company assumed that having all the information in a central location would help them automate the NPD approval processes. After following a lead product developer around for a few days, I ran to my leader and said, “We would be crazy to automate the current process these folks are using!”

It was a great case of “the tail wagging the dog.” Product developers spent more time navigating all the approvals required—sometimes more than once depending on development stage—than they did developing products. The approval requirements overlapped and were unclear. Clarifying and streamlining the development process took months, but was well worth the time. It ensured that the PLM tool locked in a more robust process. I recommend that anyone leading a project like this take some time to “walk in the shoes” of the people doing the work to truly understand the process before attempting to design or implement a new IT system.
Second – Ensure stakeholder understanding

To make sure that everyone understands the process, take the leaders of all approving functions offsite for a day and ask each to present the following:

- Their current review process and requirements for new product approval
- The reason why: trace every requirement to its source—either an external agency (FDA, etc.) or policy from headquarters

Two things will happen:

1. 50% of the current requirements disappear as they are the result of adding patches to cover one-time events that happened over the past twenty years. Very few are really required by an external agency or corporate policy.

2. Everyone sees the monster they’ve created for product developers. Six or seven approving functions asking for information that is 90% the same, but asking it to be formatted slightly different for their use!

Third – Empower the people close to the work

Finally, let the people near the work design the new processes and workflows. There clearly is a role for management to play in projects like this, but it is not to get down in the weeds and design workflows. Management must set clear business requirements but empower the people near the work to design the process and workflows. They know better than anyone how to build an efficient process and are extremely motivated to do so.

Taking these steps to ensure that your processes are effective, efficient, and ready for the support of a new IT system will exponentially increase the return on your investment and longevity of the solution.
MISTAKE 4

Ignoring Inconsistencies
Start off on the right foot by aligning on terminology and processes

Implementing an IT system is challenging enough in a single location, but add multiple campuses to the mix and new and unique challenges begin to arise. The biggest of which is simply starting off on the right foot with all campuses aligned on processes and terminology. To avoid the “Who’s on first?” routine, take the following steps to ensure global organizational alignment.

First – Get everyone on the same page of the dictionary

This can be a huge undertaking if you have multiple campuses using different processes and sometimes even different languages. On one project I was a part of, we had several campuses around the world that had been acquired over the years and each had different development processes and terminology. For example, the meaning of the word “prototype” was different to people in different locations. Set the standard definition for common words used throughout the process to avoid confusion and a lot of headaches while designing and implementing an IT system.

Second – Harmonize to leading practices

At the first face-to-face meeting of the previously mentioned project, representatives (product developers, not managers) from each campus presented the development processes and approvals they each used. Requirements, work flows and approvals to move forward varied greatly from campus to campus, as did leading practices. Someone asked, “Why do I need this approval when you don’t at your location?” This process forces approving functions to reevaluate and make their requirements globally consistent. It took about a year to agree on a global development process with common steps and approvals that met the business requirements for that project.
While not jumping right into system design drove some folks nuts, it was a year well spent and paid great dividends in the end. Remember, a leading practice can be found anywhere!

Design globally and only modify locally if needed. Limit local modifications to the global process developed by the team to those demanded by local business requirements; usually by a country’s regulatory agencies. This maintains consistency in process and terminology so that all that work put into achieving global alignment doesn’t go to waste.

Finally – Ask yourself, is the organization really ready for this?

Large multi-campus organizations, like R&D, go through stages of development. They often start out as operational silos, supporting their local business units. After some time, they may share information for the sake of efficiency. To eliminate redundant work, they may develop into centers of excellence. Ultimately, they can become interdependent, where the objectives of each site depend on work done at other locations.

At this point, project and product information must pass seamlessly from campus to campus in near real-time and an IT system becomes a necessary resource to support that effort. While having a global PLM system may sound exciting to an organization running as a bunch of silos, often, there isn’t a burning platform that will justify the effort required until the organization is closer to the interdependent state.

Ignoring inconsistencies is a sure-fire way to set yourself up for failure and unnecessary headaches.
By establishing global alignment from the beginning of the project, you ensure maximum efficiency and eliminate redundancies throughout the process.
MISTAKE 5
Missing the Point
Clearly define the burning platform driving a global implementation

I once witnessed a global implementation project go on for a year before the team had a serious discussion of why the project existed. They couldn’t figure out why folks weren’t more excited about the new system they were implementing. After hours of discussing the point of the project, the best reason they came up with was, “Because the boss wants it.” That’s not a great motivator and progress was slow at best. In order to create buy in and successfully implement a new system or process, you must clearly identify the burning platform, or *raison d’etre* for the project up front.

I’ve made this same mistake and assumed that because a senior executive requested a PPM implementation project across several campuses, all the campuses would be onboard and cooperative.

I’ll never forget the day a highly-respected campus VP called me and said,

“My team doesn’t see a lot of value in doing this, so we’ve decided not to participate.”

Years later a global PLM project emerged. This time around I was wiser and insisted that we were not going to take a single step without all campus leaders agreeing on the necessity of the project. Fortunately, they all agreed with the following statement—“50% of my product developers’ time is wasted redoing products transferred from other sites and entering data in multiple approval processes.” Pain is a powerful burning platform!
Phase-gate process and system implementation projects often suffer from the same burning platform problem. Ask yourself,

“Why are we implementing the phase-gate process and system?”

On a large product development project, with tens of millions of dollars at risk at different stages of development, it’s obvious that you need some gates where the team can pause, review the project, and decide whether or not to make the next big phase investment. But, what’s the real reason for a gate if the next stage only puts a few days at risk?

I’ve seen a lot of phase-gate process and system projects stumble because no one ever took the time to clearly articulate the global need and understand the needs of all the stakeholders across the organization. This is important because different needs lead to different process/system requirements.

For example, here are some common needs for phase-gate processes and systems:

- **Risk management.** When millions of dollars are on the line, the magnitude of resources consumed at the next development stage justifies a review gate.
- **Organizational diagnostics.** What’s the distribution of resources across phases? Are simple projects getting done at the expense of bigger, bolder initiatives?
- **Seeking a common language.** Senior management is tired of needing a secret decoder ring to understand how different campuses are reporting progress because their phase definitions and processes differ.
- **Trust.** Different phases are owned by different people, and those who own phase n+1 don’t trust those who own phase n to have done their job completely.
- **Everyone else has one.** So it must be a good idea, right?

Some of these reasons are better than others. The important thing to note is that the data required for risk management is very different than the data required for seeking a common language, and a phase-gate process may not be necessary at all.

To increase acceptance and usefulness, identify the real point and purpose to the project at the start and design the processes and systems with that in mind.
MISTAKE 6

Jumping the Gun
Perform readiness assessments to improve ROI throughout the implementation

One of the most valuable, but underutilized, tools for managing change while implementing large process systems is the **readiness assessment**. It’s important to gauge whether the process system is necessary and if the organization is ready for its implementation to avoid wasted time and energy. Although significant formal efforts are required, there are two types of assessments that will improve ROI throughout the project implementation.

Organizational readiness assessment

As discussed previously in *Mistake #4—Ignoring Inconsistencies*, large multi-campus organizations, like R&D, go through stages of development. Until an organization has progressed from silos to interdependency, there often isn’t a need to justify the effort required to implement a global IT system. The organization really isn’t ready for such a tool.

While determining your readiness, make sure to avoid organizational overload. This is a question of coordination and prioritization across process system projects. At one of my past organizations, every function in the company seemed to be launching some new system, completely uncoordinated. It felt like we were in training for a new system every month and we had no idea most were coming at us. I can’t claim to know the magic number, but most organizations can only handle one or two major process system implementations each year. Past this point, folks get overloaded and overwhelmed with change and nothing goes smoothly.
Technology readiness assessment

It’s incredible the range of computer savviness and comfort levels that can exist across campuses that have been isolated for years. One campus may still enter everything in paper notebooks and share data by emailing spreadsheets, while another, more tech savvy location may have developed home-grown systems to aid the development process. These differences are often ignored until it’s too late. Formal readiness assessments can be done in parallel with the process harmonization work already described.

On one project, I saw the PMO team spend one week every month working from the campus with the least experience in global systems and technology, just to make them more comfortable. A good readiness assessment should not become a “book on a shelf.” It must lead to concrete actions that will get the campuses as close to a common level of readiness as possible.

Organizations that jump the gun and attempt to implement new systems before honestly assessing their organizational and technological readiness risk project failure and will struggle to gain buy in from end users. By investing the effort to check all systems before launch, you increase buy in and functionality across the organization.
MISTAKE 7

Skipping Stakeholders
Ensure that—at worst—users see no increase in workload, while most see a net decrease

Earlier in this series in *Mistake #2—Playing the Shell Game*, I mentioned the principal statement for a truly successful IT tool implementation:

Anyone who touches the new system will see at worst, no net increase in workload, and usually a net decrease in their workload.

While it’s easy to say that a system implementation will increase efficiency in the organization, it’s critical to ensure that all stakeholders will see the benefits of the system first hand. An increase in efficiency for the organization becomes a moot point if daily users see an increase in workload. Good luck getting buy in from anyone who’s workload increases, and without buy in, you risk the project ultimately failing.

One of the key change management tools used to hold yourself to the above statement is the *Stakeholder Analysis*. Many functions touch a global PLM, phase-gate, PPM or finance IT system, including:

- Marketing
- Operations
- R&D – Management
- R&D – Product Development
- Packaging
- Legal
- Regulatory Affairs
- Product Safety
- Finance
- Tech Transfer
- Business Unit
- Purchasing
While the exact functions and roles that touch a new system depend on the type of system, they always fall into three categories of users:

### Approving functions

These folks approve the product or project’s advancement and ultimately the new product’s launch. Examples include finance, product safety, regulatory affairs, and legal. They may not use the system every day, but their role is critical. Stakeholder analysis of approving functions typically uncovers several complaints that a well-designed process and system can resolve. Here are some examples:

- Excessive time is required to check detailed new product designs against the hundreds of relevant rules, coupled with the product developers’ lack of detailed understanding of those rules. A great example is regulatory affairs. A new product may have numerous ingredients or components, each needing to be checked against regulations in all countries targeted in the product launch. A state of the art PLM system has a robust “guidelines and restrictions” module that tells the product developer their formula violates a regulatory rule. The benefit is two-fold as the violating formula never makes it to regulatory affairs and the regulatory affairs approver knows all the common rules have been pre-checked. This was so important on one project I saw that the entire set of regulatory and safety rules for Europe (European Cosmetic Directive), US (OTC’s) and South America (both North and South cones) were preloaded in the system.

- Approvers receive information to review and approve late in the process, in their view. And it’s hard to identify the final product for review as it continues to evolve until late in the process. This is particularly painful if the current process is done by e-mail.

- Finally, these folks often feel the information they receive is incomplete, so they waste time contacting the responsible party for more information.

All of these issues can be addressed through process design and system configuration, simplifying both the approvers’ and product developers’ lives.

### Observers

Many people, including R&D management, the PMO, business unit, etc., simply want more visibility to projects currently underway. Their needs are easily met, but a stakeholder analysis is still valuable to understand why for two reasons.
First, many of these systems are sold “by the seat” and are not cheap. Just because 300 people say they would like more visibility does not mean you should buy 300 more seats. And secondly, who wants 300 more people watching their work?

**Hard core users**

These are the associates who use the system every day, entering information and seeking approvals to keep their project moving. For a global PLM system, these are often the product developers working in the lab. They are the most important people needing a stakeholder analysis. By definition, they will be required to do more work for the good of everyone else as they typically are asked to enter all the data and product design information. It is crucial that their net workload decrease and that they see clear benefits from the system.

Stakeholder analysis often uncovers one or more of these challenges that a well-designed process and system can address:

- **Entry of information in multiple systems.** I’ve seen organizations where the product developers were expected to put information in up to six different systems over the course of a project. One time, I pointed this out to an IT implementation team, about their project already underway. Their response was “We didn’t budget for any interfaces.” So they became the seventh system for data entry! Building interfaces to eliminate redundant work is a huge consideration that needs to be scoped out during the budgeting process.

- **Repackaging the same information ten different ways.** Approving functions often want to see the product information in different formats and with slightly different information. This is a lot of non-value-added work for the product developer. Again, a well configured system can present these different views easily.

- **Waiting for approvals.** Checking a new product design against global regulations can be very time consuming. This, coupled with a backlog of products needing “immediate approval,” causes an excessive turnaround time for approvals (at least in the eyes of the product developer). A well-designed process and system can cut turnaround times from weeks to days or even hours.

A thorough, formal stakeholder analysis is the key to delivering on the promise to make everyone’s life simpler through a well-designed process and supporting IT system!
MISTAKE 8

Skipping the Dress Rehearsal
Perform real-world testing to ensure smooth sailing

Throughout my experiences implementing global IT systems, I was not the IT expert. I usually represented the customer—typically a global R&D organization. Despite that, I learned very quickly just how important system testing is to the success of the project. It’s always easier to fix a mistake caught in testing than one caught after go live. Believe me— I know firsthand. While any testing is better than no testing, I recommend gauging where your organization is to understand what needs to be done to work towards the highest level of testing, and ultimately, success.

Here are the four levels of testing I’ve seen over my career:

Level Zero: “Turn it on, it will work”

I actually saw one project go live with essentially zero real-world testing!

The system was designed to automate the global specification approval process, which had always been done manually. When the system went live, a basic raw material substitution that would require two or three manual approvals, triggered over a thousand required approvals in the automated system! Oops! User feedback was fast and direct, to say the least. No project should go live this way. Every project needs to provide a means for post-launch feedback, as even the best testing protocols can miss something. In the true tradition of Dilbert, the team that launched this mistake was recognized for the great work they had done (obviously not determined by popular vote).

Level One: Positive testing

This is a small scale, but thorough, testing of the system’s functionality performed by trained lead users. A good test requires a lot of planning, especially for a global system. Every campus should participate and every system functionality should be tested. The question being addressed here is, “If the users do what they’re supposed to do, does the system perform its functions properly?” I’ve seen positive testing done poorly, where folks who design and install the system just “kick the tires” of a few key functions, go live, and then resort to Level Zero to deal with complaints. I’ve also seen it done very well, with all functions and locations...
tested. Guess which system had a smoother implementation?

Level Two: Stress testing

As an R&D professional, I had never heard of this type of testing until I had the opportunity to watch a global PLM system as a member of the PMO. Since then, I’ve always pushed for it. My IT partner leveraged his relationship with a large software company to get this done. If we expected 100 users to be on the system simultaneously, a software package was used to simulate 1000 simultaneous users. After it was done, my IT partner came to me and said, “Ted, you have a memory leak!” To which I responded, “What do you mean? I’m only 50 years old, my memory is fine!” That’s when I learned that a “memory leak” is when a user (real or simulated) logs off the system, but the memory they’ve been using is not entirely freed up for the next user. The 10X stress test saw this accumulating memory loss. The system would have crashed due to insufficient memory a few days or weeks after going live. Without stress testing, we would not have known to fix the problem before going live.

Level Three: Negative testing

This is an important level of testing that rarely gets done at all, much less done well. What seems unfortunate to me is that with system training becoming more online and less thorough (have you ever raised your hand to ask the online training module a question?), this testing is even more important. While Level One asked how the system would perform given correct inputs, Level Three asks “How will the system respond if the user makes mistakes?” Examples include data entry errors, trying to advance without all required information and logic conflicts. Any necessary tweaks can be proactive rather than reactive.

Through experience, I learned how important testing is to the success of the project. As the testing process gets more stringent and more levels of testing are completed, the project implementation goes more smoothly. Skipping this step is like skipping the dress rehearsal before a big performance. A few test runs before launch can save both you and all end users a lot of pain and frustration in the long run and prevent an onslaught of tomatoes.
MISTAKE 9

Self-Gathering Data
Don’t ask the user community to migrate data to a new system

Data migration is a huge issue, especially if the new IT system is replacing different legacy systems and/or is the first automated system at some locations. One project I saw basically held users responsible for migrating old data to the system. While inexpensive in the short term, this approach is a big mistake for a number of reasons.

- **Consistency** – Users may refer to the same material by different names and enter them as such. Having the same materials entered under different names compromises the results of future searches and other system functions.

- **Accuracy** – Asking users to enter data will result in errors that compromise other functions of the system.

- **Completeness** – Suppose you are migrating product formula information to a new PLM system. The system will want the overall formula, as well as sub-formulas for any ingredients comprising several chemicals themselves, plus the manufacturers and costs for all ingredients. This can take hours and busy users won’t make the time or effort to do it well.

- **Compromised Functionality** - A very important functionality of any good PLM or formula management system is the “where used” function. This function lets you enter a specific ingredient and returns a list of all the products where that ingredient is used. When there is news that some chemical has been found to be contaminated at such-and-such location in such-and-such country, the CEO immediately calls someone and asks: “Are we using anything from that location in our products?” A complete, accurate and consistent set of data, along with a robust “where used” function, makes answering that call a lot more fun. Trust me, I know. I used to receive those calls! This type of complex, interconnected data should not be migrated by end users.

One challenge to any data migration effort is the different forms the data may take. One campus may keep data in the legacy online system (good news), another in spreadsheets (bad news) or even on paper (really bad news). Accurately budgeting the resources required to migrate data is
impossible until the various forms of data are understood. So it’s important to do some of this work before locking down the overall project budget.

Luck was on the side of one project I saw that needed to migrate a few thousand product designs, ingredients, and sub-formulas along with manufacturers’ information to a new PLM system across seven or eight campuses around the world. Data existed in all the forms shown above, often incomplete and sometimes with errors. The magnitude of the challenge wasn’t realized until the project was underway.

Fortunately, a scientist familiar with much of this information decided to leave the company, with no employment plans in the near future. At her going away party the project manager (now losing sleep over the data migration mess) asked her, “How would you like to consult for us? Doing nothing but data migration, from your home, with equipment we’ll provide?” She agreed and the PM slept well for the first time in days. Some formulas took days to track down and enter, but having a single knowledgeable person doing this was great. Consistency, accuracy and future functionality were all guaranteed!

While not all projects are as lucky as this one was, I have come to believe that data migration is best accomplished by a very small, focused team that is dedicated only to addressing the data gathering issues. Don’t forget to spend some time understanding the magnitude of the challenge before setting the budget!
MISTAKE 10

DIY Projects
Find a trusted advisor to manage expectations and deal with vendors

Above a certain project size, a third-party integrator becomes a necessity, especially when neither the software vendor nor the company have the resources required to do the job. But in many cases, even for smaller projects where the company thinks they have resources, teams should engage a third-party integrator. Perhaps the integrator just manages resources largely supplied by the client. But, there are still two very good reasons to engage a third party for your project:

Managing internal expectations and requests

*Scope creep is real.* Companies can be their own worst enemies and expectations can be hard to manage. Internal leaders are often too accommodating of requests from various parts of their organization, including:

“Can we include this in the project?”
“Can these people also have access?”
“Can we customize the system for our team?”

An internal project leader may feel the need to say “yes” to increase engagement and because he/she needs to work with these folks after the project is done. But accommodating these types of requests without thinking them through causes scope creep and therefore budget problems. System customization is a long-term issue because future system releases may not be compatible with customized modules.
Any good third-party integrator can say “no” as needed, and will take a tough “no customizations” position from the get-go.

Managing the vendor

Someone will need to play hardball with the software vendor whenever expectations are not being met. An internal project leader, who doesn’t do this regularly, won’t know when and how far he/she should push the vendor. An experienced third-party integrator has probably worked with the vendor before and has a much better understanding of their capabilities, limitations and inner workings. This comes in handy when the project faces challenges, as all projects do!

Don’t even think about writing your own code

The worst example of DIY projects is writing your own code when something meeting your critical needs is available off the shelf. I’ve seen companies try to maintain internally-developed code after the developer has left the company, and it’s not pretty. No one is dedicated to maintaining and upgrading the system and no commercial systems will want to interface with the home-grown code. The short-term cost saving is overwhelmed by the long-term risks involved.

A good third-party integrator will advise against such DIY efforts and play a critical role in keeping the project in scope, on track, and successful in the long term for both large and small projects. The budget required for a third-party integrator will easily pay for itself in time and effort ensuring smooth sailing through the entire project.
No project is perfect, and every implementation will be challenged by one or more of these issues at some point.

Over my 30+ years, I have made several of these mistakes first-hand and have seen varying levels of success - from failure, to unrealized potential, to a truly sustainable solution in place for over a decade.

While sometimes it is difficult to convince leaders to take the steps required to avoid these common mistakes, part of that challenge is recognizing the long-term payoff. I believe the level of success achieved directly relates to three areas:

- The level of effort put into Organizational Change Management before the system is even discussed
- The level of effort devoted to streamlining and harmonizing leading practices across the enterprise, before the system is even discussed
- The level of ownership given to the people near the work to design the work

My advice: if you’re considering a global system implementation to support business processes, invest the time and resources to avoid these mistakes. You’ll likely experience a more sustainable system implementation and you’ll significantly improve efficiency and effectiveness along the way. A little extra time up front sets the foundation for your project to provide value for many years to come. The system work is easy; the sustainable part is not!
Ted’s 35 years of research and development experience were built in the CPG industry, where he held leadership roles in advanced research and R&D. Early in his career, Ted served as a researcher at P&G and Kimberly-Clark and has 18 US patents. Later, he held R&D leadership positions at Kimberly-Clark, Georgia-Pacific, J&J and PepsiCo. Ted has also been active in the Industrial Research Institute for 20 years.

Ted holds a Ph.D. in Chemical Engineering from the University of Maine.

Phone: (267) 261-3567
Email: ted.farrington@kalypso.com
Kalypso is a global consulting firm, helping clients deliver better results from innovation in a digital world.

We deliver a comprehensive set of capabilities across strategy, operations and technology to improve innovation performance.

For more information, visit kalypso.com