

## Toward a User-Driven Approach to Radiology Software Solutions: Putting the Wag Back in the Dog

Matthew Morgan, Jonathan Mates, and Paul Chang

The relationship between healthcare providers and the software industry is evolving. In many cases, industry's traditional, market-driven model is failing to meet the increasingly sophisticated and appropriately individualized needs of providers. Advances in both technology infrastructure and development methodologies have set the stage for the transition from a vendor-driven to a more user-driven process of solution engineering. To make this transition, providers must take an active role in the development process and vendors must provide flexible frameworks on which to build. Only then can the provider/vendor relationship mature from a purchaser/supplier to a codesigner/partner model, where true insight and innovation can occur.

**KEY WORDS:** User-driven, software design, software industry, value innovation, agile software methodologies

A recent editorial in the *Journal of Digital Imaging* addresses the challenges of transferring innovative Radiology Informatics research into industry solutions.<sup>1</sup> The author's metaphorical reference to the industry "tail" wagging the healthcare "dog" appears justified. While industry is focused on market share, profit margins, and competition-driven engineering,<sup>2</sup> quality healthcare providers and their knowledge workers are interested in customized solutions, optimized workflows, and "data-driven engineering."<sup>1</sup> The result of these misaligned priorities is a gap between the generalized solutions industry is willing to provide and the optimized solutions healthcare providers<sup>3</sup> want—a value innovation gap. Those experiencing this gap have a nagging frustration that the end-users are not driving the process. Yet, the question remains, how do we bridge this gap and produce true technical innovation and value distinction, and in so doing, *put*

*the wag back in the dog?* From among the various alternatives, we believe a promising approach is to shift our paradigm of software solution engineering from a traditional, passive, vendor-driven model to an innovative, active, user-driven model that more effectively captures the needs of the users and more naturally aligns incentives. To explain how this shift is possible, we will discuss (1) the limitations of the industry-driven model, (2) how current technology and methodologies have made it possible for institutions to put value innovation back in their own hands, and (3) an alternative provider/vendor partnership for maximizing value innovation. Although our focus is mainly within radiology, we feel that these concepts may be applied to healthcare software engineering as a whole.

### THE INDUSTRY-DRIVEN MODEL (AND WHY IT ULTIMATELY FAILS THE END-USER)

Industry-driven software engineering models in healthcare have significant limitations for two important reasons: (1) traditional market-driven engineering methodologies are flawed in disrup-

---

*From the Radiology Informatics, University of Pittsburgh Medical Center, N 421 MUH, Pittsburgh, PA 15213, USA.*

*Correspondence to: Matthew B. Morgan, Department of Radiology Informatics, University of Pittsburgh Medical Center, 200 Lothrop Street, Pittsburgh, PA 15213; tel: +1-412-6926641; fax: +1-412-6924477; e-mail: morganmb@upmc.edu*

*Copyright © 2006 by SCAR (Society for Computer Applications in Radiology)*

*Online publication 07 June 2006  
doi: 10.1007/s10278-006-0591-y*

tive environments, and (2) provider/vendor incentives are misaligned.

### Limitations of Traditional, Market-Driven Engineering Methodologies

Software design is a challenging process of understanding a user's vague notions of needs, values, and constraints, and transforming these into an accurate software representation. The difference between success and failure hinges upon understanding and extracting the *essential metaphor*—the shared context that bridges the gap between the user and the developer.

In traditional, market-driven engineering methodologies, industry uses standard tools such as focus groups, time-motion studies, and surveys to capture the user's requirements. Then, after a long development cycle, the product is finally ready to use. This deliberate approach is successful when user requirements are well understood and relatively stable over time. Many of today's Informatics challenges, however, violate these two conditions. Today's healthcare providers are composed of diverse, evolving organizations whose expectations and requirements are complex and volatile. In radiology specifically, the recent transition from film to digital image management has caused significant changes in workflow and practice patterns. Economic pressures and technical advancements are triggering trends toward consolidation of radiology groups and greater subspecialization.<sup>4</sup> In this disruptive setting, where both the customers and their requirements are moving targets, traditional market-driven methodologies lack both the agility and the proximity to respond to the user's constantly evolving needs, and often the result is a product that "doesn't fit."<sup>5</sup>

For example, focus groups can miss the mark for several reasons. First, vendors often *focus* on the *wrong group*. Instead of targeting expert clinicians who experience the real day-to-day pressures of practice, they tend to seek domain experts who are largely biased toward academics and enamored with technology. The result can be a *self-resonating* panel whose opinions and recommendations may not reflect those of the mainstream, thus guiding the vendors toward solutions that neglect a large contingent of users. Also, focus groups tend to favor consensus over individualization. Yet, healthcare providers are

appropriately idiosyncratic, requiring customized—not generalized—solutions. Finally, focus groups assess what customers *say* they do, not what they *actually* do. Because there is often a great difference between the two, focus groups can be misleading.<sup>6</sup>

Another standard marketing tool for understanding the customer's needs, time-motion analysis, also has its pitfalls. A form of direct observation, time-motion analysis records a knowledge worker's actions to look for opportunities for process improvement. Because a provider's existing workflows and business models have developed over time to exactly fit their needs, effective time-motion studies require close familiarity with the organization. Vendors, however, are typically outsiders to the organization and therefore lack the necessary proximity to the workflows to achieve this awareness. This makes their observations inherently passive rather than hypothesis-driven, and makes genuine insight less likely.

Surveys, notwithstanding their popularity, may be the least effective means for understanding the users' requirements. Low response rates and selection bias (from "techie" or those with "an axe to grind") greatly limit the validity and generalizability of the results. With these significant limitations, a survey is unlikely to obtain an accurate picture of the general user's needs.

### Provider/Vendor Incentive Misalignment

Another limitation of the industry-driven model is provider/vendor incentive misalignment. A good example of this is the *value innovation gap*—the misalignment between the customized solutions a healthcare provider wants and the generalized products that a vendor is willing to provide. Where healthcare providers want optimized solutions for their unique workflow and business models, vendors are looking to produce universal solutions for a mass market.

The reasons for these differing perspectives are straightforward, but the consequences are problematic. Healthcare providers are appropriately idiosyncratic and have developed best practices and business models that are uniquely suited to their particular needs and situation. Better than anyone else, they understand their unique workflows and challenges. Hence, they are not inter-

ested in “one-size-fits-all” products, but in solutions that leverage their unique strengths and capabilities to promote competitive advantage and local distinction. Yet, requests for solutions that would give them this leverage appear to go unheeded by industry.<sup>1</sup> The problem is that the current industry-driven model gives vendors very little incentive to respond.

As a competitor in a free-market economy, the vendor’s goal is to maximize sales and market share. Consequently, it takes a “lowest common denominator” approach to software engineering, focusing its efforts on developing generalized solutions to satisfy the needs of the average customer. This economic strategy allows industry to maximize its sales and minimize its costs. Because multiple customized versions of a product are costly to support and maintain, once the software is purchased and has thereby met the vendor’s needs, any vendor promises of extensive customization after purchase are rarely fulfilled.

The result of these competing strategic and economic goals is often discontentment on the part of the healthcare provider. When vendors design a product for the masses, healthcare providers are left with a generalized, “as is” product. So, instead of the software adjusting to the needs of the customer, the customer is forced to change its practices to adapt to the software.

#### TOWARD A USER-DRIVEN MODEL

Fortunately, advances in both hardware and software engineering methodologies and capabilities have created an alternative to the industry-driven model in which the knowledge user can and must take an active role. Early on, computers were large, very expensive, and tedious to program. Computer pioneers recall (with some degree of pride) using punch cards to write machine code for computers that “filled rooms.” A lack of standards meant that systems were “monolithic,” meaning that a single, large program had to account for all hardware and software processes. Only large institutions and industry had the expertise and resources to participate in long, expensive development cycles. The challenges were technical and knowledge workers had low expectations, simple requirements, and little overall contribution to the process (Fig. 1).

Over the years, computer technology, development methodologies, and system architectures have undergone exponential progression. Ever-increasing processor speeds and storage capacities have paved the way for increasingly reliable, cheap, and usable computers. Hardware improvements are paralleled by advances in programming languages, operating systems, and development platforms. In place of the monolithic model where systems were built piece by piece from the ground up using low-level programming languages, modern programmers can now use integrated development environments that leverage modular, reusable “objects” and “code libraries.” Proprietary system architectures are being replaced by platform-independent technologies and services (i.e., HTML, XML, and web services). Service-oriented architectures are self-contained and do not depend on the context or state of other services. The result is an interconnected global network and the potential for almost limitless information access and exchange using the World Wide Web.

The bottom line is, today’s major challenges are no longer technical and computational. Hardware is a cheap commodity, programming software is relatively easy, and modern system architectures

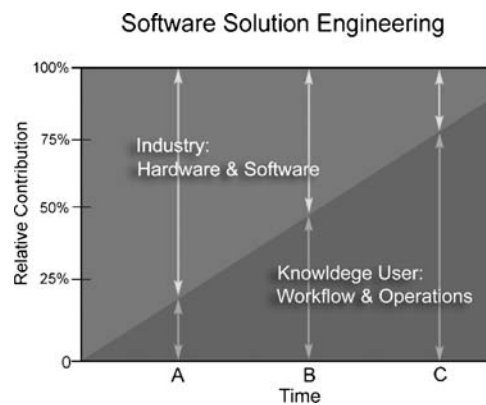


Fig 1. A comparison of the relative contribution of industry vs. knowledge users to software solution engineering over time. In the early days of computers (A), the challenges were mostly technical; users had low expectations and few requirements, and the relative contribution of industry was significant compared the knowledge user. Today (B), hardware is more capable and software development is relatively easy, but users now have sophisticated requirements and complex workflows. Unlike before, solving these challenges demands extensive user involvement. In the future (C), as software development tools continue to become increasingly reusable and modular, knowledge users will likely play the major role in the development process with the aid of powerful, enabling development platforms provided by industry.

facilitate open communication and data sharing. Today, the major challenges are as follows: understanding and extracting the essential workflow model and transferring it into optimized solutions; understanding how to leverage the technology to meet sophisticated requirements and high expectations; and responding with agility and precision in a disruptive and evolving environment. Before, the big questions were, “How do we move bits and bytes around?” and “How do we display information to the user?” Now, the big questions are, “How do I optimize my workflow?” and “How do I leverage this technology to be a better value?” The answers to these questions are going to come more and more from local knowledge workers, and less and less from industry (Fig. 1). This is a shifting paradigm, and failure to recognize it is the likely source of much of the current discontent between providers and vendors.

## USER-DRIVEN, AGILE METHODOLOGIES

### New Provider/Vendor Partnership

Today’s disruptive healthcare climate demands a new approach to software solution development that both recognizes and leverages the relative strengths of vendors and healthcare providers, and naturally realigns incentives. Users understand their appropriately idiosyncratic workflow model; industry understands how to produce solid, stable, sustainable software. Because industry is not well suited or incentivized to understand and provide optimization and customization, how does a healthcare provider go about taking an active role in driving the development process? The answer includes a combination of administrative commitment, an organizational framework, and a development methodology that provides for local, agile, user-driven software development that picks up where industry leaves off.

### The Role of the Healthcare Provider

To take a more active role in solution engineering, healthcare providers must take a new look at the Informatics discipline and its role in their organizations. Far too often, Informatics is confused with Information Technology (IT), Informatics education is confused with computer literacy, and Informatics specialists are confused with power users. If solution engineering is likened to building construction, IT is the contractor; Informatics is the architect. Informatics is not just about configuring hardware, software, and networks, but about applying information science to solve the informational needs of the organization. In essence, Informatics facilitates the efficient application of the scientific method. In medicine, it represents the cycle of gathering and interpreting patient data

to produce information and hypotheses that enable appropriate, timely interventions. The goal of informatics is to fine-tune this cycle so that knowledge workers have access to high-quality, usable information at the point of decision making.

As healthcare administrators catch this vision, they will see the value of incorporating local software development into their organizational framework. Healthcare providers must shed the self-perception that they are passive recipients of industry software. The expertise and innovation needed to solve the most challenging problems is found within the institution. Remember, the problem is not building software; it’s optimizing workflow. Fortunately, with today’s advanced technology and plentiful, talented programmers, the power to drive software development is finally within reach of the healthcare provider (i.e., the end-user). For most organizations, “build vs. buy” is a false dilemma. The most robust and flexible solutions will be a combination of both vendor and in-house products.

To take advantage of this potential synergy, the healthcare provider needs not only skilled developers, but also the leadership and vision to guide them. Just as construction projects are prone to failures if contractors are used without architects, information systems are at risk without the leadership and strategy provided by trained informaticists. Local clinicians who are skilled and/or formally trained in Informatics are best positioned see the bigger picture and to “architect” this process. The IT department and the vendor, although very skilled and capable, do not have a local, clinical perspective. Notwithstanding, many healthcare organizations currently have no organizational structure to support this leadership and thereby defer important strategic decisions to “contractors” rather than to “architects.” It is no surprise when the metaphorical end result is a rambler when what was really wanted was a colonial.

Once the right organizational infrastructure is in place, a responsive methodology is needed to move from the problems to the solutions. Because healthcare system requirements are unpredictable, unstable, and always evolving, the methodology should emphasize user input, rapid reaction, and high maneuverability. Relatively new software development methodologies have emerged in the last decade that help meet these disruptive demands.<sup>7</sup> Although they are no panacea, these agile methodologies focus on collaborating with users and responding to change. These characteristics make them well suited for volatile environments such as today’s healthcare.

A user-driven, agile approach to software development has various implications. First, success is more likely because the leaders driving the process work side by side with the other knowledge workers. This insider position gives them much more credibility and buy-in from the users. Clinicians, who serve dual roles as knowledge workers and as systems architects, are able to engage the local users, observing them “in the wild” to facilitate solutions that meet the needs of a real work environment. By building organically, with rapid, iterative cycles, the development team can make small, midcourse corrections toward the constantly changing system requirements.

### The Role of Industry

In this alternative, user-driven model of software engineering, what is the role of industry? Vendors must recognize that

the current market-driven, one-size-fits-all approach to software development will rarely satisfy the appropriately idiosyncratic and increasingly sophisticated demands of the individual healthcare provider. Vendors have too much distance (geographic and intellectual) from the user, too much latency in their development cycles, and in the end, no real incentive to optimize the product for any individual customer. Perpetuating the current model will only result in continued provider discontent.

Knowing this, vendors have the opportunity to explore innovative technical and business models that embrace the disruptive atmosphere and use it as competitive advantage. Instead of “as is,” one-size-fits-all, turnkey software *products*, vendors can provide flexible, scalable, and enabling *platforms* that allow local innovators to customize and optimize “at the edge.” In other words, the vendor can provide powerful, high-performance engine and allow healthcare providers to decide whether to make it into an SUV or a sports car. This requires vendors to move from closed, proprietary, guarded software architectures that make information exchange purposely difficult, to standardized, service-oriented architectures that decouple the data from the presentation state and support fluid transfer of information between systems (web services, application programming interfaces, software development kits, etc.).

These changes reflect the natural evolution of industry/provider business models. Innovative vendors will recognize that IT is now entering a more responsible, business-connected portion of the relationship life cycle characterized by greater mutual understanding and shared responsibilities. They will consider moving to business models that involve more shared risk and reward ongoing commitment and responsiveness. Shared risk naturally aligns economic incentives for both the vendor and customer, keeping both parties engaged and motivated. The adoption of these innovative software architectures and business models will facilitate information exchange not only on a local level, but will lay the groundwork for regional and national medical information exchange. And with the growing national impetus to create health information networks, the vendors who adopt these models early will also have a significant competitive advantage.

## CONCLUSION

Current discontent with the industry-driven model of software engineering is justified. In today’s disruptive healthcare environment, traditional marketing-based models of product development are ill suited to meet the needs of the

appropriately idiosyncratic healthcare provider. Industry has neither the local expertise nor the economic incentive to create solutions that bridge the value innovation gap. As IT becomes more of a commodity, the biggest breakthroughs will come not from the technology itself, but from the optimization of work and information flow. Innovation in these critical areas demands that knowledge users take a more active role in driving the software development process. This requires that providers allocate resources and modify organizational structures to accommodate local, agile teams of Informatics architects and developers who can leverage their familiarity and proximity to the institution to respond to its continually changing needs. Industry, in turn, can recognize that this maturing model is not a threat, but an opportunity to move from a purchaser/supplier relationship to one of codesigner/partner. Then, instead of being bound to an “as is” product, healthcare providers can be free to leverage their own expertise and drive true innovation and local distinction. So, how can we “put the wag back in the dog?” Adopt a user-driven software engineering approach so the end-user can finally start doing the wagging.

## REFERENCES

1. Reiner B, Siegel E, Siddiqui K: The tail shouldn’t wag the dog. *J Digit Imaging* 17:147–148, 2004
2. Channin DS: Driving market-driven engineering. *Radiology* 229:311–313, 2003
3. Porter ME: *Competitive Advantage: Creating and Sustaining Superior Advantage*. New York, NY: The Free Press
4. Nauert RC: Radiology system evolution in the new millennium. *Radiol Manage* 23:30–33, 36–40, 2001
5. Christensen CM, Bohmer R, Kenagy J: Will disruptive innovations cure health care? *Harv Bus Rev* 78:102–112, 199, 2000
6. Nielsen J (1997) The use and misuse of focus groups. Date accessed: October 13, 2005 <http://www.useit.com/papers/focusgroups.html>
7. Beck K: *Extreme Programming Explained: Embrace Change*. Boston, MA: 1st Addison-Wesley Professional, 1999