waters
Financial Technology Intelligence

The Road to Success

Achieving success in large technology initiatives can be difficult and fraught with challenges, but with a smart, level-headed approach, firms can achieve astounding results. *By Craig N. Birkelund and Yogesh S. Kulkarni*

Craig N. Birkelund

Yogesh S. Kulkarni

The successful implementation of large strategic IT projects is a key component of meeting the organizational business goals in highly competitive industry environments. Technology is a tool that, when implemented in a disciplined and rapid fashion, can help drive company performance and shareholder value.

It is not uncommon, however, for organizations to experience major delays and significant cost over-runs in these initiatives. The result is hardly surprising: Initiatives fail to meet the desired objectives, and in turn put the realization of the organization's business goals at risk.

In this article we will explore why large projects fail, the root causes of these failures and how your organization can increase the probability of success. We will also compare and contrast two fundamental approaches—a "single release approach," which delivers all functionality into production at the end of the program, versus a "multiple releases" approach which is characterized by multiple deliveries to the business throughout the course of the program.

WHY PROJECTS FAIL

At Infosys, our experience shows that there are several common themes that characterize delayed and over-budget IT projects. These themes include:

- A lack of clarity and alignment concerning the business and IT objectives and drivers. This leads to gaps between the execution plan and the expected outcome.
- Insufficient preparation and groundwork done to determine how the project will be executed in terms of the number, sequence and scope of releases, and associated data migration.
- Infrequent or a complete lack of intermediate deliverables during the course of large implementation initiatives. This includes a lack of early validation through prototyping and/or proofs-of-concept.
- Inadequate analysis of the impact and synergies between the current project and other initiatives within the organization.
- Lax project management characterized by issue reporting rather than proactive risk mitigation, and regular metrics-based tracking and monitoring.

The common thread linking these factors—and this is not a comprehensive list—is that they all relate to the quality and quantity of pre-development system implementation planning. These factors can not only be avoided at the start, but can also be tracked and mitigated before they impact delivery.

INCREASE YOUR PROBABILITY OF SUCCESS

Significant groundwork is done assessing the functional and technical requirements associated with a new IT initiative, including the architecture and design associated with a new system. Less effort is typically spent on determining the optimum method to actually implement the system. While both analyses are important, increasing the focus on how the system is implemented will improve your chances of success. The key to determining the right implementation strategy is making the decision based on a robust methodology that can objectively evaluate various options and recommend the most suitable one for your organization.

The following methodology allows you to systematically evaluate implementation options in the context of the drivers you determine are central to your decision-making process. At the conclusion of the process, this methodology provides you with a quantitative basis to compare and evaluate various implementation options.

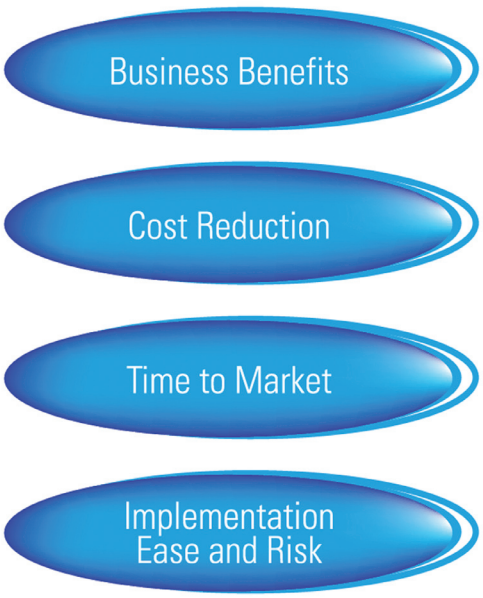


Figure 1: Implementation Strategy Drivers

1. **Be Clear on Objectives and Drivers—**A fundamental element to evaluating how you would go about implementing a technology platform is determining and prioritizing the drivers that are important to your company. In our experience, there are several key drivers that companies identify as important to them—these drivers should form the basis for an implementation strategy. *(See Figure 1.)*

   - **Business Benefits—**One of the key considerations when determining the choice of implementation strategy is the "value add" provided to the customer and to the company. A single release approach allows you to implement the system in any order you like, but no stakeholders will receive the enhanced functionality until the entire system is completed. With a multiple release approach, you may be somewhat restricted in the order in which you can implement the system, but as functionality goes live, it can be rolled out to your customers, thus providing business benefits to them after each release.

   - **Cost Reduction—**In the context of implementation options, this driver encompasses both the relative cost of building and deploying a new system, and the cost reductions achieved post-implementation. In a multiple release approach, these could include reductions resulting from the early de-commissioning of other applications that are no longer needed, and/or a reduction in support costs for manual processes that are being phased out. With a single release approach, development costs are reduced, as this is the least expensive way to actually do the implementation. The downside is that the post-deployment cost reductions are deferred until the entire system is deployed into production.

   - **Time to Market—**The ability to address critical pain points in the existing systems, while delivering business benefits fast enough to both your business and your customers, is a key consideration in determining the number, duration and the content of releases.

   - **Implementation Ease and Risk—**During any large development effort, legacy systems will most likely continue to run and be a crucial component of your day-to-day operations. This overlap stresses the IT organization in a number of ways, including the need to build bridges between the two systems and to minimize the number of releases scheduled during implementation. Building bridges can be expensive, and they are often "throwaway" solutions. On the flip side, any organization has a finite ability to manage releases. Determining the optimal balance is of paramount importance as it impacts both risk and cost.

2. **Drill Down and Prioritize Drivers—**These drivers, and others you may deem to be important, are often consistent across organizations. But once you have determined the final set of drivers that will guide your decision-making process, they need to be drilled down further into a set of business, technology and delivery parameters. These parameters then need to be prioritized based on a series of criteria and their respective importance. This process lays the foundation for a scorecard that, when finished, will facilitate a quantitative evaluation of your various implementation options. *(See Figure 2.)*

Driver	Scorecard Parameters
Business Benefits	• Service accuracy and enhancements • Flexibility, performance and scalability of the new platform • Operational efficiency • Ability to onboard clients faster • Ability to cross-sell • Ability to free up staff for strategic work
Cost Reduction	• Ability to decommission legacy applications • Respective software and hardware costs • Ability to reduce support and maintenance costs • Ability to reduce cost per service/transaction
Time to Market	• Ability to deliver business benefits early
Implementation Ease and Risk	• Impact and synergy with other projects • Flexibility in the build and release approach • Complexity of building bridges with the existing applications • Issues related to migration and replication

Figure 2: Scorecard Parameters

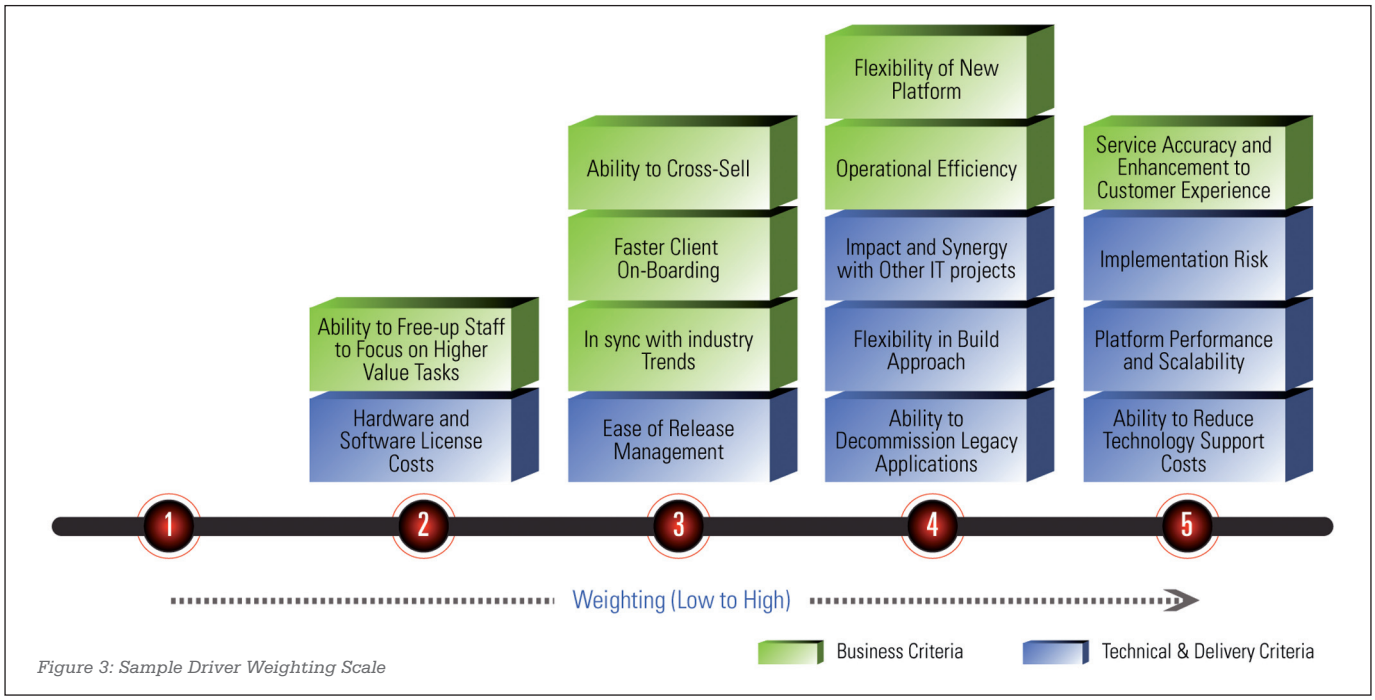


Figure 3: Sample Driver Weighting Scale

3. Define the Scorecard—Once you have defined and detailed the drivers you will use to help determine your implementation approach, they need to be applied to a scorecard that allows you to assign relative weightings to them. This is the first step in creating a quantitative comparison. In Figure 3 above, you can see how we have graded the various drivers and their parameters on a simple scale of 1 to 5, with 5 being the most important. Naturally, your drivers and the weightings you apply to them will be unique to your organization, but this diagram gives you a sense of the process. You will also note that the parameters are divided into technical, delivery and business issues—which we loosely consider to be internal and external drivers.

4. Identify Various Implementation Options—Drivers determine implementation strategy. In summary, there are two main options for delivering large IT projects. *(See Figure 4.)*

- **Single Release**—This involves warehousing the components until the entire program is completed. This requires all modules of the system to be completed before it is put into production
- **Multiple "Incremental" Releases**—This involves delivering components to users as they are developed. This allows each part of the new system to be used as they come online.

There are pros and cons to each of these approaches, but there are many potential permutations of the incremental approach, depending on your objectives and drivers.

The single release approach is the most risky from a delivery perspective as there is no early validation. However, it is a low-cost option in terms of implementation due to the fact that there is no overhead associated with managing multiple production releases. In addition, you can build it in any order you choose and it is not necessary to build bridges to the legacy systems. This approach can be attractive, especially if delivery risk can be mitigated and there is no urgent time-to-market pressure.

That, of course, is rarely the case. Time-to-market is typically of key concern, as are associated costs such as legacy decommissioning and reduction of manual processes. These early gains, coupled with low delivery risk, can often lead to the judgment that a multiple release approach is preferred. If that is the case, the prioritization process will assist in the determina-

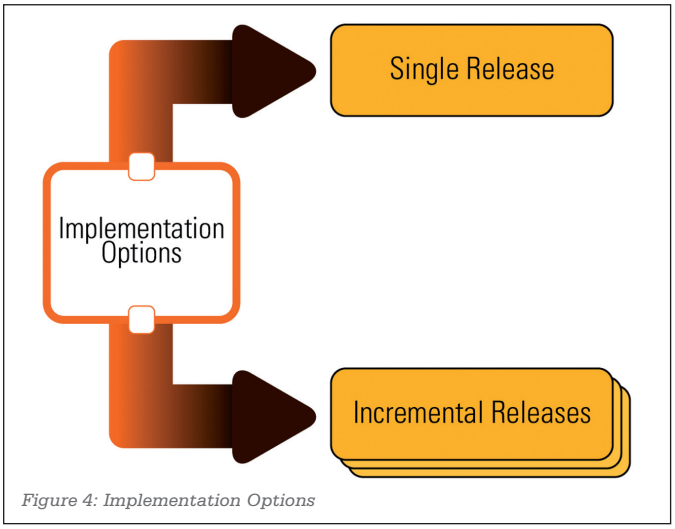


Figure 4: Implementation Options

tion of which incremental approach is appropriate in terms of number of releases, scope and sequence of releases.

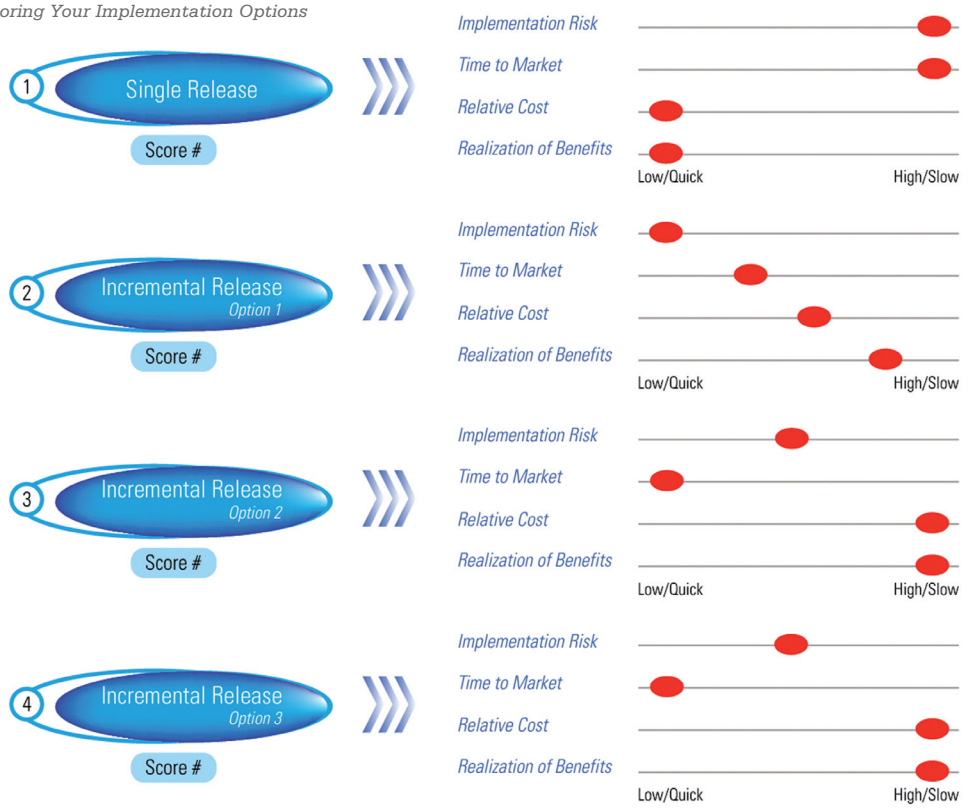
5. Score Various Options and Choose the Most Suitable One—By this point, you should be able to compare the scores associated with each driver. Through the scoring process, the various parameters you choose create an overall weighting for each driver. In Figure 5 below, you can visualize the comparative ranking of the drivers and their overall scores. It typically becomes clear which implementation option is best suited for your situation.

THE WAY FORWARD

Technology can and should enable the realization of business and financial goals. Too often, though, it becomes a source of time and budget over-runs leading to substandard results. However, these issues can be mitigated through the careful, upfront planning of an application's business use and implementation method.

Finally, determining the drivers of your program and designing your implementation approach based on the prioritized weighting of those drivers should significantly improve your probability of success. ■

Figure 5: Scoring Your Implementation Options



Craig N. Birkelund is a senior engagement manager with the banking and capital markets group at Infosys Technologies Ltd., where he focuses on issues of strategy and operations in the securities industry.

Yogesh S. Kulkarni is a principal architect with the banking and capital markets group at Infosys Technologies Ltd., where he focuses on strategy and delivery of large programs.

Infosys®
POWERED BY INTELLECT
DRIVEN BY VALUES

About Infosys Technologies Limited

Infosys Technologies Ltd defines, designs and delivers IT-enabled business solutions that help Global 2000 companies win in a flat world. These solutions focus on providing strategic differentiation and operational superiority to clients. Infosys creates these solutions for its clients by leveraging its domain and business expertise along with a complete range of services. With Infosys, clients are assured of a transparent business partner, world-class processes, speed of execution and the power to stretch their IT budget by leveraging the Global Delivery Model that Infosys pioneered. Infosys has a global footprint with over 40 offices and development centers in India, China, Australia, the Czech Republic, Poland, the UK, Canada, Mexico and Japan.

Infosys' service offerings span business and technology consulting, application services, systems integration, product engineering, custom software development, maintenance, re-engineering, independent testing and validation services, IT infrastructure services and business process outsourcing.