



# Investing in Agile: Aligning Agile Initiatives with Enterprise Goals

by Dan Murphy and Dave Rooney

## UNFREEZING REQUIREMENTS

IT project management has evolved considerably with the introduction, in recent years, of agile methodologies. However, the organization as a whole has not kept pace. Specifically, the strategy, planning, and budgeting processes need to be modified and interwoven in these agile methods. Our objective in this article is to clearly illustrate how a more iterative planning/budgeting process and a change in governance models within the organization will provide the necessary ingredients to create the agile enterprise.

Current planning and budgeting processes and organizational governance models are designed to support traditional waterfall processes. Waterfall processes assume that:

- Detailed requirements of a project must and can be documented up front
- Requirements will not change over the life of the project

In today's world, it is a fairy tale to assume that requirements will not change over the two- to four-year life of a traditional large IT project. The business environment that drives these IT requirements cannot be frozen for the life of the project. This inability to encompass change is the Achilles' heel of the traditional IT project and many CIOs' careers as well.

These same assumptions are true for traditional budgeting processes. They also are waterfall in nature. No project can move forward until a detailed business case is complete. The business case leads to the project budget estimate, and it is the underlying reason why requirements must be frozen and why projects fail.

We assert that organizations must structure themselves and their IT implementations as *operational processes*, not *capital projects*. Capital allocation is at the heart of any organization's decision-making processes, and it is at this inception point where agile methods can have the most significant effect on driving consistent business value.

## THE STATUS QUO: TRADITIONAL WATERFALL METHODS

Before we can argue for agile planning and budgeting methods, it is essential to understand the current paradigm under which most organizations are operating. Traditional waterfall thinking is pervasive in most organizations, and it has a profound impact on their strategy, planning, budgeting, structure, and governance processes.

It is important to understand that the waterfall defines what management does. Management's role is to allocate enterprise resources in the best manner possible in order to optimize return. Before senior managers can make a business decision, they generally like to have all the facts, or as many as possible, at the start. Hence, there is a great emphasis on up-front planning. The following list outlines the steps in the traditional waterfall approach:

1. Project idea or request
2. Feasibility study/business case
3. Detailed requirements and planning
4. Architecture design
5. Implementation
6. Testing
7. Evaluation

The project actually starts with the third step, detailed requirements and planning. At this point the enterprise is already committed to the traditional waterfall methodology. The requirements are gathered at the beginning and captured in a written document, which, at a given milestone, is frozen; afterwards changes are not permitted. Freezing requirements is thought to be a necessary step because the business case, capital allocation, and control are tied to this process.

In the waterfall approach, detailed requirements gathering is a necessary step for the implementation of any project regardless of size. It is not the requirements

themselves that present the problem for the traditional planning process, but rather the latency between the time they are captured and the time they are implemented (see Figure 1). Requirements have a shelf life: the longer the time between gathering and implementation, the greater the risk impact of change. Change makes requirements obsolete. Obsolete requirements make projects much more costly and can dramatically extend project timelines.

In addition, long latency periods between the gathering and implementation of requirements mean that project teams get no feedback. Feedback is the key requirement of any quality process. Without feedback, there is no learning. The waterfall process attempts to remedy this by investing more resources in the early stages on planning and risk management and on the back end of the process through change management. Unfortunately, these measures rarely solve the problem.

Finally, this need to gather and freeze the requirements up front is costly and provides no returns in the early part of the project. For smaller projects this is of less concern, but on large projects it creates a large deficit. It is not uncommon to see 40%-50% of the total project funding invested in the requirements gathering phase with little to no return. If the requirements are wrong, the investment is lost. Again, organizations often respond by devoting even more attention and resources to the requirements gathering and risk management components, which in turn results in a larger deficit before returns can be realized.

The waterfall method is still quite valid in environments with little to no change, in which the organization knows at the outset exactly what it is implementing because it has done so before. However, in today's competitive environment, this is seldom the reality. There are many factors that can dramatically change an organization's business requirements overnight. The events of 11 September 2001 prompted immediate changes to most organizations' security requirements. The US Sarbanes-Oxley Act produced instant requirements for much more rigorous accounting practices. Firms can be caught off guard by innovative moves from competitors, and it would be impossible for the traditional

waterfall planning process to predict or, more importantly, to accommodate these changes. In fact, introducing change to the waterfall process increases the project cost and extends the schedule, often exponentially.

## ORGANIZATIONAL STRUCTURE

The waterfall paradigm has also had an influence on many traditional organizational structures (see Figure 2). The client, or line of business (LoB), is responsible for the requirement, and the IT organization is responsible for fulfillment. The traditional organizational chart separates these functions. The requirements document is often a major milestone in a project. Once requirements are frozen, the client is "off the hook." Now IT, as a staff function, must deliver the solution. The client is removed from the IT implementation team, and as a result of the structure, the relationship between the two is often confrontational.

The IT department itself is also structured for waterfall processes. Classic examples are IT organizations that have separate database, test, and development departments. This structure is well suited for the separate architecture, development, and testing components of the waterfall approach. It does not encourage teamwork, however, and it is not well suited to accommodate change.

Finally, many organizations still see IT as a staff function separate from the main business functions of the organization. The CIO needs to be at the executive table, and organizationally, there needs to be a structure that encourages much more sharing and teamwork.

The shortcomings of the traditional waterfall methods are well documented in Craig Larman's book *Agile and Iterative Development, A Manager's Guide* [1]. Larman provides substantial evidence to illustrate IT projects' historical failure to deliver consistent and predictable business value to the enterprise. It is still common to see IT project failure rates in excess of 60%. Most of the indicators point to the requirements gathering phase of the waterfall process as the cause of failure. Given the fact that many of our business operations are dependent on the success of IT and IT projects, it is easy

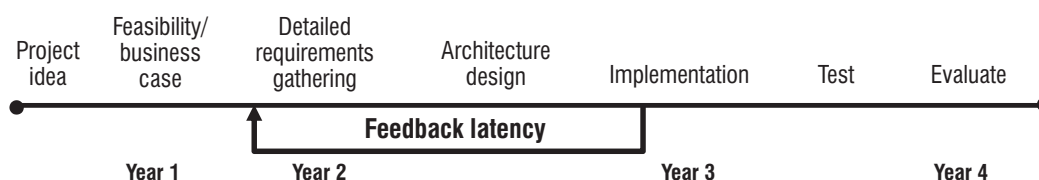


Figure 1 — Waterfall latency.

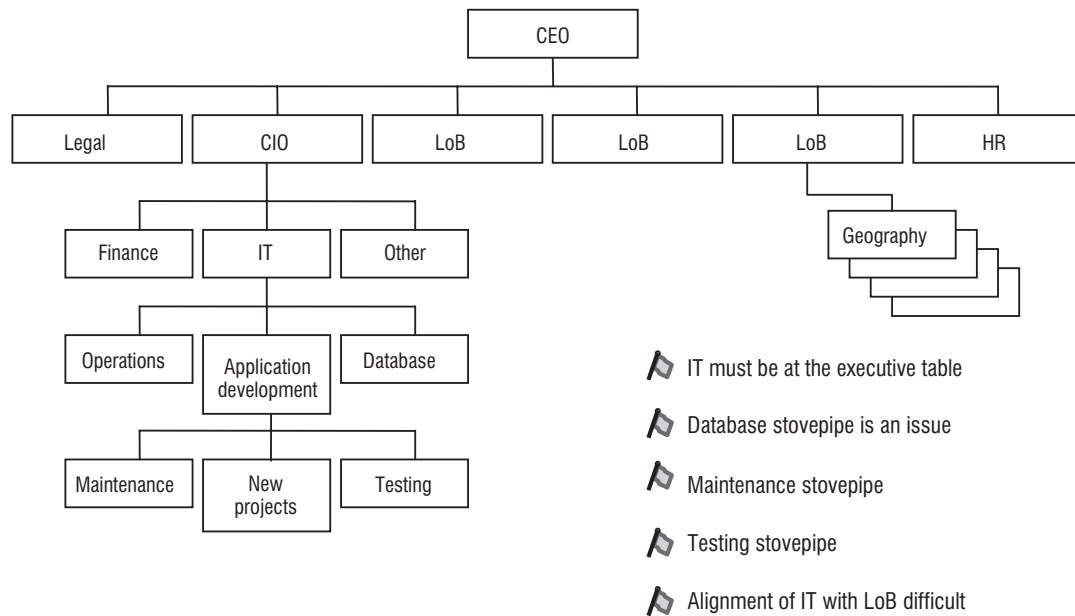


Figure 2 — Traditional organizational structure.

to understand the value to the business in finding a way to improve the odds for success in this area.

## MOVING TOWARD AN AGILE ENTERPRISE

Agile methods are a class of methodologies that try to accommodate change rather than control it. Agile methods advocate gathering detailed requirements in real time, directly from the user, as close to implementation as possible. That is, rather than trying to gather detailed requirements up front, these methods propose that high-level requirements be gathered and then the project or implementation be broken down into digestible components and implemented immediately. Large projects are broken down into subcomponents, which are further broken down into iterations. The detailed planning process happens at the iteration level.

Coding is done with integrated teams. The user organization is engaged throughout the project, continually defining and refining requirements. If change occurs in the business environment, it is incorporated into the project and implemented through the iterative process. If the implementation is not achieving the desired outcomes, it can be modified or cancelled in order to minimize loss.

Many firms understand this process at the technical level. However, they still have issues with agile methods in the following areas:

- Integration of agile into the overall enterprise planning process

- The organizational structure needed to optimize agile processes
- Financial/budgeting considerations

## Integration of Agile

Senior managers are often unaware that their organization is conducting projects using agile methods. The result is that the organization is still operating in waterfall mode at the senior management level, and agile methods receive little credit for successful project implementations. Consequently, these implementation methods are isolated and sporadic and often in conflict with the waterfall approach senior management is using.

To address this problem, we propose a top-down approach to create awareness and to provide an enterprise-wide framework for agile methods. IT portfolio management is the first step in developing the framework for the integration of agile implementations (see Figure 3). This is a critical factor for the successful implementation of IT in general, and it will create the necessary awareness of agile at a senior level. This awareness provides a more prolonged, enterprise approach and commitment to agile methods.

Business goals and objectives driven from the strategy need to be aligned with the IT portfolio of projects and prioritized at the corporate level. This happens with the CXO team as well as the LoB executives. Application portfolio management (APM) is an ongoing process of prioritizing project initiatives. It is critical with respect to buy-in and accountability at the CXO level, yet not many

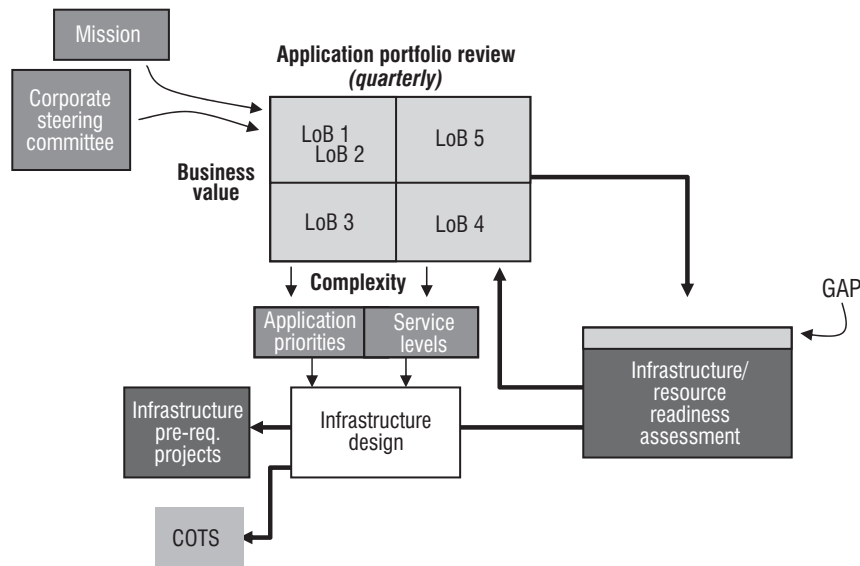


Figure 3 — APM process.

firms have implemented it as a framework for their agile initiatives. APM ensures the prerequisite infrastructure is in place for all projects in the enterprise.

The organization needs to have a list of IT implementations that are categorized according to their business value versus their complexity. The business value must be measurable, and the business executive who is accountable for the initiative must validate the measure(s) used. This provides the necessary ingredients for making clear and meaningful tradeoffs among project initiatives.

The APM process consists of two major components:

1. The application portfolio review (APR)
2. The infrastructure/resource readiness assessment (IRRA)

The APR is the actual planning session held by the executive team that produces the list of project/implementation priorities and associated service levels. This serves as input into the second component, the IRRA.

The IRRA answers the question, “Do we have the technical and people resources and facilities to implement the projects that have resulted from the APR?” The IRRA usually provides a list of prerequisite gaps, which should be factored back into the complexity score. A second APR ensures that the priority of the implementations is correct. This process is ongoing, and the senior management team needs to revisit it on a quarterly to semiannual basis depending on the nature of the business (i.e., the amount of change). The IRRA ensures that

the necessary components are in place to implement the portfolio of application projects.

APM ensures that:

- High-level goals and plans are aligned with IT initiatives
- High-level business value drives prioritization of initiatives
- There is accountability and responsibility at the executive level
- The infrastructure is kept up to date and is capable of supporting the applications the business needs

### Structuring the Agile Organizational Governance Models

Before we can begin to implement the portfolio, we need to establish clear lines of accountability and responsibility and connect them directly with the agile teams. Figure 4 outlines a potential governance structure for an agile organization.

Governance models ensure the vertical integration of agile into the enterprise. Those familiar with agile methods will recognize the lower part of this chart. It includes the user team, whose members must work together to provide the user stories (requirements to be implemented in small increments of a few weeks to a month, depending on the variation of agile), and the development team, whose members are responsible for delivering high-quality, production-ready code every iteration according to their estimates.

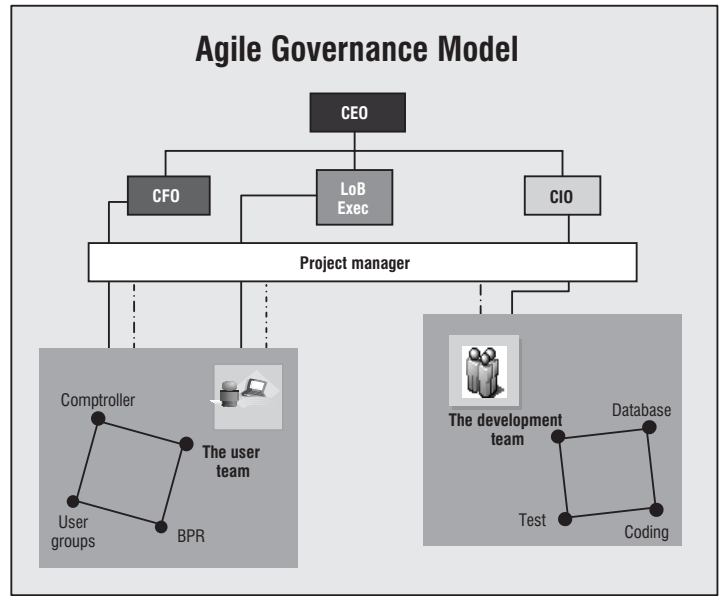
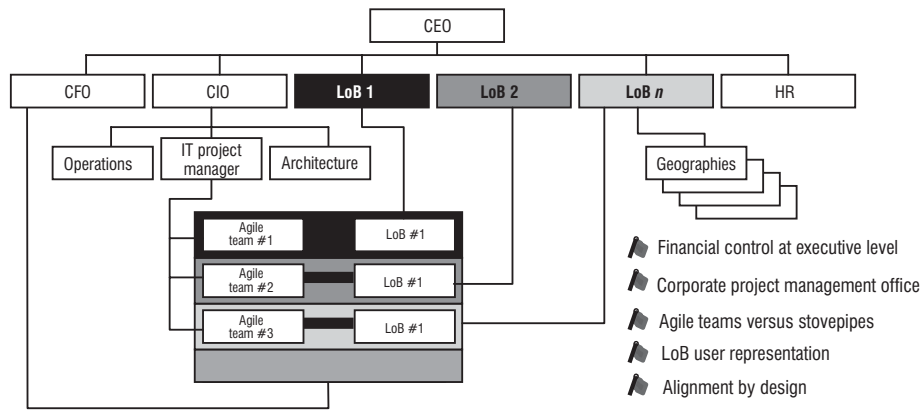


Figure 4 — Agile organizational structure.

The user team is responsible for the development and prioritization of the user stories. The team must have dedicated resources from the line of business that can articulate the requirements and the changes in the requirements as the organization sees fit.

The comptroller function is a key component of the user team. The comptroller must ensure that the iteration meets the fiscal considerations of the enterprise. Specifically, he must make certain that the cost/benefit metrics of the iteration meet the organization's business requirements and are connected with the business value articulated in the APR. The comptroller is also responsible for the documentation of those business value metrics on a continual basis. This results in an ongoing cost-benefit analysis of the project. The comptroller reports directly to the CFO and indirectly to the project manager.

“The user” is the other key member of the user team. For the purposes of this discussion, the user is the person responsible for the development and prioritization of the user requirements. She reports indirectly to the project manager and directly to the LoB executive, who benefits from the business value created from the project. If there is any confusion regarding the prioritization of user stories, the LoB executive is responsible for clarification. If the project is cross-functional, the CEO or another cross-functional executive must be assigned to be accountable for the outcome of the project from a business perspective.

The collocation of the user team with the development team provides a structure that supports and facilitates IT alignment. The development team lead reports directly to the CIO and indirectly to the project manager. This governance model ensures alignment, accountability, and responsibility. It also ensures quick escalation of issues, making the enterprise more agile.

Multiple agile projects can be running at once, but they require a significant investment by the enterprise architect in understanding the overall business architecture. It is important to follow the agile practice of starting small and growing versus taking the big-bang approach.

The other key component of the governance model is the CIO. It is not uncommon in organizations today for the CIO or IT director to report to “Corporate Services,” along with Human Resources and Finance. In these organizations, IT is seen as a staff function and therefore nonstrategic. The historic track record of IT projects and IT’s inability to connect itself to the strategic mission of the business have eroded the credibility of IT and the CIO in many organizations. For agile to work, the CIO must be at the executive table.

### Agile Budgeting: Operational versus Capital

Most organizations still look at IT as a group of projects that need to be implemented in order to drive business goals and objectives. Once the word “project” is introduced, we automatically assume a capital-based initiative (CBI). A CBI mandates that the requirements be known at the outset; clear understanding of the requirements leads to a detailed business case. This is the key component of the application process for capital funds in most organizations. On large projects and/or in organizations with a history of failed IT projects, the requirements gathering and business case processes become more rigorous and detailed. Once again we have the classic waterfall dilemma of more money being invested up front with no real tangible returns to the organization (see Figure 5).

This would be analogous to a four-year investment in which there would be a 70% chance of losing anywhere from 100% to 200% of the original investment. The investor would agree to spend 50%-60% of his investment on research, knowing that any possible returns would not be realized for 24 months. After 24 months have passed, the investor realizes that he is losing his money. Yet instead of aborting, he continues to invest two to three times his initial investment in order to try to recoup this embarrassing loss (see Figure 6).

The process of allocating capital funds is a key factor in the freezing of project requirements. Often the requirements are frozen for contractual reasons. Prices and costs are fixed to requirements in the tendering process. If the client changes the requirements, the vendor usually has the option of increasing the price. Changes introduced after contract award can also significantly lengthen schedules. Agile methodologies,

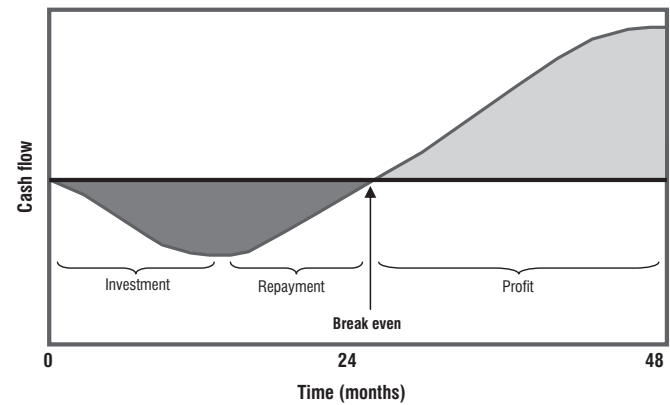


Figure 5 — Successful waterfall project.

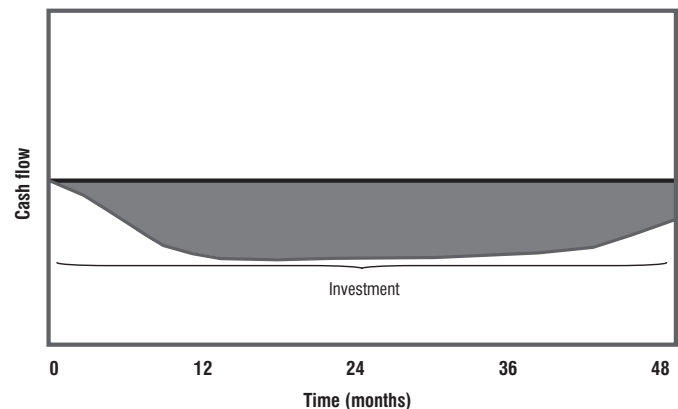


Figure 6 — Unsuccessful waterfall project.

however, are designed to accommodate change and thus can provide some benefits when applied to the budget planning process.

### AGILE BUDGETING CONTROL (ABC)

In our opinion, CBIs no longer fit within the context of software development initiatives in most organizations. IT initiatives should be largely operational in nature. The very nature of agile is operational in its approach. Using the word “initiative” to replace “project” would be a key starting point in shifting organizations from a capital perspective to an operational perspective. The current trend of breaking larger projects up into smaller components and holding funding until the previous milestone is accomplished is a also small step in this direction. Unfortunately, many capital-based waterfall projects are phased into components of the waterfall process itself. As a consequence, many organizations still spend 50%-60% of their capital project allocations developing requirements and doing business process design work before one drop of code is delivered.

By contrast, the agile budgeting control (ABC) approach ensures funding for business value. An initial three-month startup period is required to set up the teams and governance structure. There is also some time required to learn the new process, rules, and procedures. Subsequent to the startup period, the executive team should review funding decisions every quarter (see Figure 7). In this way, the corporation ensures business value and accountability on an ongoing basis. The business value measures must be defined and maintained by the business, not IT. Projects that are not producing are stopped (see Figure 8), while successful implementations are realized earlier, resulting in strong net present value (NPV) scores.

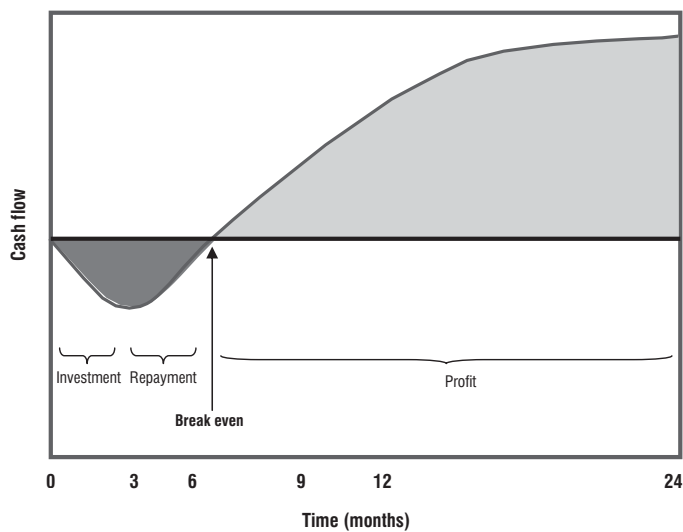


Figure 7 — Successful agile project.

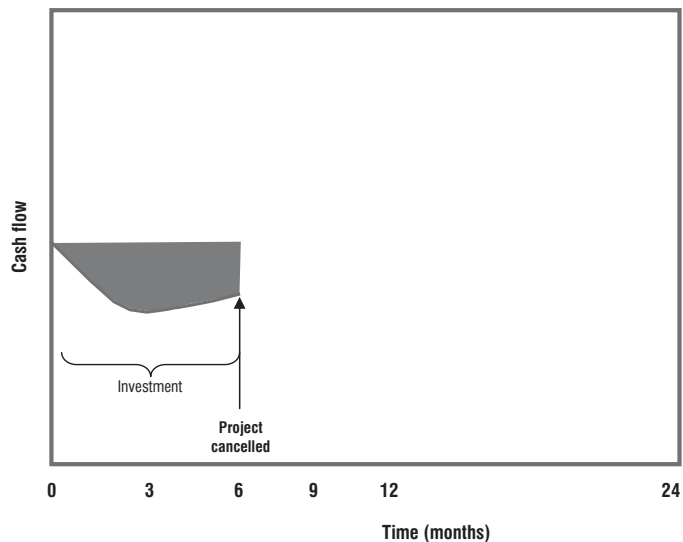


Figure 8 — Unsuccessful agile project.

The comptroller is responsible for setting up the financial measures within the user team and comparing the team's achievement against those measures on a quarterly basis. This accounting exercise is a key component in connecting agile to the business planning strategy side of the organization. The feedback provides essential input into the planning process.

## AGILE AND ORGANIZATIONAL CHANGE MANAGEMENT

In most organizations, waterfall methodologies create an environment in which change occurs in large batches two to five years apart. This is challenging for most organizations. Users become comfortable with a particular system or systems and have trouble handling a large change all at once. This phenomenon has created a change management industry among the big consulting and training companies.

By contrast, agile methods introduce little bits of change every month or two. This almost entirely eliminates the need for large-scale change management. Everyone gets used to a little change on a routine basis. Over the longer term, the organization develops the skill and capacity to adapt.

## AGILE AND RISK MANAGEMENT

Waterfall methods also require more stringent risk analysis up front on large projects because so much of the investment is made early in the project and the returns are not realized until late in the process. The organization has one chance to "get it right" or the investment is lost or compromised.

By contrast, agile methods do not require large amounts of capital and resource commitment early on. The investment is made in much smaller, controlled increments. If the implementation is having trouble meeting the expected metrics, agile teams can take action in short order. In addition, the much tighter feedback mechanism provides for learning and better estimates on iteration resource allotments and potential returns.

## INTEGRATING AGILE

Ensuring the success of agile methods across the organization requires the implementation of a process that is integrated with every aspect of the planning, budgeting, and governance structures within the enterprise. This starts with strong and ongoing engagement from the CEO.

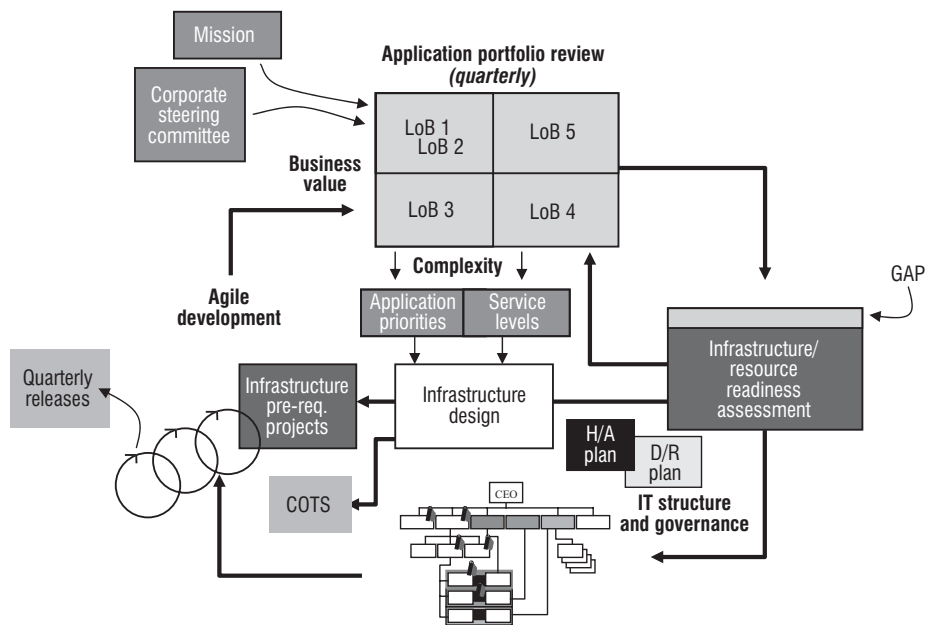


Figure 9 — Creating the agile enterprise.

To recap, applications initiatives need to be prioritized according to the value they offer (as defined by the business owners) versus the risk/complexity they present. IT then needs to understand the baseline of prerequisites that are required to implement the chosen portfolio of applications. The IRRA will reveal which gaps in the infrastructure need to be addressed by IT infrastructure projects, and this information will factor into the application complexity scores calculated in the first APR. The second APR determines the final priority scores for the applications in the portfolio. The APR should also determine associated service-level requirements for each application initiative. These feed the infrastructure design as well as the high-availability and disaster planning processes.

The next step is the IT governance planning process, which answers the question, “Are we structured to ensure teamwork, cooperation, and success?” This entire process is essential for the success of the agile enterprise. In our opinion, without a thorough approach to planning, budgeting, infrastructure, and governance, agile initiatives are nothing more than technical initiatives operating outside the enterprise planning process. Such initiatives will be ad hoc and unmeasured, and they will not enable the enterprise to achieve the full measure of success with agile methods.

## GROWING YOUR OWN AGILE ENTERPRISE

At first glance, agile methods appear to be more ad hoc than traditional waterfall processes. In practice,

however, they are much more rigorous. It is important to understand that your organization will still require talented people in order to implement these kinds of changes. A key component for success is to start small with a couple of agile coaches, who can do an assessment of where you are and how best to start. The idea should be to eventually make the coaches obsolete, as your organization adopts and implements these methods across your agile enterprise.

## REFERENCE

1. Larman, Craig. *Agile and Iterative Development: A Manager’s Guide*. Addison-Wesley Professional, 2003.

*Dan Murphy is a consultant working for the Canadian federal government. Mr. Murphy spent 10 years working with IBM in Ottawa in various sales and marketing roles. After leaving IBM, he worked at Cisco Systems, first in sales and marketing, then moving into consultative and public relations roles, articulating Cisco’s IT best practices to senior government clientele.*

*Mr. Murphy can be reached at DJM Systems, 337 Crichton Street, Ottawa, ON K1M 1W3, Canada. Tel: +1 613 852 4827; E-mail: damurphy@sympatico.ca.*

*Dave Rooney is the principal of Mayford Technologies Inc., and he has been using agile development with great success since early 2001. He was initially drawn to agile development by its focus on delivering business value to the customer. Mr. Rooney has been developing business applications with Java/J2EE, C/C++, and PowerBuilder for the past 17 years.*

*Mr. Rooney can be reached at E-mail: dave.rooney@mayford.ca; Web site: www.mayford.ca.*