Cutter Benchmark Review

Analyzing IT Metrics for Informed Management Decisions

Vol. 8, No. 4
April 2008

Creating and Managing the Agile Enterprise

Introduction
by Gabriele Piccoli 3

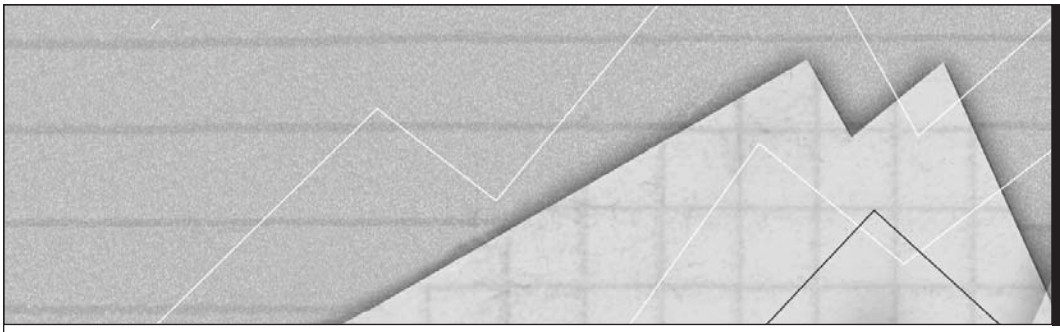
Building the Agile Enterprise: Myths, Perceptions, and Reality
by Alan MacCormack 5

Understanding Perceptions of IT Agility
by Lou Mazzucchelli and Tim Lister 14

Enterprise Agility: Tweaking Won't Deliver Expected Results
by Gabriele Piccoli 19

Enterprise Agility and IT Infrastructure Survey Data
Collected by Cutter Consortium 20

CUTTER CONSORTIUM



Cutter Benchmark Review

About Cutter Benchmark Review

With *Cutter Benchmark Review*, you get in-depth statistics, analysis, and advice on the hottest IT trends and challenges that IT professionals like you face each day.

We Do the Research for You

CBR gives you high-level analyses of the current trends and hot-button issues you face. You'll discover the strategies and tactics of companies worldwide and, most importantly, where they're succeeding and failing.

Your Access to the Experts

Each issue of *CBR* focuses on a different topic, presenting fresh empirical data describing the current landscape. But beyond the statistics, you'll get insight into what the findings mean. Analysis from two different viewpoints — that of a distinguished academic and also an expert practitioner — plus future predictions and their potential impact from editor Gabriele Piccoli provide you with a framework in which to apply the data to your situation. This is insider information you can't afford to miss.

With *CBR*, your organization benefits from the unique opportunity to glean insight from the cutting-edge ideas being studied in some of the world's leading educational institutions, as well as the perspective that only seasoned experts with hands-on experience solving real problems in real enterprises can provide.

CBR gives you a unique opportunity to:

- Look critically, objectively, and through a variety of lenses at business technology issues impacting your enterprise
- Understand how your organization compares to others in your industry and/or around the world
- Understand the potential impact of today's trends on the future health and direction of your company
- Identify key IT initiatives that are worth pursuing and — just as importantly — those you should avoid
- Support your IT initiatives with unbiased research and judicious statistical analysis

Nowhere else will you find a marriage of the academic and practitioner perspectives like you get in *CBR*. Add to this the interpretation and key take-away lessons from Gabriele Piccoli, and you'll find *CBR* is a resource unlike any other available today.

Cutter Benchmark Review® is published 12 times a year by Cutter Information LLC, 37 Broadway, Suite 1, Arlington, MA 02474-5552 (Tel: +1 781 648 8700; Fax: +1 781 648 1950; E-mail: service@cutter.com; Web site: www.cutter.com).

Publisher: Karen Fine Coburn; Group Publisher: Kara Letourneau (E-mail: kletourneau@cutter.com); Managing Editor: Cindy Swain (E-mail: cswain@cutter.com); Production Editor: Linda M. Dias (E-mail: ldias@cutter.com).

Subscription rates are US $477 a year in North America, US $577 elsewhere, payable to Cutter Information LLC. Reprints, bulk purchases, past issues, and multiple subscription and site license rates are available on request.

From the Editor, Gabriele Piccoli

Creating and Managing the Agile Enterprise

While we have not quite scripted it this way, *Cutter Benchmark Review* issues seem to have recently coalesced around some major themes. One is innovation and the role of IT in fostering/enabling it, another one is talent development and staffing — with the looming crisis of talent being the catalyst of our interest. This month's issue also fits within a larger theme that we have been exploring recently: agility. Avid readers of *CBR* will remember our July 2007 issue on agile software development ("Making Agility Stick: What's Working, What's Not," Vol. 7, No. 7), as well as our November 2007 issue on dynamic and improvisational IT capabilities ("Dynamic IT Capabilities: Becoming Nimble Through IT Agility," Vol. 7, No. 11). The July 2007 installment of *CBR* was of course quite pragmatic and focused on the systems development process.

Our concern in that issue was helping readers who had embraced the agile philosophy make sure the approach would stick in their organization. To the contrary, the November 2007 issue was one of the most theoretical issues of *CBR* to date. In that installment, we reported on the research literature focusing on how to mobilize and deploy IT-based resources in combination with other resources and capabilities so as to better compete in increasingly turbulent environments.

This current issue strikes a balance between these two approaches and focuses at the midlevel between the trenches — where software development takes place — and the executive suite — where strategy design takes place. It focuses on the management of the agile enterprise and on understanding how organizations can facilitate and foster agile practices through investments in IT infrastructure and technology practices. As such, the survey our contributors crafted tackles issues of strategy, relative positioning and competition, as well as technology infrastructure, software development methodologies, and IT architecture. This issue is the glue that connects the previous two and completes our trilogy on agility.

To state that agility is critical in today's increasingly turbulent environment is almost a tautology. But for as widely accepted as this notion is, creating and managing an agile enterprise remains an elusive and difficult task. I was reminded of this difficulty in my most recent case study on Hilton Hotels Corporation.

Hilton is a behemoth of an organization with more than 3,000 hotels in more than 70 countries and managing nine brands in every segment of the hospitality industry. Not content with this, it is set to open a new property every two days for several years in the future. Its complexity and diversity notwithstanding, Hilton runs off of a standard enterprise infrastructure built on best-of-breed applications collectively named OnQ — as in "we are ready to serve guests *on cue*." But serving guests "on cue" inherently requires the ability to be flexible and responsive in your interactions with them: every guest is unique, so every interaction needs to be unique. How do you structure an organization that at the same time is able to deliver a consistent level of service — the critical success factor for any global brand — and yet serve individual guest needs? You do it by tackling both aspects of your information systems: the technical and the social. You ensure that your system architectures have inherent flexibility from both a design and a use standpoint. You then develop a culture of improvisational capability within the boundaries of brand-defined service levels — clearly not an easy task!

Without digressing too much, it is clear that an issue of *CBR* on enterprise agility was needed and would prove very valuable. We therefore recruited a team of experts on this subject and issued a survey to benchmark the state-of-the-art amongst our readers. Our academic perspective is provided by Alan MacCormack, associate professor in the Technology & Operations Management Unit at the Harvard Business School and member of Cutter's Innovation & Enterprise Agility team. Alan's work has traditionally focused on the management of technology and product development

in rapidly changing environments, such as the Internet software industry and the computer workstation and server industry. Alan's previous work and current interests make him uniquely suited to bring a new perspective to the notion of enterprise agility. Our view from the field comes from Lou Mazzucchelli and Tim Lister, both Fellows of the Cutter Business Technology Council.

Alan starts us off by making an important point: "The search for agility requires a consistent pattern of decisions to be made across different areas of the firm over time, emphasizing flexibility over efficiency and responsiveness over control." He then provides a definition of IT agility and dives into the survey data, identifying patterns and surfacing important relationships. He discusses the patterns emerging in firms that seek to become agile, those that achieve agility, how they do it, and the systems architectures and infrastructure decisions that seem to foster success.

Lou and Tim take a much narrower focus to the survey and mostly look at IT agility. They immediately tackle the survey data with their characteristically colorful writing style. They first challenge some fundamental notions about IT agility, then they attempt to understand the differences amongst respondents addressing both the organization as a whole as well as the departmental and divisional levels.

Agility remains one of the most discussed and difficult challenges for both strategic and IT management. We at *CBR* have been looking at agility from a technology and strategy perspective. With this issue, we bring it all together and focus on the management of agility. With environmental turbulence continuing to increase, agility will become more important and, most likely, harder to achieve. As you continue the quest, I hope you will agree that, taken together, our three installments of *CBR* on agility provide some very useful guidance.

— Gabriele Piccoli, Editor,
Cutter Benchmark Review

THIS MONTH'S CONTRIBUTORS

Alan MacCormack

Associate Professor, Technology & Operations Management Unit, Harvard Business School

Lou Mazzucchelli

Fellow, Cutter Business Technology Council

Tim Lister

Fellow, Cutter Business Technology Council

COMING NEXT MONTH

Approaches to Green IT and Green IS

AN ACADEMIC PERSPECTIVE

Rick Watson, J. Rex Fuqua Distinguished Chair for Internet Strategy, and Director, Center for Information Systems Leadership, Terry College, University of Georgia; and Marie-Claude Boudreau, Assistant Professor of MIS, University of Georgia

A VIEW FROM THE FIELD

Emily Ryan, Manager of Client Services, international software company IT department

by Alan MacCormack, Associate Professor, Technology & Operations Management Unit, Harvard Business School

Building the Agile Enterprise: Myths, Perceptions, and Reality

As I write this article in the spring of 2008, the need for enterprise agility is once again a topic on the minds of executives everywhere. Stock market gyrations push prices down 3% one day, up 4% the next. Firms that months before had record profits struggle to stay alive. As if the impending recession is not hard enough, new technologies continue to make inroads in traditional industries, challenging old ways of doing business: Apple is the number one US music retailer; Google, while "doing no evil," scares everyone; and a product developed by hundreds of individuals around the world — Linux — is forcing the world's largest commercial software firm to rethink strategy. Today, more than ever, firms must respond to a turbulent environment. How topical then that this issue of *CBR* examines how firms can achieve such agility.

How does one make an organization "agile?" First of all, we must recognize that agility is a capability that results from the complex interplay of many different processes throughout an organization. How fast can you introduce a product in a newly emerging market segment? How quickly can you respond to equipment failures in production? How rapidly can you find and fix a security breach in your Web site? These are all aspects of agility, illustrating the difficulty in making simple recommendations that have immediate benefits along every dimension. The search for agility requires a consistent pattern of decisions be made across different areas of the firm over time, emphasizing flexibility over efficiency and responsiveness over control.

A critical concern for firms seeking greater agility is the need for an infrastructure to support this ability. In this respect, the role of a firm's IT system is critical. These systems are integral to many of the functions a firm performs, including product development, production, operations, and financial management. The ability to respond to changing demands is influenced significantly by the functionality built into these systems. Critically, however, it is also influenced by the organization's ability to make *unanticipated changes* to these systems. We refer to this concept as IT agility. It captures how quickly and effectively a firm can adapt its IT system to meet new demands.

In this issue of *CBR*, we explore the topic of IT agility, reporting data from a survey of 124 organizations. The aim of our survey was first, to understand the extent to which IT agility is an explicit objective for firms, and second, to understand the mechanisms by which this ability is achieved. In analyzing the results, we look both at broader descriptive trends reported by firms and the underlying drivers of these trends, as revealed by the patterns of correlation among variables. While descriptive statistics provide insights into *how* the world looks, correlations help us to understand *why* the world looks this way.[1] In short, these types of analyses help us to develop managerial recommendations in the search for improved performance.

My own interest in this topic stems from past research on a related topic: how firms develop new products in highly uncertain environments. My work in this area revealed that successful firms focus on two key areas: first, they adopt a more iterative style of development process, geared to generating early feedback on product performance that can be used to inform subsequent design decisions; and second, they emphasize the use of modular product architectures, which facilitate the ability to make changes to a design quickly and at low cost.[2] In essence, I found that increased responsiveness in product development could be achieved only by rethinking some of an organization's core processes and structures. The question I have is, is the same true for IT agility?

The findings that emerge are intriguing. While almost 65% of our sample organizations identify IT agility as an explicit objective in developing IT strategy and plans (see Graph 1 in the Survey Data section beginning on page 20), only 26% of them have developed concrete measures of this ability (see Graph 2). We must ask how they know if they are achieving their objectives? This is important given that there are significant differences between how managers perceive they perform and their "real" performance, as captured by some of our survey

variables. While perceptions are strongly tied to reality, as one would hope and expect, they are also driven by the *actions* that firms take and the expected results from these actions. Unfortunately, not all of them work as intended. IT agility, it seems, is an elusive phenomena.

So what *really* counts? The bottom line is that achieving agility requires that firms make investments in application development processes that are more flexible than traditional processes. But they must also design a system architecture that facilitates flexibility by reducing the level of interdependency between different applications in the overall portfolio. To achieve this, it is not enough to adopt high-level methodologies such as service-oriented architectures (SOAs) or layered architectures. Though these methods bring the promise of better structure, we find that they are not correlated with either reduced complexity or increased agility. Instead, firms must attack complexity directly, investing in system redesign efforts to make them more "improvable."

In this article, we examine the extent to which IT agility is a concern for firms and explore why some firms seek agility while others do not. We look at the actions firms take to achieve this ability and examine the drivers of IT agility as revealed by the data, drawing insights into the differences between managerial perceptions and competitive realities. We conclude by looking at what we didn't see in the survey results, which is often just as interesting as what we do see.

IN SEARCH OF AGILITY: WHO WANTS IT, WHY, AND WHAT DO THEY DO?

In the survey we first looked at the extent to which agility is a factor in shaping a firm's approach to managing its IT infrastructure. Our survey reveals that this topic receives a great deal of attention in many organizations. Almost 65% of our sample report explicitly considering the need for agility when developing their IT strategy and plans (see Graph 1). How you view this figure depends upon whether you believe the glass is half full or half empty. While it is heartening to see so many firms addressing agility in their formal plans, it is still surprising that one-third of respondents do not.

Why then, do some firms pay so much attention to agility while others do not? One hypothesis is that the former operate in more uncertain environments, hence they need the ability to react to changing circumstances. For example, consider the situation at Apple as it moved from being a focused computer firm to a firm that sold computers, MP3 players, and smart phones, while also delivering music, movies, and software to these devices. These changes generated demands on its IT systems that could not have been predicted a few years earlier. Apple needed to be more agile, hence it would be expected to emphasize this ability much more than firms in industries with a slower "clock speed."

We used the survey data to test this explanation, with surprising results. The level of *external uncertainty* a firm faces — in terms of understanding customer needs or predicting next-generation product technologies — has little correlation with whether a firm considers the need for IT agility. By contrast, the level of *internal complexity* a firm faces — in terms of the breadth of its product line and the scope of its geographic operations — is a strong predictor of the need for agility. From an IT perspective, therefore, it seems as though agility is viewed as a way to manage an organization's internal challenges, rather than its external ones.

Why might this be so? Perhaps it is because the challenges of developing and deploying IT systems are particularly salient in firms that operate across multiple industries and geographies. If these characteristics change, through the addition of new products or locations, firms must ensure that their IT systems are quickly adapted to suit. Even so, the lack of association between the need for agility and external uncertainty still comes as a surprise. It implies that firms must pay greater attention to understanding how the demands of the external context should be met through investments in their IT infrastructures.

We next looked at actions taken by those organizations concerned with agility. One of the big surprises was in what they do *not* do: measure how they perform! Only 26% of our respondents report having explicit measures of IT agility that are monitored on a regular basis, less than half of those that explicitly consider the need for this ability in developing their IT strategy and plans (see Graph 2). One must ask how these organizations know if the steps they are taking lead to improved performance.

Part of the reason for the absence of metrics may lie in the complex and multifaceted nature of agility, which makes it difficult to agree on how it should be measured. Any set of measures will inevitably be viewed as incomplete. Indeed, we have observed many arguments inside firms as to what exactly it means to be "flexible" and from whose perspective this should be measured: the IT department or downstream customers of its services. But neglecting to define *any* measure of agility merely avoids the problem. It is far better to engage in a strategic discussion of the ways in which agility

creates value, given this will typically expose a number of hidden assumptions that must be made explicit before progress can be made. Clearly, this is an area for greater attention.

Looking at the management practices adopted by organizations that emphasize the need for IT agility, we see clear patterns (see Table 1). They are more likely to use agile processes and to adopt high-level design methodologies that emphasize the need for modularity. With respect to the former, our survey reveals that firms that consider the need for agility adopt agile methods in 36% of their application development projects; whereas in firms that are less concerned with this ability, the figure drops to 14%. With respect to architectural design practices, we examined both the commitment to an SOA as well as the number of layers of abstraction identified in the IT system architecture (a proxy for the degree to which the functions within each layer have been isolated from one another). Clear differences are found on both measures. Two-thirds of the firms that emphasize the need for agility have committed to an SOA; by contrast, 25% of the remaining firms have adopted this methodology. In addition, those firms concerned with agility, on average, design an extra layer of abstraction into their IT system architectures.

Clearly, firms that seek greater agility seem to be doing the right things. They adopt processes that are more responsive than traditional waterfall models and hence are better equipped to respond to changing requirements. And they adopt architectural design methods that emphasize the need for modularity, which should allow changes to be made to one part of a system with minimal impact elsewhere. Without a way of measuring performance, however, it is hard to know if these choices are working. That is what we set out to understand in the second part of our survey.

ACHIEVING IT AGILITY: MANAGERIAL PERCEPTIONS AND COMPETITIVE REALITIES

The main aim of our study was to explore the factors that lead to improved performance with respect to IT agility. We captured data on both manager perceptions of performance, as well as several measures reflecting an organization's actual performance. Given that there are no widely accepted measures for the latter, we focused our attention on two areas: responsiveness to unforeseen problems and responsiveness to changing demands. In particular, we captured data on the time taken to resolve problem/bug reports; the time taken to implement new feature requests; and the percentage of new application development projects that are completed in less than six months. We provide descriptive data for these measures in Table 2. We found strong correlations between these measures of performance (see Figure 1). This makes us confident that we are capturing a central component of IT agility and doing it in a way that is both robust and repeatable.

We looked first at the factors that best explain managers' perceptions of their organization's level of IT agility, relative to the competition (see Graph 3). (Note that Lou and Tim have an interesting take on this issue

Table 1 — Development Practices Split by Firms' Emphasis on IT Agility

	High Emphasis on IT Agility	Low Emphasis on IT Agility
Percentage of development projects that use agile method	36%	14%
Percentage of firms adopting a service-oriented architecture	66%	25%
Number of layers designed into the system architecture	3.57	2.57

Table 2 — Descriptive Statistics for Actual Measures of IT Agility

	Mean	Median	Min.	Max.
Percentage of development projects completed <6 months	39%	30%	0%	100%
Time to implement a new feature (weeks)	7.3	4.5	1	30
Time to resolve a problem/bug (weeks)	2.5	1.3	1	18

as well.) These perceptions are strongly correlated with our real measures of agility, as one would expect. However, we also found that these perceptions are driven as much by *what an organizations does* as by how well it performs. That is, managers believe their IT organizations are more agile to the extent that they measure this ability and deploy practices thought to improve it. Unfortunately, not all these practices have the impact they might expect. In Figure 2, we show the correlations between perceived and real measures of IT agility as well as the factors influencing each of these measures as revealed by our analysis.

Let's take a look at those factors correlated with perceptions of IT agility but *not* with our measures of actual performance. The first is whether the firm has defined explicit measures for IT agility (see Graph 2). Those that have measures believe they are more agile than others, yet in reality, they perform only at the same level. Unfortunately, while measuring performance is often the first step on the road to improving it, the mere act of measurement does not mean that you are better at it.

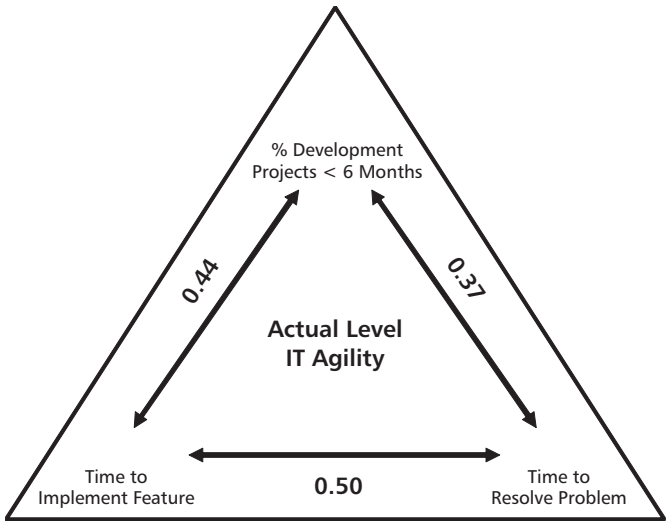


Figure 1 — Correlations between actual measures of IT agility.

The second factor associated with perceptions of agility, but not with actual performance, is the number of layers of abstraction identified in the IT system architecture. Respondents appear to believe that a greater number of layers implies a better structured and more modular architecture. This makes sense, given we noted above that firms concerned with agility tend to adopt more layered architectures. Intriguingly, however, committing to an SOA (see Graph 4) is *not* correlated with perceptions of agility, despite the fact that those organizations concerned with this ability are more likely to adopt such initiatives. Perhaps managers view SOAs as a prerequisite for attaining agility but not as a way of directly influencing this ability? The lack of correlation between adopting SOAs and our actual measures of performance suggests that this intuition may be spot on.

We next examined the factors correlated with both perceptions of IT agility and actual performance. We found only one factor with this pattern, though it is one of the most important in our survey: the percentage of a firm's application development projects that uses an agile process. Across our sample, organizations report using agile methods in 27% of projects. Organizations that report a greater use of such methods, however, have higher perceived levels of agility (3.8 versus 2.8 on a

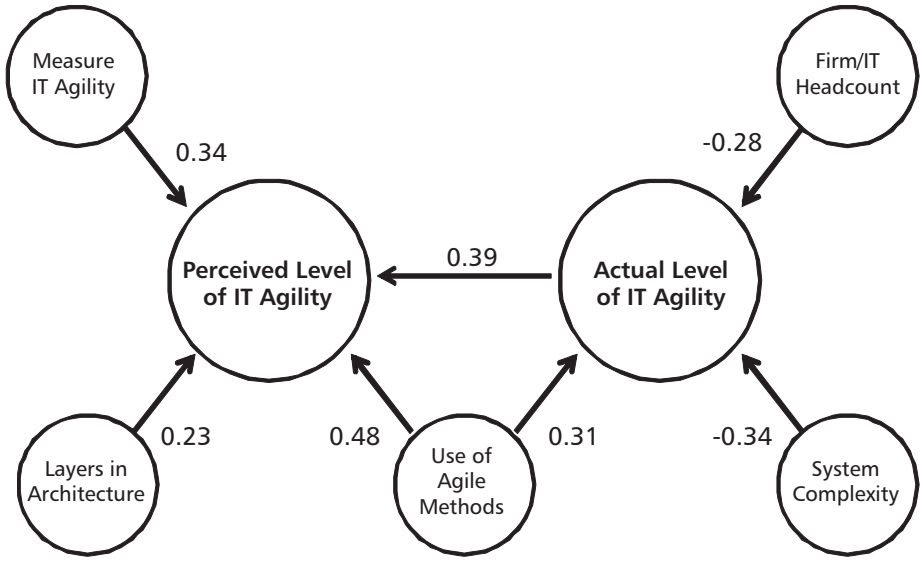


Figure 2 — Factors influencing perceived and actual measures of IT agility.

five-point scale) and higher levels of actual agility, as illustrated through the descriptive statistics shown in Table 3.[3] In sum, this variable is the single most significant factor in explaining perceived levels of IT agility and the second most significant factor in explaining our real measures of IT agility.

As we dig deeper into these results, an interesting pattern emerges. We observe that penetration of agile methods has a higher correlation with perceived levels of IT agility than with actual performance. That is, a greater use of agile methods is associated with an increase in perceptions of agility that exceeds the "real" increase that could be attributed to these methods. Put bluntly, organizations believe they are more agile *because* they have adopted agile processes. This is a subtle point. We are not saying agile processes don't work — in fact our results suggest definitively that they do. What we are saying is that the mere adoption of agile processes is not what makes the firm agile. Agility is achieved only via a coherent set of actions taken across a firm. Adopting agile processes without these supporting actions is likely to lead to disappointment.

In order to tease apart these effects further, I conducted a secondary analysis of the results (indulge my academic interests for a second). I removed that part of the variation in the use of agile processes that was associated with a real performance gain and hence could be explained by the correlation between actual and perceived performance. I then examined the remaining variation — the part not associated with real performance gains — to assess its relationship with perception. The findings support the conclusions made above (see Figure 3). Perceptions of agility are higher to the degree that a firm uses agile methods, regardless of whether these methods lead to real improvements! The size of the correlation is almost equal to that associated with real improvements.

The implication for firms is clear. While agile processes often lead to better performance, they certainly do not guarantee it. Hence firms must not be fooled into thinking that they are agile just because they adopt such methods. Perception and reality are not perfectly aligned. Indeed, there are other factors that firms do not perceive as important that turn out to be just as powerful in explaining performance.

Finally, we turn to look at those factors that explain a firm's actual level of IT agility but that are not correlated with managers' perceptions of performance. At the top of the list is a variable we have not yet mentioned: system complexity. This captures the degree to which making major changes to one application requires that consequent changes be made to the others. That is, it measures the degree of interdependency between the applications in a firm's portfolio. Our analysis reveals that this is the strongest predictor of a

Table 3 — Differences in Actual Levels of IT Agility Split by Use of Agile Methods

	Low Agile Usage	High Agile Usage
Percentage of development projects completed <6 months	32%	52%
Time to implement a new feature (weeks)	8.40	5.34
Time to resolve a problem/bug (weeks)	2.83	1.98

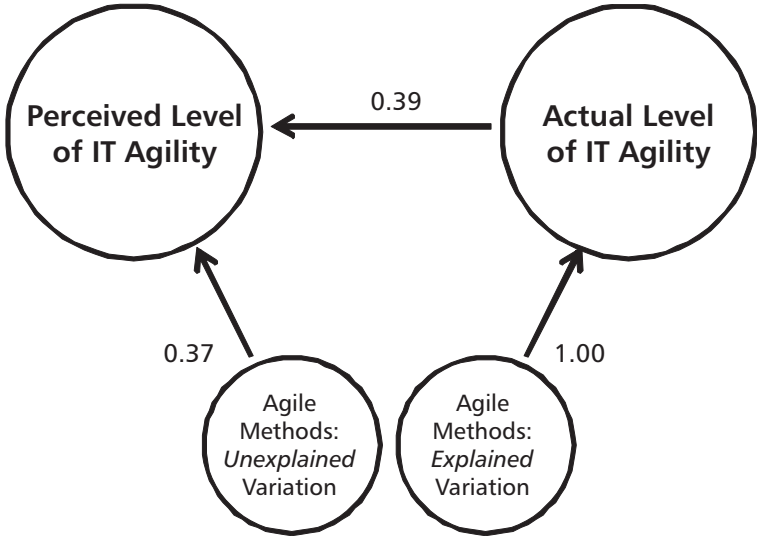


Figure 3 — The impact of agile processes on perceptions and reality.

firm's actual level of IT agility. Organizations that possess systems with high levels of interdependency are much less agile than others. This finding is illustrated by comparing the differences in performance between firms that report having different levels of system complexity (see Table 4[4]).

How can system complexity have such a significant impact on actual performance, yet so little influence on perceptions of IT agility? In my view, the answer to this paradox lies in managers' beliefs that the application of well-known high-level design methodologies will solve the challenges of complexity, without the need to understand what is going on "below." Consider that, in this survey, we asked managers about the use of SOAs as well as the number of layers of abstraction in the system architecture. These measures capture attempts to better structure IT systems from the top down. Indeed, the latter measure is correlated with managers' perceptions of IT agility. Yet while these methods often have compelling benefits in terms of overall performance, they tell us relatively little about potential interactions between applications at the ground level. This is especially true for the legacy systems that an organization possesses, which may have been designed to meet different priorities, are often hard to understand, and typically prove more difficult to change.

This dynamic was highlighted in a recent meeting with managers at one firm who believed their system was highly modular. To illustrate, they mapped out the design on a whiteboard, with rectangles representing each of the major modules. The design was impressive; the conceptual thinking behind it, flawless. But these managers had to admit that they had no idea of the real level of interaction between modules at the level of the actual software code. The subsequent analysis revealed major interdependencies between almost every part of the system.

The challenge is that high-level designs *always* look modular. But the real level of modularity emerges only from myriad individual design and configuration decisions made during implementation. We are not saying that these decisions are made with malicious intent. In fact, the situation is quite the reverse. Creating dependencies between different parts of a system often makes it better at what it does today (e.g., by eliminating redundancies or increasing its speed of operation). But in doing so, it typically makes it harder to change the design thereafter.

So what can organizations do when methods such as SOAs and architectural layering do not guarantee the level of modularity needed? Our results suggest that organizations must attack the complexity of their IT systems in a more direct fashion. Unfortunately, this is extremely hard to do. It requires that coherent actions be taken across the application portfolio to reduce dependencies between them. The challenge is magnified by the lack of visibility we have into the dependencies between applications in a modern IT system. Consider your own organization: Do you know the relative level of complexity or modularity in your IT system? *How* do you know? Clearly, we are in need of new tools that can reveal system structure and guide managerial actions to make systems more "improvable." I expand on this idea below, using insights from my ongoing research.

Another factor that predicts a firm's actual level of IT agility but that is not correlated with perceptions of agility is the size of the organization. Larger organizations, as measured by total firm headcount or the size of the IT staff, tend to be less agile than smaller organizations. I was surprised that size is not correlated with lower perceived levels of agility, given managers in large organizations are often concerned about the impact of bureaucracy on responsiveness. But I did not have a view either way on the relationship between size and actual agility. One could argue that large firms have more resources, hence can potentially respond more quickly to changing demands. On the contrary, more resources can often be a disadvantage, given the inertia that comes from the consequent increase in systems for planning and controlling these resources. The latter dynamic is what we appear to see in this survey. In fact, our data reveals that the negative correlation

Table 4 — Differences in Actual Levels of IT Agility Split by Level of System Complexity

	High System Complexity	Low System Complexity
Percentage of develop. projects completed in <6 months	27%	46%
Time to implement a new feature (weeks)	9.54	6.23
Time to resolve a problem/bug (weeks)	3.33	2.29

between size and agility is greater for the size of the IT staff than for total firm headcount, indicating that "diseconomies of scale" may stem more from larger IT departments than from larger firms per se.[5]

ARCHITECTURE, DESIGN, AND IMPROVABLE SYSTEMS

The critical importance of system complexity in explaining IT agility mirrors the results I have found in parallel research on the topic of software architecture.[6] It is helpful to consider the insights from this work, as they inform the challenges we see here. In this research, we have developed ways to visualize and measure differences in architecture by analyzing the dependencies between components in a design. In software products, the components are defined as source files. While there are several different types of dependency between these components, we focus on the most significant one: function calls (requests by one component to access functionality in another). We can display the dependencies between system components in a square matrix (see Figure 4, and later, Figure 5). By analyzing the pattern of these dependencies, we can measure the potential for a single design change to propagate through the system. We call this measure the system's "propagation cost."

The first insight we gained in this work is that systems that perform very similar functions often have very different architectures. This is surprising given one would think that architectural choices are driven mainly by the tasks a system is intended to perform. Consider the two systems in Figure 4, which perform the same task, but have very different designs. System A has a low propagation cost, while in System B it is much higher. The implication is that it is much more difficult to make changes to System B, given each change can potentially affect more of the design. In System A, by contrast, the loosely coupled design facilitates making changes. Indeed, in our work, we have demonstrated that greater interdependency makes a system harder to adapt, harder to maintain, and harder to augment. In essence, interdependency is the enemy of agility.

The second insight we gained is that differences in architecture often stem from differences in the way that development is organized. For example, we observed that smaller, colocated teams tend to develop more tightly coupled architectures than larger, more distributed teams. While all teams doubtless wish to develop designs that perform well, the extent to which they create and leverage dependencies between different components differs. In essence, we find that products tend to mirror the organizations that develop them, an insight known as "Conway's Law" that dates back to the 1960s.[7] Firms must understand the biases that stem from the way that their IT organizations are structured in terms of any possible predispositions toward developing certain types of architecture.

The final insight we gained is that increasing system responsiveness often requires major redesign efforts, which have the explicit aim of reducing complexity (i.e., increasing modularity). In programming circles, this activity is known as "refactoring," wherein a program's structure is improved without changing its results. The challenge for firms is that they often believe refactoring efforts are a waste of time and effort, given no new

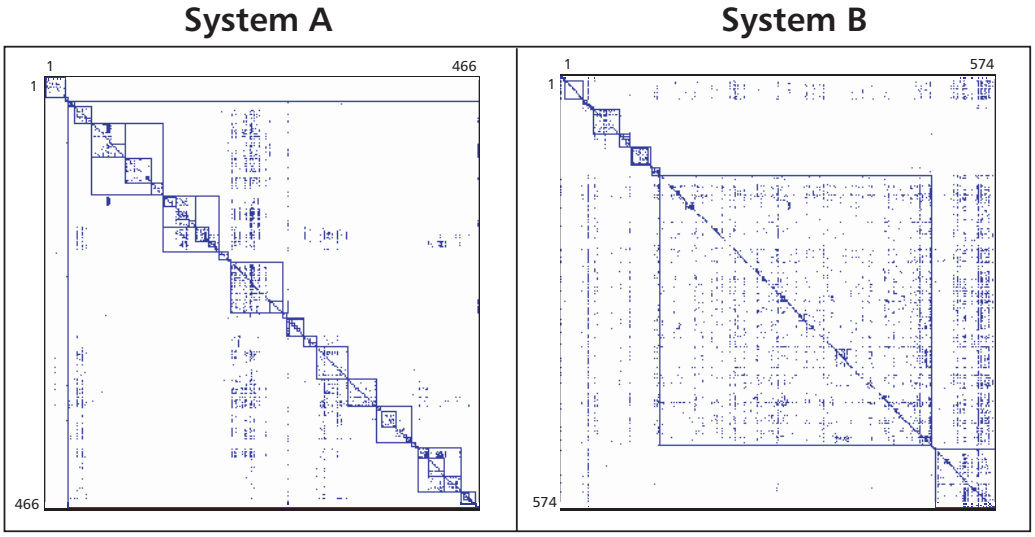


Figure 4 — Comparing the architectures of two systems that perform similar functions. (Adapted from Alan MacCormack et al., Harvard Business School Working Paper [see endnote 6]).

functionality is added to the system. They fail to realize that these efforts are often a prerequisite to making a system more improvable. These efforts create value, but this value comes in the form of an option to make changes to the design *in the future*.

To illustrate, consider what happened when Netscape released the source code for its Mozilla Web browser in 1998, in the hope that outside contributors would help add new functionality. The problem was that the existing design was very complex, meaning it was difficult for developers to make contributions without affecting many other parts of the system. In the fall of 1998, a small team of programmers redesigned the code base with the sole aim of making it more modular, hence easier to change. The result was striking (see Figure 5). The team significantly reduced both the number of dependencies and the system propagation cost, improvements which led to an increase in the number of contributors. Their efforts eventually led to the introduction of Firefox, a Web browser that has gained both critical acclaim and market share.

"NO RESULTS" AND INTERESTING CAVEATS

It is worth noting what we did not see in our analysis, while remembering that this does not mean these factors are unimportant. We may not yet have enough data to reveal the more subtle relationships that influence agility. In addition, survey measures are notoriously noisy, making it difficult to detect all of the signals that lie within the data.

We did not see a relationship between a firm's level of IT agility and measures of its IT strategy. We examined whether a firm tends to buy packaged solutions versus develop custom (bespoke) solutions and whether it tends to adopt integrated solutions versus best-of-breed solutions (see Graph 5). I was surprised that these factors did not impact agility, given there are many arguments that they should. Yet this may be a positive sign, highlighting that achieving agility is not dependent on one particular approach to configuring a firm's IT investments.

We also did not see a relationship between a firm's level of IT agility and the percentage of application development work that is outsourced. Again, this is somewhat surprising but is consistent with the notion that outsourcing can impact agility in multiple ways, some good and some bad. Having the ability to quickly access external resources without committing to full-time staff can be an advantage when responding to varying workloads. On the contrary, effectively responding to new feature and application demands often requires a dedicated internal team with a deep understanding of the end-user context. In essence, the relationship between outsourcing and agility is complex and as such will need greater attention before we can arrive at robust prescriptions for action.

CONCLUDING THOUGHTS

Firms that compete in an increasingly uncertain world strive to be agile. In this article, we sought to understand how organizations can facilitate this capability

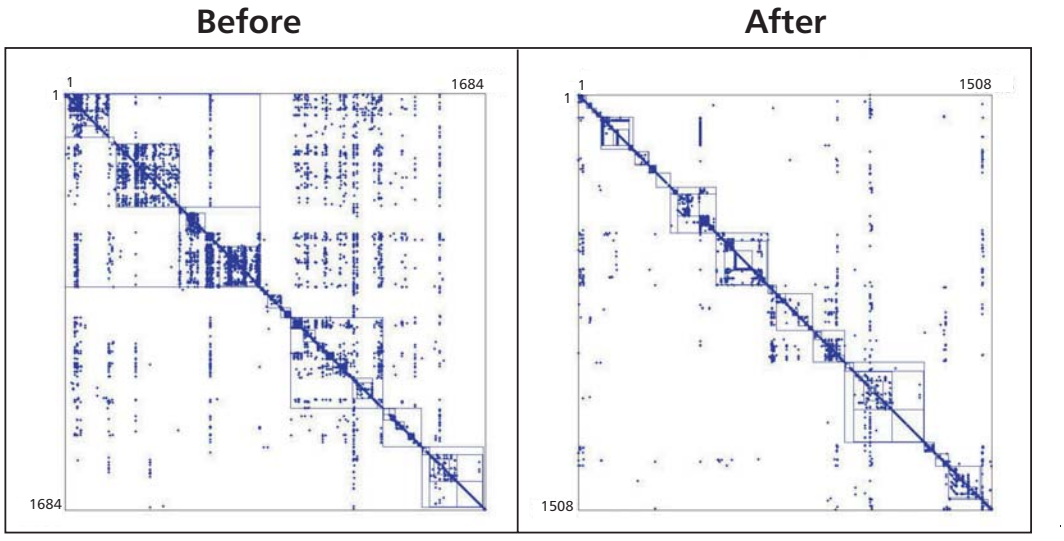


Figure 5 — The architecture of Mozilla before and after a redesign effort. (Adapted from Alan MacCormack et al., *Management Science* [see endnote 6]).

through investments in their IT infrastructure. We discovered that while agility is an explicit concern for many firms, few have worked out how to measure it and fewer still understand the real drivers of performance. Perceptions are driven as much by what firms do as by the level of agility they actually achieve. So what really counts? Agility requires that firms invest in application development processes that are much more flexible than traditional methods. But they must also design system architectures that facilitate this flexibility by reducing the level of dependency between different applications in the portfolio. Critically, these factors are likely to be complements; doing one without the other may constrain a firm's ability to improve. Our results represent a first step in helping to make the phenomenon of IT agility a little less elusive.

So what first steps should firms take? Based on the results of this survey, let me suggest six areas for focus:

1. **Begin a strategic discussion on what aspects of IT agility are most important.** But center this discussion on *external* demands, assessing the level of responsiveness needed to cope with changing market, technology, and regulatory conditions.
2. **Put a stake in the ground and begin to measure the attributes you identify as being important.** You won't get it right the first time, but agility is about the need for change, so it's natural that as your abilities evolve so will the relevant measures.
3. **If you haven't done so already, begin piloting agile development methods.** As you do so, measure the results achieved. You should see improvements along performance dimensions you care about.
4. **Dig a little deeper into understanding your "real" system architecture.** While committing to SOA and architectural layering are good first steps, you must also measure their impact at the ground level. Don't be satisfied with high-level block diagrams that seem to say all is well.
5. **Take corrective action where necessary.** If a system is too complex, it's often worth investing in refactoring. Software systems tend to outlive the tenure of their creators, so if problems aren't fixed at an early stage, you live with the results for a long time.
6. **Make agility an integral part of your decision-making processes, at both a strategic and tactical level.** This is critical, for greater agility requires that firms make investments today that create the option to do different things tomorrow. Every decision has an immediate and a longer-term impact, so firms must balance the two.

Do these things and you are well on the way to building a more agile enterprise. But critically, you won't fool yourself into thinking you have built one already just because you appear to be doing the right things.

ENDNOTES

[1]The presence of correlation, by itself, does not provide evidence of causality. It is only one step in the chain of reasoning through which a causal relationship is established. In this article, I use causal terminology where I think it is warranted by our research design.

[2]MacCormack, Alan D. "Product-Development Practices that Work: How Internet Companies Build Software." *MIT Sloan Management Review*, Vol. 42, No. 2, Winter 2001, pp. 75-84.

[3]We used the mean level of agile usage to divide the sample into high and low levels of use.

[4]System complexity is measured using a five-point scale. The top two scores are used to group systems of high complexity and the bottom two scores are used to group systems of low complexity.

[5]Note we observe the same pattern with firm revenues and the size of the IT budget. The latter has a more significant negative correlation with IT agility than the former.

[6]MacCormack, Alan, John Rusnak, and Carliss Baldwin. "Exploring the Duality Between Product and Organizational Architectures: A Test of the Mirroring Hypothesis." Harvard Business School Working Paper 08-039, 2008; MacCormack, Alan, John Rusnak, and Carliss Baldwin. "Exploring the Structure of Complex Software Designs: An Empirical Study of Open Source and Proprietary Code." *Management Science*, Vol. 52, No. 7, 2006, pp. 1015-1030.

[7]Conway, Melvin E. "How Do Committees Invent?" *Datamation*, April 1968.


by Lou Mazzucchelli and Tim Lister, Fellows, Cutter Business Technology Council

Understanding Perceptions of IT Agility

The term "agile," which has gained increasing popularity in today's IT and larger business world, is applied as an adjective to a variety of activities, ranging from computer programming to organizational behavior. The word agile stems from the Latin *agere*, an imperative form of the verb *ago*, meaning do or drive (or sometimes "drive back," a meaning that might resonate with harried IT practitioners). Common definitions of agile include phrases like "moving quickly and lightly" and "with quick and easy grace." One dictionary even adds the notion of "quick motion in the limbs," helping to dissuade using the word to describe, say, whales.

But here we are talking about agility applied to the closest organizational construct we have to whales — the business enterprise. One could argue that "enterprise agility" is an oxymoron. But, if we take the "at the limbs" idea to heart, it seems to make sense that we could talk about, for example, the agility of an elephant's trunk; likewise, in the business world we can talk about the agility of some smaller but attached part of a larger enterprise. It remains an open topic in our minds whether agile components can be aggregated, scaled, and controlled in ways that preserve agility. The purpose, however, of this Cutter survey on enterprise agility and IT infrastructure was to collect data to help us better understand how organizations view themselves with regard to IT agility and to understand the factors driving perceived (as opposed to measured) success or failure in the quest for agile IT development.

We received a strong response to the survey, which is perhaps indicative of the sample audience's curiosity surrounding the topic. Our first task in analyzing the survey results is to see if the data leads us to any obvious conclusions. We then look at the IT agility perception versus reality. Finally, we take a step further and examine the relationship between agile methods and problem reports.

SOME FUNDAMENTAL ASSUMPTIONS ABOUT AGILITY

We think of agile methods as a family of approaches aimed at dealing with uncertainty. If there is a high level of uncertainty about what exactly you will end up building, then the agile approaches (all of which call for multiple short iterations with ongoing customer feedback) make enormous sense. With the agile methods, we incrementally steer our way to a useful system. On the other hand, if there is a clear and explicit definition of what needs to be built, then the iterative approach significantly decreases in value, and a preplanned, classic, phased approach may be sensible.

In examining the Cutter survey data, we separated the respondents into two camps: those who responded "yes" and those who responded "no" to the question, "Does your organization explicitly consider the need for IT agility in its IT strategy and plans?" (see Graph 1 in the Survey Data section beginning on page 20). Basically, two-thirds of the survey respondents are in the "yes" camp, and one-third are in the "no" group.

We then looked at predictability of requirements as a key indicator that an organization would be more likely to report agile behavior. The assumption is the more uncertainty you have in your requirements, the more likely you are to use an agile method. We focused on the question, "How easy is it to predict the requirements for your organization's IT systems looking three years ahead?" (see Graph 6). For our analysis, we chose to divide the responses to this question into two main groups: "predictable" and "uncertain." The predictable group is made up of the combined respondents in Graph 6 who chose "very predictable" (5%) or "mostly predictable" (36%); the uncertain group represents those combined respondents who chose "mostly uncertain" (23%) or "very uncertain" (1%). We then further segregated these survey results and the responses to the question mentioned earlier regarding IT agility being considered in the IT strategy. Table 1 shows the relationship between those who have and do not have agility as part of their strategy and the level of predictability in requirements.

Yikes! According to this data, higher predictability leads us to agility, while higher uncertainty leads us away from it. As it is sung in *The King and I* musical, this is "a puzzlement."

Table 1 — Survey Data Segregated by IT Agility in Strategy and Predictability of Requirements

Yes, IT Agility in Strategy	No, IT Agility Not in Strategy
40 of 80: Predictable (50%)	11 of 44: Predictable (25%)
16 of 80: Uncertain (20%)	14 of 44: Uncertain (32%)

Table 2 — Survey Data Segregated by IT Agility in Strategy and Percentage of Outsourcing

Yes, IT Agility in Strategy	No, IT Agility Not in Strategy
17 of 80: 50%+ outsourced (21%)	6 of 44: 50%+ outsourced (14%)
4 of 80: 100% outsourced (5%)	2 of 44: 100% outsourced (5%)

Table 3 — Survey Data Segregated by IT Agility in Strategy, Adoption of Uniform Agile Methodology, and Percentage of Projects that Use an Agile Process

Yes, IT Agility in Strategy	No, IT Agility Not in Strategy
12 of 80: No agile projects (15%)	21 of 44: No agile projects (48%)
16 of 20: 75%+ agile projects uniform (80%)	3 of 4: 75%+ agile projects uniform (75%)

So then we thought that maybe those who are actually doing their own application development work report agile behavior, while those who are outsourcing have little need for agility since they are willing to be at an arm's length with the actual construction process.

Again, we separated the same "yes/no to agility in strategy" groups and then looked at the question, "What percentage of your organization's application development work is outsourced?" We looked specifically at those who had significant outsourcing; for this, we set the bar at 50% outsourced or higher (see Table 2). We even made a category for those 100% outsourced.

Yowzers! (Lou disclaims this exclamation, but agrees with its intent.) Again, this shows the opposite of what we thought we'd find. Heavier numbers reported outsourcing in the agile population, and even four organizations that explicitly consider the need for IT agility have sold the ranch: they are 100% outsourced. Did they explicitly consider the need for IT agility and reject it?

Desperate to leave this "Mad Hatter's Tea Party," we then looked at our yes/no populations in terms of the question, "In projects that use an agile process, does your organization adopt a uniform agile methodology?" (see Graph 7). Responses could either be "yes," "no," or "none of our projects use an agile process." (In the sample as a whole, of those reporting use of an agile methodology, about half reported a uniform methodology and half did not. A bit over a quarter of all respondents used no agile process.)

We assumed that the truly agile would not be lockstepped into a single uniform process; their process itself would be agile. So we also looked at those who reported "75% or more of their projects" in response to the question, "What percentage of application development projects in your organization use an agile process?" (see Table 3).

So much for our assumption: there is a clear bias toward a uniform process for those heavily involved in agile development (when we looked inside that group, Scrum seems the hands-down winner of all the named processes). This apparent process uniformity might spring from immature agile teams who have not yet learned to tweak their process. But we are at a loss to explain the results from 12 respondents in the "yes" population who explicitly consider the need for IT agility but respond that they have no agile projects. One possible explanation for this would be an assumption that an outsourced project was not agile by definition. The overall IT strategy might be considered agile, but the projects are not.

IT AGILITY PERCEPTION VS. REALITY

Perhaps the most important question about agile approaches is "Do they work?" Problem reports per month might be a good indication of development efficacy. The data reported in our survey across all respondents was as follows:

- Problem/bug reports per month
  - Average: 65
  - Median: 20

In the aim of full disclosure, we did a tiny bit of smoothing here as one respondent reported 7,000 problem/bug reports per month, and it was adjusted

down by an order of magnitude (Tim and I also considered sending an emergency team to this site).

If we take the results above as "reality" for our sample, it is interesting to compare this against the perception of IT agility level. Graph 3 shows the data reported by the entire group.

We note a slight negativity among some of the group, in that the "somewhat worse" response is slightly higher than the normal distribution would predict. But, generally, there is a skew toward "better" in the population's self-assessment of IT agility compared to its competition.

We also asked respondents to qualify the size of the group to which their answers to this survey applied: department, division, or an entire company. This response allows us to segment the data by size, as shown in Figure 1.

Responders for companies seem to have a much higher opinion of their IT agility than their division or department counterparts. Perhaps this is because of a more global awareness of agility efforts being undertaken throughout the company. Or, it could be a result of what Tim and his colleagues at the Atlantic Systems Guild call "news improvement"[1] in action: someone high enough in an organization chart to report for an entire company might be the recipient of skewed data from the troops below.

We see that perception of IT agility varies with organization size, but how does that perception match the reported reality? We separated out responses by size and compared their self-ratings of IT agility against the reported actual problem report frequency in Table 4.

When we compare the reported problems per month by rating with the overall average problem reports per group, we can get a sense of the gap between perception and reported reality (see Figure 2).

As shown in Figure 2, department respondents rating their IT agility as "somewhat worse," "about the same," and "much better" submitted problem report rates well below the average for their group (we ignore "much worse" here since there was only one data point). One could speculate that a smaller group might assume, due to limited resources, that it must be worse off than others (the grass is always greener on the other side of the fence). However, this does not explain why those rating themselves "somewhat better" than their competition also reported problem rates above their peer averages.

Our division respondents fared somewhat better in this gap analysis. We see that those who rated themselves "somewhat worse" actually had poorer performance than their peer group average. The group reporting "somewhat better" had better performance than the group average, and the "about the same" responders were close to the average. The apparently more accurate

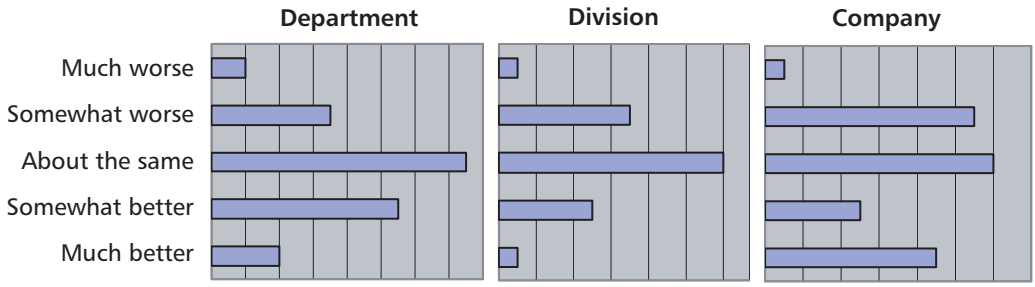


Figure 1 — Self-rated IT agility level vs. competition by group size.

Table 4 — Problem Reports per Month by Agility Rating

	Department		Division		Company	
	Average	Median	Average	Median	Average	Median
Much worse	5	5	20	20	100	100
Somewhat worse	14	7	173	75	83	33
About the same	30	8	117	50	85	20
Somewhat better	103	35	46	38	21	15
Much better	14	7	20	20	22	10

self-assessment may be because the larger group has more experience or more resources.

This suspicion is borne out by the company-wide data. Again we see that those reporting poor IT agility also have higher problem-report rates, and those reporting better agility are experiencing lower problem-report rates.

USE OF AGILE METHODS VS. PROBLEM REPORTS

One survey question asked for the percentage of projects using agile methods. We compared the reported problem-report frequency with reported agile use. In a world of perfect methods, perfect teams, and perfect organizations, we would expect to find a strong inverse relationship between this data (i.e., the higher the percentage of agile projects, the lower the problem-report frequency; we know this might not account for a singular flaming project disaster, but, hey, we're not statisticians).

The responses, as you might expect, do not paint such a definitive picture. The first set of graphs in Figure 3 shows the results for respondents answering "about the same as my competition" for their IT agility level, grouped by response scope.

For department responders, there appears to be no relationship between agile methods use and problem-report frequency. The company responders reported more agile method use, but the data does not indicate the kind of inverse relationship we might expect. The closest result to our assumption is for division responders, but we would still call the data "lumpy" at best. One possible explanation for this is that the problem-report frequency question related to all projects, not just those developed with agile methods, and responders in the "same" category would not be expected to be doing so much agile development to have an influence on overall development performance.

Next we looked at the data for "somewhat better" respondents; here we would expect a higher percentage of agile method projects and therefore a more pronounced global impact (see Figure 4).

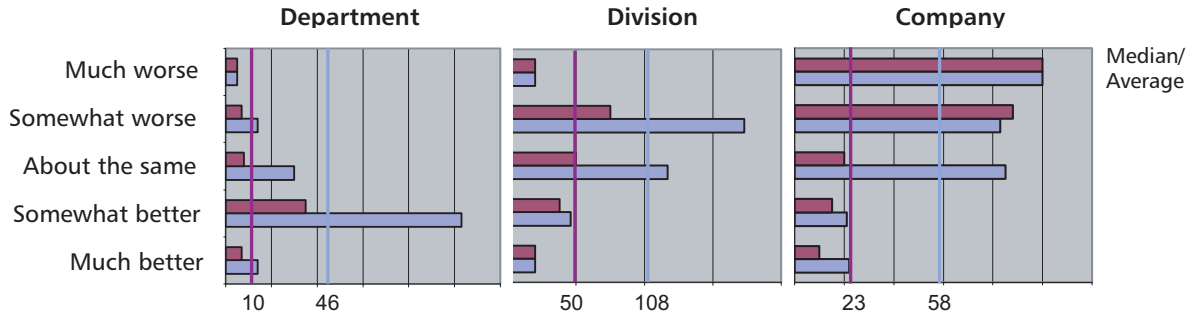


Figure 2 — Agility perception/reality gap using problem-report rate proxy.

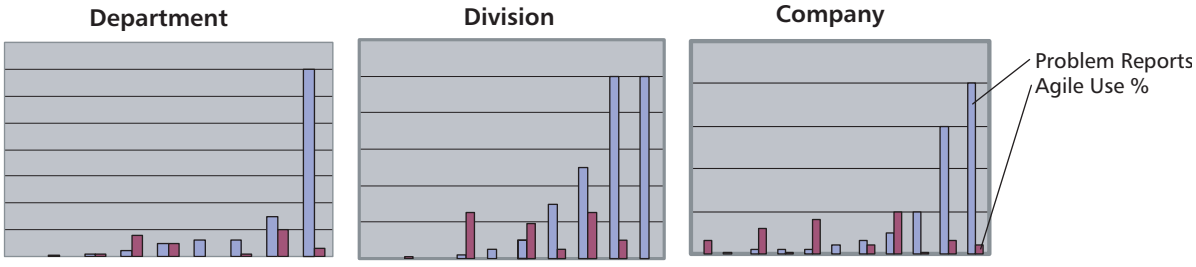


Figure 3 — Problem-report frequency vs. agile use: "same" responders.

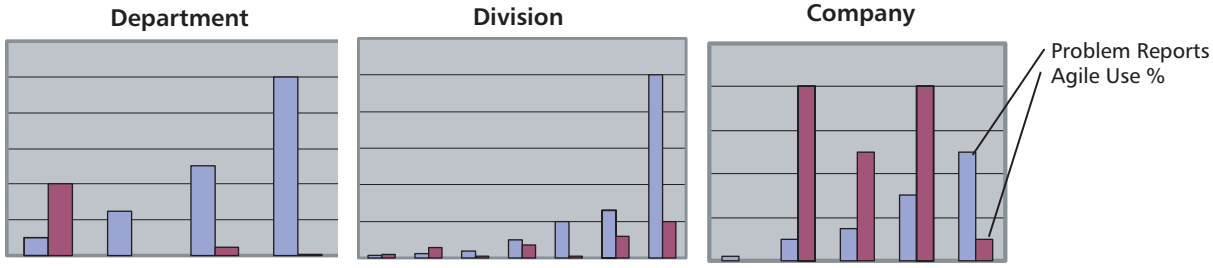


Figure 4 — Problem-report frequency vs. agile use: "somewhat better" responders.

Amusingly, in this data set, we begin to see the relationship we're looking for in department and company responders but not in division responders. In fact, the division data suggests that there is a direct relationship between agile usage and increased problem-report frequency — not a claim we're prepared to make. The best result is in the department data (but with a very small sample size). The company data reflects higher agile method use overall, and an encouraging suggestion of reduced problem-report frequency with higher agile method use.

ONE MAN'S CEILING IS ANOTHER MAN'S FLOOR

It appears that agility takes on very different meanings in different organizations. We need to be extremely careful about making any assumptions when someone tells us they are in an agile organization. As Alan points out in his article, many organizations identify IT agility as an explicit objective, but few (less than 25%) have developed rigorous measures of this attribute.

We think this is the primary source of the inconsistencies in the survey results: organizations are using wildly different metrics and assumptions about what constitutes "agility" in both strategy and tactics.

We looked at the problem reports as a proxy for agile method efficacy and were not able to draw strong conclusions based on the data. However, Alan's analysis of new development and problem resolution times seems to show a promising correlation between agile method use and reduced delivery times on both tasks. Our survey did not explicitly call out problem-report rates by project; this means that large shops with less than 100% agile projects would report a blended number that makes analysis difficult.

Responses to one specific question — "Has your organization committed to a service-oriented architecture (SOA)?" — revealed a significant difference between our yes/no populations (see Graph 4 and Table 5).

Table 5 — Survey Data Segregated by IT Agility in Strategy and SOA Commitment

Yes, IT Agility in Strategy	No, IT Agility Not in Strategy
53 of 80: Commit to SOA (66%)	11 of 44: Commit to SOA (25%)

This is a work in progress, as only 12 of the 64 organizations committed to SOA reported that the number of implemented services equaled the number of published services. But it is also heartening to see that real investment in architecture is occurring. In our opinion, the limiting factor to advances in higher velocity development is most often the architecture, or lack thereof. Building an architecture on to large numbers of legacy applications (or a software shantytown, as dubbed by Cutter Fellow Lynne Ellyn) cannot be done cleanly. The urban renewal of SOA has to take place to liberate the potential of increased productivity and quality through agile development.

SOME OVERALL OBSERVATIONS AND UNFINISHED BUSINESS

In trying to establish some baseline measures of agile behavior, we did not focus the survey on organizational issues that inhibit agility. Identifying and mitigating these issues has been a recent focus of our consulting activity, and we have collected a preliminary though not exhaustive list, which includes in no particular order:

- Conflicting demands of fixed-price contracting and agile development
- Preserving agility across business unit/department/outsourcing boundaries
- Glacial or pathological decision-making processes
- Poor architectures and interfaces that impede progress on legacy systems
- Employee incentive programs that are not aligned with business strategy
- Inadequate funding to support test environments

In summary, we are encouraged by the results of this survey, which show some reasonable awareness and practice of agile behavior. We are particularly encouraged by less than unanimous reports of agile behavior. As observers of trends past, we note that this one has not yet moved to the "gratuitous acknowledgement" stage; debate means that people are taking the ideas seriously enough to vote "no."

ENDNOTE

[1]DeMarco, Tom et al. *Adrenaline Junkies and Template Zombies*. Dorset House, 2008.

From the Editor, Gabriele Piccoli

Enterprise Agility: Tweaking Won't Deliver Expected Results

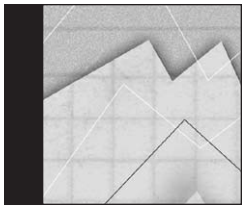
This issue of *CBR* focused on a very important and timely topic: enterprise agility. It is also the third issue of *CBR* to center on agility; these installments looked at agility from various angles: the technical, the strategic, and now the managerial. Enterprise agility calls for structuring the unstructured, for institutionalizing improvisation. It requires that the organization become adept at reacting with speed and precision to changes in the competitive environment, customer needs, and any other change of significant magnitude. The rate at which such changes occur has accelerated (and will likely continue to accelerate) in the increasingly turbulent environment.

In this issue of *CBR*, our academic perspective was provided by Alan MacCormack, associate professor in the Technology & Operations Management Unit at the Harvard Business School and member of Cutter's Innovation & Enterprise Agility team. Alan's work has traditionally focused on the management of technology and product development in rapidly changing environments, such as the Internet software industry and the computer workstation and server industry. Alan's previous work and current interests make him uniquely suited to bring a new perspective to the notion of enterprise agility. Our view from the field came courtesy of Lou Mazzucchelli and Tim Lister, both Fellows of the Cutter Business Technology Council.

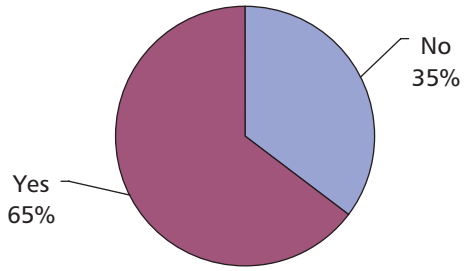
The main insight in this issue for me comes from the notion, introduced by Alan, of interdependencies in the application portfolio. He produces an interesting analytical tool as well: "propagation cost." Perhaps the pendulum theory is at play here (see my comments in the March 2008 issue of *CBR*, Vol. 8, No. 3, for more on the pendulum theory). After heavily investing in enterprise systems and standardization, organizations are coming to realize some of the constraints that such systems create. It would be interesting to find out the degree of complexity inherent in such systems and the propagation costs they entail. If, as one would expect, both are high, this will prove to be unwelcome news from many organizations that will need to reconcile the increasing needs for enterprise agility with past investments in technology and business process redesign.

Another very interesting result that surfaced in our survey is the fact that agility is mostly sought by those organizations that face significant internal complexity rather than environmental uncertainty. At this point I am not willing to completely buy into the notion that environmental complexity and turbulence do not affect the need for enterprise agility. On the other hand, the survey suggests that we need to pay as much attention to internal complexity. Environmental complexity is of course related to internal complexity, and the makeup of the survey respondent population may have influenced its perspective. Nonetheless, I think that this result raises some novel insight that you should corroborate within your own organization. If supported, these findings have implications for how you seek funding and make the case for improving the agility of your infrastructure and practices.

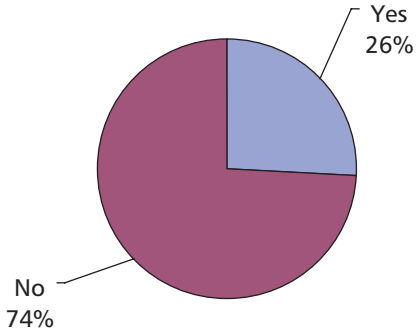
The bad news in this issue of *CBR* is that if you are serious about pursuing enterprise agility, tweaks and spot improvements to your existing infrastructure will not do. Yet, not all news is bad. While refactoring systems simply to lower propagation cost and increase their flexibility is typically a tough sell given that no new functionalities are delivered, applying an option-value approach to this investment may be the solution. We don't have the opportunity here to thoroughly describe this approach (perhaps a good topic for a future installment of *CBR*). Suffice it to say, however, that the option-value perspective enables you to justify an investment based on the ability to launch future initiatives more cheaply, with superior speed, and with lower risk.



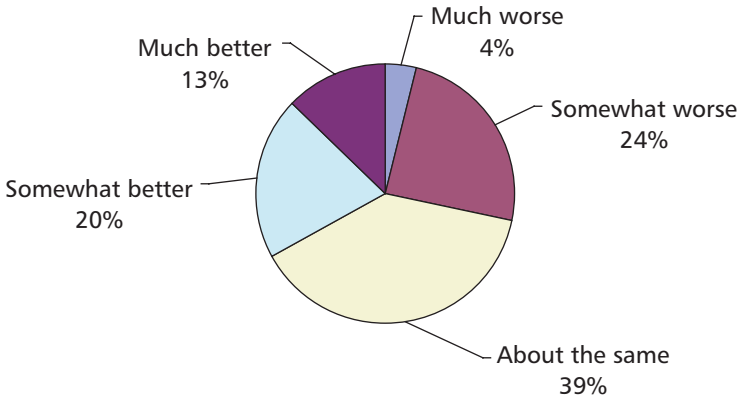
Enterprise Agility and IT Infrastructure Survey Data



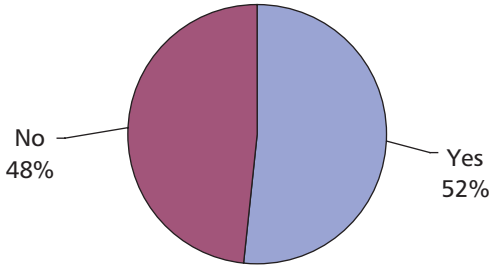
Graph 1— Does your organization explicitly consider the need for IT agility in its IT strategy and plans?



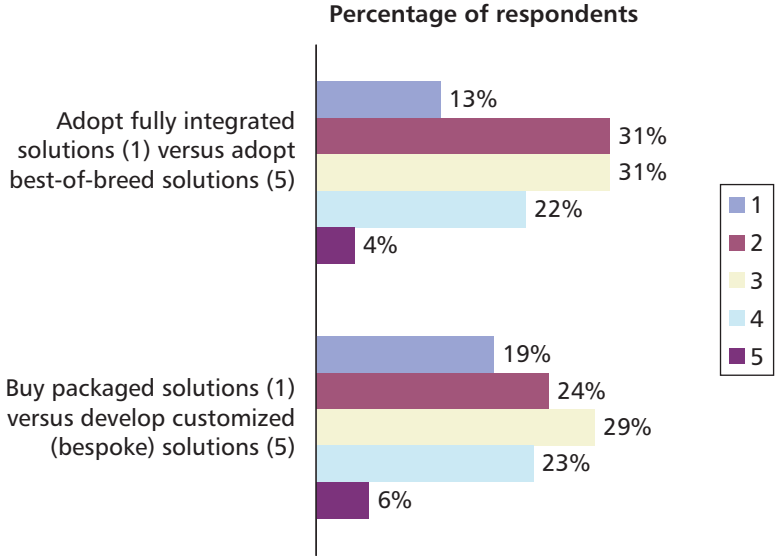
Graph 2 — Has your organization developed explicit measures for IT agility that are monitored on a regular basis?



Graph 3 — How would you rate your organization's level of IT agility relative to your competition's level?



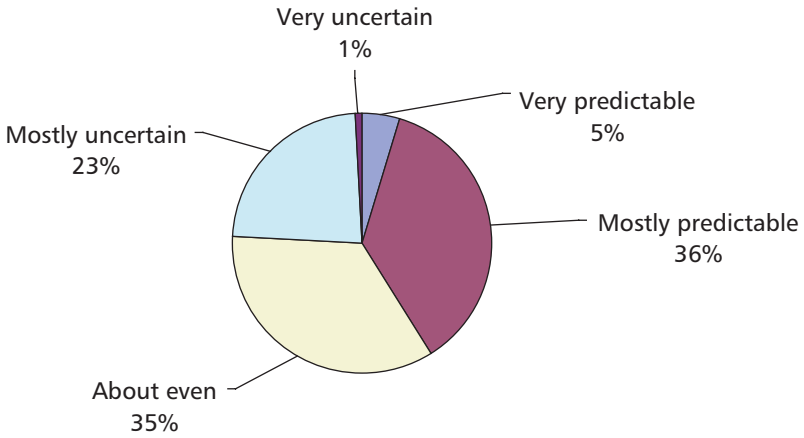
Graph 4 — Has your organization committed to a service-oriented architecture (SOA)?



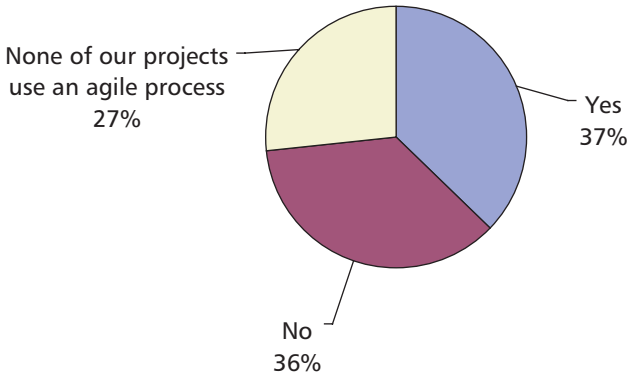
Graph 5 — Please indicate how you would characterize your organization's IT strategy along each of the following dimensions.

SURVEY DEMOGRAPHICS

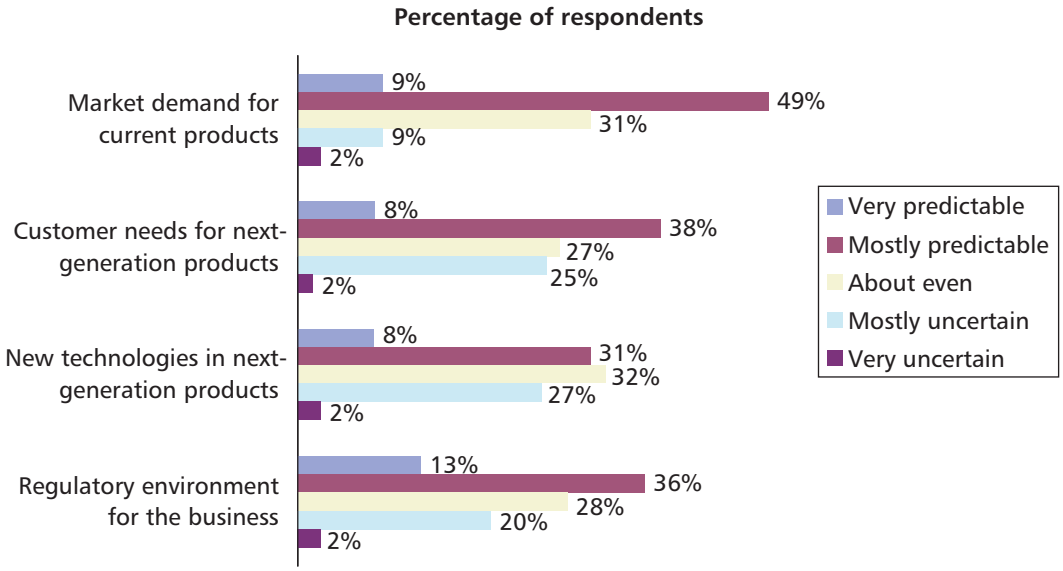
This survey investigated enterprise agility and its relationship to IT infrastructure. Nineteen percent of the 124 respondents come from companies with more than 10,000 employees, 28% from companies with between 1,000 and 10,000 employees, 33% from companies with between 100 and 1,000 employees, and the remainder from companies with less than 100 employees. Annual revenues range from more than US $10 billion (12%) to less than $1 million (11%), with 15% having annual revenues between $1 billion and $10 billion, 19% between $100 million and $1 billion, 24% between $10 million and $100 million, and 19% between $1 million and $10 million. Annual IT budgets range from less than $100,000 (9%) to more than $100 million (12%) with 23% having annual IT budgets between $100,000 and $1 million, 28% between $1 million and $10 million, and 21% between $10 million and $100 million (7% of respondents do not know the dollar amount of their annual IT budget). Thirty-three percent of the respondents identify themselves as the role of IS/IT management or senior management/policy making, 20% as project management, and 15% as consulting, with the remainder holding a range of titles.



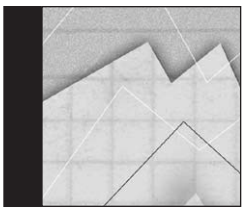
Graph 6 — How easy is it to predict the requirements for your organization's IT systems looking three years ahead?



Graph 7 — In projects that use an agile process, does your organization adopt a uniform agile methodology?



Graph 8 — Please indicate the predictability of each of the following business factors looking three years ahead.



About the Authors

Tim Lister is a Fellow of the Cutter Business Technology Council and a Senior Consultant with Cutter's Business-IT Strategies, Agile Product & Project Management, Innovation & Enterprise Agility, Enterprise Risk Management & Governance, and Sourcing & Vendor Relationships practices. He is also a frequent keynoter at Cutter *Summits*. Mr. Lister is also a principal of the Atlantic Systems Guild, Inc. Currently, he is tailoring software development processes using software risk management techniques. He has been a frequent speaker at the *Agile Development Conference*, was a Guest Lecturer on software risk management at Stanford's School of Business, and gave the Dean's Lecture at RIT. Mr. Lister was a member of the Airlie Software Council, advising the US DoD on best practices for software development and acquisition. Along with the other five principals at the Guild, he is coauthor of *Adrenaline Junkies and Template Zombies: Understanding Patterns of Project Behavior*. He is also coauthor with Cutter Fellow Tom DeMarco of the Jolt Award–winning *Waltzing With Bears: Managing Software Project Risk*, and *Peopleware: Productive Projects and Teams*. Both are also coeditors of Software State-of-the-Art: Selected Papers and have produced the video Productive Teams. Mr. Lister has more than 35 years' software development experience. Before the formation of the Guild, he worked at Yourdon Inc., where he was an Executive VP and Fellow. He holds a bachelor's degree from Brown University and is a member of the IEEE and the ACM. Mr. Lister also serves as a panelist for the American Arbitration Association and has served as an expert witness in litigation proceedings involving software problems. He can be reached at tlister@cutter.com.

Alan MacCormack is a member of Cutter's Innovation & Enterprise Agility team. Dr. MacCormack is an Associate Professor in the Technology & Operations Management Unit at the Harvard Business School. His research examines the management of technology and innovation in high-technology industries. Dr. MacCormack's work explores the dynamics of product development in these industries as well as the mechanisms by which firms can sense emerging threats to their core businesses and new opportunities for value creation. He is currently conducting a multiyear study of how firms use global collaboration networks to bring innovations to market. His work has appeared in several books and journals, including *Harvard Business Review* and *Sloan Management Review*, and he is the author of more than 50 cases and teaching notes. Dr. MacCormack has worked with leading organizations like Intel, Microsoft, and General Motors to help develop better approaches to the management of innovation as well as to understand how these firms leverage global networks in the pursuit of newly emerging technologies. He received his PhD from Harvard Business School in 1998, where he was a recipient of the George S. Dively Entrepreneurship Award for distinguished research. Dr. MacCormack has a master's degree in management from MIT's Sloan School of Management and a BSc in electrical engineering from the University of Bath in England. He can be reached at amaccormack@cutter.com.

Lou Mazzucchelli is Fellow of the Cutter Business Technology Council and Senior Consultant with Cutter's Innovation & Enterprise Agility and Social Networking practices. He is a venture partner at Ridgewood Capital, where he helps manage their technology portfolio. He is also a consultant to LightSpace Technologies and a director of Forgent Networks. Prior to joining Ridgewood Capital, Mr. Mazzucchelli was an investment banker at Gerard Klauer Mattison. He was named to the Wall Street Journal all-star team in 1998 and in 1999 was one of nine "Home-Run Hitters" analysts recognized for his stock-picking performance. Previously, he spent 13 years leading Cadre Technologies, a pioneering CASE tools company that he founded in 1982 and that grew to become one of the top 50 US ISVs before its sale in 1986. During this period, he was listed in the "Top 200 in the Software Industry" by *Software Magazine*. Mr. Mazzucchelli began his career in data communications, moving to IT management and consulting before founding Cadre. He can be reached at lmazzucchelli@cutter.com.

Gabriele Piccoli is a Senior Consultant with Cutter Consortium's Business-IT Strategies, Business Intelligence, and Innovation & Enterprise Agility practices. He is Associate Professor of Information Systems at the School of Economics and Business Administration at the University of Sassari (Italy). His consulting, research, and teaching expertise is in strategic information systems and the use of advanced IT to support customer service. Prior to moving to Italy, Dr. Piccoli was Associate Professor of Information Systems at the School of Hotel Administration and Hospitality Management at Cornell University. He has also held positions as Adjunct Professor of Information Systems at the AB Freeman School of Business at Tulane University and as instructor at the EJ Ourso College of Business at Louisiana State University, where he received his MBA as well as his PhD in business administration with emphasis in management information systems. Dr. Piccoli is the author of the book *Information Systems for Managers: Text and Cases*. His research has appeared in both academic and applied outlets such as *MIS Quarterly*, *Decision Sciences Journal*, *MIS Quarterly Executive*, *Communications of the ACM*, *Harvard Business Review*, *The DATA BASE for Advances in Information Systems*, and *Cornell Quarterly*. He can be reached at gpiccoli@cutter.com.

About Cutter Consortium

Cutter Consortium is a unique IT advisory firm, comprising a group of more than 150 internationally recognized experts who have come together to offer content, consulting, and training to our clients. These experts are committed to delivering top-level, critical, and objective advice. They have done, and are doing, groundbreaking work in organizations worldwide, helping companies deal with issues in the core areas of software development and agile project management, enterprise architecture, business technology trends and strategies, enterprise risk management, business intelligence, metrics, and sourcing.

Cutter delivers what no other IT research firm can: We give you Access to the Experts. You get practitioners' points of view, derived from hands-on experience with the same critical issues you are facing, not the perspective of a desk-bound analyst who can only make predictions and observations on what's happening in the marketplace. With Cutter Consortium, you get the best practices and lessons learned from the world's leading experts, experts who are implementing these techniques at companies like yours right now.

Cutter's clients are able to tap into its expertise in a variety of formats including print and online advisory services and journals, mentoring, workshops, training, and consulting. And by customizing our information products and training/consulting services, you get the solutions you need, while staying within your budget.

Cutter Consortium's philosophy is that there is no single right solution for all enterprises, or all departments within one enterprise, or even all projects within a department. Cutter believes that the complexity of the business technology issues confronting corporations today demands multiple detailed perspectives from which a company can view its opportunities and risks in order to make the right strategic and tactical decisions. The simplistic pronouncements other analyst firms make do not take into account the unique situation of each organization. This is another reason to present the several sides to each issue: to enable clients to determine the course of action that best fits their unique situation.

For more information, contact Cutter Consortium at +1 781 648 8700 or sales@cutter.com.

The Cutter Business Technology Council

The Cutter Business Technology Council was established by Cutter Consortium to help spot emerging trends in IT, digital technology, and the marketplace. Its members are IT specialists whose ideas have become important building blocks of today's wide-band, digitally connected, global economy. This brain trust includes:

- Rob Austin
- Ron Blitstein
- Christine Davis
- Tom DeMarco
- Lynne Ellyn
- Tim Lister
- Lou Mazzucchelli
- Ken Orr
- Ed Yourdon