

The Top 3 Cost-Cutting Mistakes CIOs Make And How to Avoid Them

Evan Campbell – MBA, CISA, CSM, Vice President of Professional Services, Rally Software Development

A Cautionary Tale – How Not to Cut IT Costs in a Downturn

Years ago I worked as a Product Line Manager, responsible for development of six products, in the IT organization of a \$17 billion dollar public firm. Immediately following senior corporate and IT leadership changes, we implemented an aggressive cost cutting program using the tactics I will describe in this paper.

Within two years, a once relatively efficient and well-aligned IT shop reached a point where it was incapable of delivering projects and running the systems the business depended on. By the end of this transition, most employees were assigned to three to six simultaneous projects, often at two hundred percent of capacity. Employees didn't know which projects mattered most, and eventually stopped expecting projects to succeed, as there was no means to remedy impediments.

The largest and most important project in the company, one involving hundreds of IT employees and contractors, was a risky new Customer Care system being implemented on a commercial CRM platform that had never been used this way. Senior Directors and above believed the green status reports they received weekly from the consulting company leading the implementation. The non-executive employees in IT knew the system wasn't functional, but the IT management team trusted the consulting company more than employees. The system broke the minute it went live, and for more than a month customer accounts couldn't be serviced and retail stores couldn't add new customers. For months after that, major outages and account data corruption caused billing errors and service problems for thousands of customers. The US Government levied investigations and fines for regulatory violations because angry customers couldn't cancel their service to join competitors.

Other departments of the company had similarly spectacular declines in effectiveness and customer service. In three months, millions of customers cancelled their service and switched to competitors; an unprecedented level of churn. The stock price went down to a third of its highs. The company plunged from being the largest, fastest growing, most innovative firm in its industry, to being the only major firm to lose subscribers and earnings in this period. It ceased to be the acquirer of choice, and was bought by a competitor at a price that reflected its distress and disarray. This transformation from industry leader to firesale disposition was accomplished in less than a dozen quarters.

So here we are again. The current economic downturn has resulted in contraction of IT Department budgets and a mandate to allocate resources to only the most critical projects, then execute flawlessly. In this paper, I outline some bad, but not uncommon tactical reactions to falling budgets, and describe problems those reactions generate. I also recommend a proven alternative for delivering more value with smaller budgets using an Agile and Lean framework that avoids costly project failures, maximizes ROI of projects undertaken, and lets us have a more adaptable and flexible portfolio of projects.

Introduction - The Need to Do More With Less

A recessionary economy puts pressure on companies to maintain earnings while revenues are static or shrinking. Many of the efficiencies business leaders seek to achieve depend on improved systems and streamlined processes, resulting in increased demand for IT projects at the exact moment when IT departments are most capacity constrained. At the same time, the consequences for waste, inefficiencies, and project failures have become extremely high.

In tough times, the key to improving the strategic value of IT is to resist the pressure to retreat to a "keep the lights on" defensive posture. CIOs must increase relative portfolio investments in projects that raise the impact and yield from IT. In other words, deliver more value, faster and cheaper, while ensuring troubled projects are fixed or terminated. But how is this possible?

The answer is to adopt Lean thinking and Agile software development practices in order to run IT as an Agile portfolio. Industry studies of the ROI of Agile projects are now widespread and prove Agile practices deliver a tremendously positive impact on time-to-market, productivity, customer satisfaction, and the quality of products delivered. With a systematic focus on delivering value and removing waste, Agile is especially compelling in tough financial times. Surveys on Agile adoption rates show that mainstream IT organizations are adopting Agile in droves, so all of our competitors will be reaping the efficiencies of Agile shortly.

So the question becomes: "Can we afford NOT to implement Agile now?"

Common (Mistaken) Tactics in an Economic Downturn

Typical responses I've seen to an economic downturn include outsourcing and offshoring, across-the-board budget cuts, and centralization of IT functions into an IT "service bureau." While companies may see some initial budget savings, each of these tactics prove to be short sighted. Each brings its own set of problems, but in combination, these cost-cutting tactics are highly damaging, hollowing out the organization's ability to maintain market share and increase competitiveness.

Cost Cutting Option #1: Ship it offshore

I've been the customer and manager of several offshore teams, and working in Professional Services I've been the subcontractor performing work in our offshore center for my customers. Of course, it's the promise of huge labor savings with hourly rates less than half those of US developers that is initially so attractive. I've found that while large-scale non-core or "utility" projects can eventually achieve savings through outsourcing, even those cost much more the first year or two of the transition. Cost savings here are possible, but are a longer term proposition at best.

However, when building core internal business applications or customer-facing systems, I've seen that that the total cost of development isn't reduced at all with off shoring. In my experience with systems involving complex or evolving requirements, the man-hours effort and calendar time required to deliver a release has been as much as double that of local development teams, a year later. Here are the main causes:

- Our offshore development facilities generated much higher levels of waste and rework due to defects and wrongly implemented features, including many features that technically "met the spec" but weren't usable. We had to mitigate these issues with spending on additional levels of oversight, reviews and communications.
- Our offshore teams required much more detailed specifications than were needed previously. A high-level requirement that would have been sufficient for our local developers who know the business had to become a detailed step-by-step specification describing the "how" instead of the "what" and "why."
- Our outsourcing contracts erected impediments to quickly responding to our feedback and market changes. This further hurt project ROI and delivery time.

Together these obstacles increase costs and erode value delivered. But, because many of them are "soft" costs, they are less obvious than lower dollars paid per developer-hour.

While I've experienced successful offshore initiatives, I've learned the hard way that it's expensive, and only appropriate for non-core projects that aren't tied to improving market share. Consequently, I have shut down several failed offshore operations, including ones we launched with high expectations and confidence after "learning our lessons" on previous failures.

Cost Cutting Option #2: Bottom 10 or Across the Board?

IT organizations are accustomed to demand for services outstripping their capacity to deliver. Many CIOs, after diligently implementing efficiencies for years, are understandably stunned when the CFO asks for a 10% or greater cut in the IT budget as part of the company's response to the economic crisis.

In the budget cuts I've managed, I've never had the luxury of obvious expendable projects or functions to kill in order to find a painless 10% savings. Once a project is approved and in process, there are stakeholders passionate about its preservation. How do we cut and avoid those ugly confrontations with our customers? IT management has two choices: a) lay off the bottom 10% of staff, or b) cut all budgets and initiatives equally to spread the pain "fairly" across all divisions and projects. I've worked in environments that alternated cycles of both. The risk here is that management fails to also address workload, thus explicitly or implicitly asking remaining employees to make up the difference with longer hours. As time passes and overtime becomes unsustainable, morale and commitment decay, turnover of the best people grows and quality and project outcomes plummet. Eventually it appears to customers that IT delivers projects ever more slowly and erratically and that quality is a luxury IT can no longer afford.

Instead of falling into the trap, smart CIOs insist that customers clarify priorities among their projects to enable good investment tradeoffs. They insist that appropriate quality and a sustainable pace are non-negotiable constants. They work with their business customers to allocate investments across a portfolio of project opportunities using a transparent capacity and delivery model of IT.

Cost Cutting Option #3: Centralize IT into a “Service Bureau” Model

Another tactic IT leadership implements to cope with shrinking budgets and excess demand is the “Service Bureau” model. Corporate IT emulates an internal IT contracting firm, accepting high-bidder projects to keep the new sources of “revenue” pouring into the department. I’ve seen this tactic give rise to several bad organizational behaviors:

- Rigid and burdensome project change control policies. This way of operating is based on the perception that bureaucratic hurdles and expensive paperwork-driven gates can use change requests to generate additional project budget. It also is conveniently used to deflect blame for excessive delays and costs onto customers who “keep changing their mind.” Besides cost overruns, this limits feedback from customers and makes projects less adaptable and responsive to market forces.
- Highly centralized IT. This is based on the hope that centralized control of resource assignment improves economies of scale in IT utilization. But IT centralization increases separation between customers and the IT personnel serving them. This separation causes both loss of alignment with customer goals and loss of knowledge of the customer’s domain and requirements.
- The creation of strong functional silos where employees report to a Functional Manager who is incented to fill project resource requests. This model’s premise is that managers who aren’t directly vested in projects will support greater resource flexibility for multitasking and reassignment. Instead, this behavior results in employee incentives and advancement that is divorced from project outcomes and customer satisfaction. In turn, this causes several pathologies:
 - Decreased employee commitment to their projects and customers.
 - Reduced business and domain knowledge in employees.
 - Personnel assigned to multiple simultaneous projects, instead of focusing on the most important. This is based on the delusion that if all the customers’ priorities appear to be “in progress,” customers won’t notice sub-par productivity.

Unfortunately, the tactics explained here negatively reinforce each other, creating a powerful system; a death-spiral of project failures, defects, waste, turnover, and unhappy customers.

Implementing Agile Practices – A Strategic Response to an Economic Downturn

While the tactics above may seem like rational responses to the stress of scarcity in the face of unlimited demand, none ultimately help an organization get more value delivered to the business at less cost. After being involved in hundreds of process improvement scenarios, I can say that there absolutely is a better path: Implement Agile development practices throughout your technology organization. It will yield high return on investment and quickly increase value delivered to customers, especially when your resources are static or decreasing.

Agile practices reflect modern best practices from manufacturing and management. Rally Software specifically coaches our clients in adopting an integrated combination of Lean Software Development, Scrum Project Management, and XP Technical Practices, implemented appropriate to the environment. We have experience implementing these practices with software companies of all sizes, corporate IT departments in every imaginable vertical and size up to the largest public companies, as well as US Agencies and State Government.

Hold the Peanut Butter

Spreading our staff thinly across multiple projects crushes productivity. No customer has to hear a firm “no,” so we avoid hard funding choices, but less value is produced per person-hour of labor. Studies show that the context-switching cost of multi-tasking a developer between three tasks is 40 percent, leaving 60 percent of capacity for productive work.¹ In this scenario, schedules slide, project overruns cascade and personnel can’t be released to focus on the next project. Productivity of the whole organization declines. This pattern leads to widespread customer dissatisfaction with IT, and eventually pressure from the Line of Business to simply “outsource the whole mess.”

¹ Clark, Kim B., and Steven C. Wheelwright, 1993. *Managing New Product and Process Development: Text and Cases*. The Free Press.

Technology leaders who want to deliver a maximum-impact IT portfolio in tough economic times implement Agile because Agile organizations:

1. Eliminate project failure risks and unrealized sunk investments.
 - Deliver highest value features first, release them early and often
 - Measure value delivered (outcomes), not secondary artifacts (outputs)
 - Fail fast; terminate investment in projects that don't prove returns
2. Use Lean principles to strip away the waste preventing you from delivering value – what really matters to the customer. Examples of waste include: inventories and queues, task switching and handoffs, excess production, rework, etc.
3. Rebalance their project portfolio at minimum cost and disruption.

1. Eliminate project failure risks and unrealized sunk investments

It is intuitive that accumulating high levels of sunk investment drives up risk severity. However, it is less obvious that it also drives up risk probability. We all learned in Business 101 class that sunk investment, defined as money spent in the past, should never be considered in making asset allocation decisions for the future. But in real business, the inherent ambiguity of software development encourages us to gamble that the money spent to date will somehow prove to be a good investment if we give the team one more chance to get the project on track. Of course we also dread the prospect of telling our CFO the project must be cancelled at a late stage and treated as a total write-off. These are the roots of optimism that have led to many projects that “smelled” troubled at the three-month mark to zombie-march onward for a year or more before finally being killed with the associated loss of assets and the accompanying acrimony and blame.

Deliver the highest value features first, release early and often

A major benefit of Agile development is its emphasis on early and frequent delivery of features to the user community. Early feedback improves an application's fit to customers' actual requirements. This is critical because even the most thoughtfully written specification contains errors, ambiguities, and wrong guesses. Requirements change constantly as our customers learn and the marketplace evolves. Frequent check-ins enable your organization to adjust and adapt at low cost.

When we shorten “time to benefit” by delivering the highest value as early as possible, projects have higher ROI and reduced risk. In long-running, traditional projects, all the financial benefit of using the most important feature is withheld until every last feature, including all low value ones, are completed. A project that yields benefit earlier starts paying for itself earlier as well. Earlier delivery of features ensures the project reaches its self-funding point earlier and therefore consumes less cash.

Measure value delivered (outcomes), not secondary artifacts (outputs)

(Some) Metrics really matter. The most practical method of measuring project status asks one thing before all others, “Was this increment of functionality accepted by the customer?” Following the rules of Agile, the increment is “Done,” if the code, tests, documents and everything needed for release was completed within the iteration. If it meets the customer's intent, value was delivered. Since the team is getting confirmation at least once a month (preferably every two weeks), the risk of the project deviating far from the “happy path” of customer needs and promised ROI is small.

Why Agile's early and frequent deliveries matter so much:

Uncertainty is inherent and inevitable in software development processes and products.
– Ziv's Uncertainty Principle

For a new software system, the requirements will not be completely known until after the users have used it.
– Humphrey's Requirements Uncertainty Principle

It is not possible to completely specify an interactive system.
– Wegner's Lemma

Fail fast to stop projects that waste money

The ability to “fail fast” is another key benefit of Agile. While it doesn’t sound like a wonderful thing, fast failing beats the long and slow kind every time. Empirical measures of project status and value delivered are far less ambiguous and subjective because they reflect value actually accepted by the customer. Projections of a project’s ROI can be based on historical actuals. “Failing fast” allows us to fix or kill a bad project long before sunk investment has grown to a huge write-off.

2. Use Lean to strip away waste

Lean manufacturing practices like the Toyota Production System have revolutionized the modern production of goods. Lean focuses on value to the customer, removing all waste (effort that doesn’t add customer value) and optimizing processes for efficiency and quality. Lean processes let workers pull in work just-in-time and eschews queues and inventory in all its forms, including work in progress. In order to reduce cycle times, Lean emphasizes cross-functional teams performing tasks in parallel, rather than specialization of labor and performing tasks in sequence.

One of the quickest and easiest ways we can deliver more value at less cost is to invest in developing only the right things (and don’t waste money developing the wrong things). Standish Group’s Industry survey indicates that 64% of the features in the average software application are rarely or never used.² That means over half the effort, cost, and maintenance that goes into our average projects is waste! Just imagine the jobs you could save and the investments you could fund from a 50% reduction in effort for value delivered!

3. Rebalance your project portfolio at minimum cost and disruption: Agile Portfolio Management

In order to successfully adopt an effective Agile portfolio management and steering process, an organization can implement a number of immediate changes:

Cut the business case requirements to the essential items

For any given project, a high-level business case is usually good enough to compare its ROI to projects in progress or in your backlog. Keep in mind that a business case is a balance between cost and benefit over a specific time interval. Developing the precision of one side (usually cost) way beyond the precision of the other doesn’t improve the business case.

Fast-track projects that promise incremental deliveries with quick ROI

Avoid making big high-stakes project bets whenever possible. This doesn’t mean you shouldn’t undertake ambitious programs, but collaborate with customers to turn big projects into a series of more measurable increments that will have lower risk and higher yield. After the team and customer have shown the ability to deliver value, scale up and accelerate velocity as needed.

Reconsider the portion of the budget your organization calls non-discretionary

In most organizations “keep the lights on” Operating Expenses can account for half to three quarters of the technology budget. This portion of the budget typically isn’t examined as aggressively as Capital Expense investment opportunities, so it hides waste and returns less value than it should. We’ve been able to show customers that 30-40% of their IT budget slated as non-discretionary Operating Expense is actually discretionary.

How do you know when you have an effective Agile portfolio management culture?

Here are a few basic traits you should see:

- *Failing projects get fixed fast or killed.*
- *Some projects end early because the business case is no longer compelling to continue.*
- *Your organization knows the relative worth of its pipeline of projects, and it can assert that projects in process are more valuable than those in the backlog.*

Reprogram the lowest performing 10% of your budget, at minimum, every two years

Ask the business unit leadership to identify the least valuable expenditures in your technology budget to allocate it to higher yield initiatives. This process will help you deliver more value and will have a positive cultural impact by demonstrating that the organization is committed to aggressively managing the portfolio for value delivery. This is good for all functions in the organization.

Conclusion

At a time when budgets everywhere are being slashed, CIOs are under more pressure than ever to cut costs. In this paper, we've looked at examples of cost cutting tactics that don't work and showed how they can even cause long-term damage to an organization. In contrast, we've also examined how using Agile and Lean principles can increase delivery value, while simultaneously cutting costs in a way that builds strength and innovation. With Agile and Lean, IT can help the business respond and adapt to the speed of the market, enable innovation and keep the business ahead of its competitors.

Some of the suggestions offered may be difficult to implement, and some may be beyond the control of the CIO, but establishing an Agile portfolio management process will help IT lead the company in strategic value delivered per dollar consumed. If IT can help align their customers around measurable value delivered, and set transparent measurable standards for investment allocation and performance measurement, the strategic value IT delivers to the company will be enormous and conspicuous.

About Rally

Rally is the leader in Agile application lifecycle management (ALM) dedicated to making leaner development organizations that respond faster to changing customer needs. According to a study by QSM Associates, teams that rely on Rally's Agile lifecycle management products and services are 50% faster to market and 25% more productive than industry averages. Rally's products were honored with four consecutive Jolt Awards (the software industry's equivalent of the Oscar® award) in 2006, 2007, 2008 and 2009. The company's end-to-end solutions for Agile development also include Agile University, the largest source for Agile training, and Agile Commons, the largest collaborative Web 2.0 community dedicated to advancing software agility.

About the Author

With more than 15 years in the software industry, Evan Campbell has had responsibility for Agile transformations from the team level to the boardroom. Prior to joining Rally, he served as Chief Technology Officer and Vice President of Professional Services at SolutionsIQ, a 450-person IT consulting company. In that role, he was responsible for building and leading the company's Agile Software Development Practice, and its Agile Consulting Practice, among others. As CTO of Versatile Mobile Systems (VMS), a publicly traded software company, he was responsible for Product Development, Professional Services, and IT. Evan joined VMS as a result of its acquisition of Mobiquity, an e-commerce SaaS company where he was Co-Founder and CTO. Prior to these roles in software companies Evan worked in Corporate IT leading development teams and managing software development and systems integration projects. Evan is a Certified ScrumMaster (CSM) and a Certified Information Systems Auditor (CISA). He holds an MBA from Rollins College and an MA in International Affairs, Economics and Finance from George Washington University. Prior to his career in technology, Evan served in the U.S. Army as an Airborne Ranger and Green Beret.