Cutting-Edge Agile II

Mental Leaps:
More, Faster, Better, Happier, and Innovative!
by Erik Schön   p. 6

LabScrum: A Case Study for Agility in Academic Research Labs
by Lisa May and Tamara Runyon   p. 15

Agile HR: The New Way to Design Employee Experience
by Zuzana Sochova   p. 21

Do We Need More Transparency?
by Emilio Gutter   p. 26

Beating Production Expectations by Embracing Agility
by Simon Orrell   p. 30

Alistair Cockburn
Guest Editor

CUTTER Business Technology Journal

As business models for creating value continue to shift, new business strategies are constantly emerging and digital innovation has become an ongoing imperative. *Cutter Business Technology Journal* delivers a comprehensive treatment of these strategies to help your organization address and capitalize on the opportunities of this digital age.

Cutter Business Technology Journal is unlike academic journals. Each monthly issue, led by an expert Guest Editor, includes five to seven substantial articles, case studies, research findings, and/or experience-based opinion pieces that provide innovative ideas and solutions to the challenges business technology professionals face right now — and prepares them for those they might face tomorrow. *Cutter Business Technology Journal* doesn't water down or delay its content with lengthy peer reviews. Written by internationally known thought leaders, academics, and practitioners — you can be certain you're getting the uncensored perspectives of global experts.

You'll benefit from strategic insight on how the latest movements in digital innovation and transformation, artificial intelligence/machine learning, Internet of Things, blockchain, analytics, and cloud, to name a few, are changing the business landscape for both new and established organizations and how cutting-edge approaches in technology leadership, enterprise agility, software engineering, and business architecture can help your organization optimize its performance and transition to these new business models.

As a subscriber, you'll also receive the *Cutter Business Technology Advisor* — a weekly bulletin featuring industry updates delivered straight to your inbox. Armed with expert insight, data, and advice, you'll be able to leverage the latest business management thinking to achieve your organization's goals.

No other journal brings together so many thought leaders or lets them speak so bluntly — bringing you frank, honest accounts of what works, what doesn't, and why. Subscribers have even referred to *Cutter Business Technology Journal* as a consultancy in print and likened each month's issue to the impassioned discussions they participate in at the end of a day at a conference!

Get the best in thought leadership and keep pace with the technologies and business models that will give you a competitive edge — subscribe to *Cutter Business Technology Journal* today!

Opening Statement

by Alistair Cockburn

Agile is spreading and changing at such a rate that we are devoting a second issue of *Cutter Business Technology Journal* (*CBTJ*) to the topic. In the first issue,[1] we examined the idea that we have entered the "post-Agile" age — an in-between period, where the original ideas from the Agile Manifesto[2] have largely been incorporated into our culture, but what comes next has not yet formed clearly enough to be named.

Agile will never be fully adopted everywhere. Geoffrey Moore's discussion of "crossing the chasm"[3] examines why. He describes five attitudes toward any new idea. A small group, the innovators, will try anything new. A slightly larger group of early adopters will see the value and bring the new idea into their lives. Then there is a gap before a large mass of people, the early majority, accept the idea and start to adopt it. The late majority follow, slowly accepting the idea, but perhaps unwillingly. Finally come those who really don't like to change, who resist the new idea until the very end, and who may never adopt the idea at all.

Traditional Agile has already passed into the late majority, following this rough timeline:

- Pre-Agile was adopted by the innovators already in the late 1990s.

- The early adopters picked it up in 2000-2005.

- By 2006, the Agile Manifesto was being used as the basis for contracts, and organizations such as the Project Management Institute, the Software Engineering Institute, and the Institute of Electrical and Electronics Engineers were looking for ways to incorporate the ideas into their platforms.

- In 2009, I gave a talk in which I declared that Agile had passed an inflection point, and people were no longer asking, "What is Agile?" Instead, they were asking, "How do we use Agile in large, distributed, and even life-critical projects?" The adoption of Agile ideas had moved deep into the zone of the early majority.

- By 2012, the innovators and early adopters, having long experimented with ordinary Agile ideas, were moving to Eric Reis's "Lean startup" concept.[4] By 2015, they graduated to what they called "hypothesis-driven development," and by 2017, the new Agile product management profession was in motion among the early adopters.

- From 2016 to 2019, while the early adopters had moved on with these ideas, "ordinary" Agile was making inroads into large, traditional companies — the late majority. Resistant to change and looking to claim the term "Agile" without actually changing anything, the dominant sales term in the industry was *scaling*. "How do we scale Agile?" — usually without actually taking on the difficult work of improving communication, speeding delivery, or softening the command-and-control culture. My view was and still is that this is not a failure of the Agile concept or even the Agile culture, but an ordinary response of the late majority.

- Finally, the laggards still resist and will resist to the end. This, again, is not a failure of the Agile culture or concept, but an ordinary upper limit to the adoption of any idea. The Agile way of working requires a certain tolerance of uncertainty and ambiguity that is uncomfortable for many people, limiting who will accept working this way.

Thus, the next wave of Agile evolution is growing past the early adopters into the early majority, while the older, baseline Agile ideas are still working their way into large, traditional organizations. The current wave of Agile and post-Agile ideas involves taking the Agile concepts out of the IT department into HR, purchasing, sales, printing, restaurants, and even energy resource production (as we see in this issue). This wave of ideas involves directly tackling the question of culture: how do we run our organizations with radical transparency into everything from financial statements to personal salaries?

As we saw in the earlier *CBTJ* Agile issue, there are three big movements these days: (1) faster feedback, (2) the use of Agile outside of product design, and (3) simplification of the Agile concepts.

Modern product design groups demand quick feedback. They know that each decision is the foundation for many decisions in the future and that they cannot correctly guess users' reactions. Each decision establishes a direction for the future, so they want to check on these decisions immediately, before putting in additional work in the wrong direction. They are looking for "probes" into the workings of the users — just enough to get feedback. They cannot wait two weeks for a Scrum delivery to gather information on these directional decisions. Changing an organization to deliver daily probes into the world is not easy; it may be as difficult as it was to change the organization a decade ago, when the aim was demos every two to four weeks. It is being done by some organizations, but not many, and therefore represents "the cutting edge" of Agile.

The second evolutionary movement of Agile has taken it outside of software, and even outside of product design. Even though we wrote the Manifesto for Agile *Software* Development, we quickly realized that the Manifesto had much broader application. The Agile concepts suit any sort of initiative involving decision making. The Manifesto really is saying that people come up with ideas and make decisions collaboratively and that they need real feedback to correct and improve those ideas and decisions. In other words, "Agile" as a mindset and program suits any mental endeavor.

As Agile moved from the early adopters into the early and late majority, it became more decorated, complicated, and rule bound. The early adopters liked the simplicity of the Agile Manifesto and the space it gave them to talk to each other informally, to follow their instincts. People in larger organizations, however, wanted more clarity into the process, more detail, more

repeatability across teams, more certainty and security in what they were doing. They requested, and got, clearer and more detailed protocols and ceremonies. This clarity came at a cost: less speed, less empowerment, less maneuverability — in short, less agility.

I believe this is not a failure of anything but simply a consequence of the forces acting within large organizations. And, while it is true and inevitable, the result is still less efficient and less agile. This is why high-performing Agile teams decry these additions and complications.

The third frontier of the new wave of Agile is the search for a way out of this dilemma — a way to make things simpler again, while still keeping all the benefits of Agile. My own work has been to boil the essence of Agile efficiency into just four words: collaborate, deliver, reflect, and improve. As you may recall from the previous *CBTJ* issue on Agile, I call these the Heart of Agile,[5] which is really just a centering device that allows busy people to remind themselves of the few things that are really important. It has been interesting to see how these four simple words are, at the same time, easier to adopt than complicated formulas, allow more space for individual tailoring to circumstances, and open up new doorways for investigation.

Among the new doorways we have found are the importance of HR departments, salary discussions, annual appraisals, and executive bonuses. These are topics not normally associated with the concept of agility, but as you will see in this issue, the implications of working on collaboration directly affect many parts of the organization.

Taking Culture Seriously

When you direct your attention to the topic of collaboration, you discover very quickly that you need to improve the organizational culture. Collaboration turns out to be a fragile thing. It is based on trust, which is even more fragile, and its enemy is fear, which abounds in the workplace and is a potent killer of collaboration. Generally, we (I include myself here, as well as executives and managers in most organizations) would like to avoid the tricky and painful topics of culture, trust, and fear. However, the best companies are working on these topics — decreasing fear, raising trust, lowering punishment, and increasing listening and inclusion. Through these efforts, they get more ideas, faster feedback, and faster forward movement on their

Upcoming Topics

Industry 4.0
Keng Siau

Is Software Eating the World?
Greg Smith

initiatives. Companies that don't work on these topics will simply get left behind.

What Does It Mean to "Work on Culture?"

All initiatives these days are based on ideas and decisions, starting with a few idea initiators and sponsors, and growing through a network of workers until final implementation. Each person makes errors in decision making at the rate of one error every five, 10, or 20 decisions. Any initiative eventually depends on hundreds of thousands of decisions, built one on top of another. That means the final implementation, if done in the traditional big-bang or waterfall style, will contain thousands to tens of thousands of errors.

With an improved, high-trust, low-fear, low-penalty culture, we detect those errors — or any form of surprising bad news — as early as possible. This gives the team the maximum time to adjust for that unexpected bad news and to lose the least amount of time pursuing an incorrect or suboptimal direction. "All bad news early" is a good mantra.

We want people to look at all ideas with friendly but critical eyes, to find ways to express their thoughts without fear, and to receive information about mistakes without damage. Creating such a culture means taking the modern concern for diversity and inclusivity seriously. It means teaching dominant people in the organization — whether senior technical people, managers, executives, or others with strong personalities — to listen more patiently to those different from them, those who are quiet, shy, or fearful. It means changing reward structures so that members of the same team or organization are not in competition for the same rewards.

In This Issue

In our first article, Erik Schön describes what had to be done in a company of 2,000 people across 10 countries to introduce agility. He summarizes the company's shift in three areas: from methods and tools to principles and mindset, from resource efficiency to flow efficiency, and from scattered experiences to continuous innovation. You will notice in his article his emphasis on first changing the mindset.

Next, Lisa May and Tamara Runyon take agility out of its normal domain of product development into the world of research, where its use is not at all obvious.

They outline the cultural blockers, note the obvious mismatches to ordinary agility, and describe how they adapted both the ceremonies of traditional agility and their own culture to form an effective final mixture.

In the following article, Zuzana Šochová describes the transition from a traditional HR world to one that fits the new culture of the Agile organization. She discusses the shifts in recruiting, appraisals and reviews, salaries, and career tracks, and the difficulties facing anyone embarking on the Agile path. She closes with a discussion of new leadership roles: leading, coaching, and facilitating.

Emilio Gutter then tells a personal story of embarking down this path. After outlining the benefits of his company's experiment with transparency, he discusses the difficulties with loss of power and control, slower decision-making processes, and what he calls "frictional costs," when ordinary workers, not specialists, are making corporate decisions. He describes his company's approach to these difficulties.

In the final article, Simon Orrell takes us into another unexpected domain: production of energy resources. It is not obvious at the outset how embracing an Agile mindset might alter energy resource production rates, so his recounting of this story is especially interesting.

I trust you enjoy your reading.

Alistair

References

[1]Cockburn, Alistair (ed.). "Cutting-Edge Agile." *Cutter Business Technology Journal*, Vol. 32, No. 3, 2019.

[2]"Manifesto for Agile Software Development" (https://agilemanifesto.org).

[3]Moore, Geoffrey A. *Crossing the Chasm*. Collins Business Essentials, 2014.

[4]Ries, Eric. *The Lean Startup*. Currency, 2011.

[5]"The Heart of Agile" (https://heartofagile.com).

Alistair Cockburn is a Senior Consultant with Cutter Consortium's Business Agility & Software Engineering Excellence practice, and Consulting Fellow at Humans and Technology. Specializing in organizational psychology and Agile development, he helps clients succeed with corporate strategy and organizational culture. Dr. Cockburn was one of the original authors of the Agile Manifesto. He can be reached at acockburn@cutter.com.

Mental Leaps: More, Faster, Better, Happier, and Innovative!

by Erik Schön

Come along with me through the experiences and insights from the ups and downs of an Agile journey in a 2,000+-person product development unit in 10 locations in Sweden, Poland, and China, resulting in a quadrupling of value throughput; a doubling of speed; a tenfold increase in quality; and happier, more engaged people who are, ultimately, more innovative.

> *Our biggest challenges were product quality, an inability to deliver what our customers needed, and keeping up with the exponential data traffic growth following the introduction of the iPhone.*

Three mental leaps emerged from our journey:

1. **From methods and tools to principles and mindset.** Tools and methods can work in some contexts, but not others. If you have your own principles and mindset, then you can adapt or create your own methods and tools to fit your context. Once we realized this, we made a mental leap from a focus on methods and tools to a focus on principles and mindset.

2. **From resource efficiency to flow efficiency.** With a need to reduce both costs and time to market, we were looking for alternatives to a resource-efficiency focus (i.e., to keeping people and equipment fully utilized at all times). We realized that our ability to innovate around state-of-the-art algorithms for optimizing packet data flows in mobile radio networks was also applicable to our product development processes. So we made a mental leap from resource efficiency to flow efficiency (i.e., to a focus on keeping work items moving through the process without waiting times, thereby delivering value as quickly as possible).

3. **From scattered experiences to continuous innovation.** We were solving problems as they occurred using taskforces in fire-fighting mode, lacking corporate memory and a common direction. By creating a shared direction, a common purpose around the need to improve, and learning how to scale our innovation efforts, we made the leap from scattered experiences to a culture of continuous innovation.

Departure

Context, Heritage, Why

The product development unit, headquartered in Stockholm, Sweden, comprised over 2,000 people, including partners in 100+ teams in 10 locations in Sweden, Poland, and China. The product consists of tens of millions of lines of real-time, embedded software (with up to 15 years' legacy code), helping more than 1 billion people in more than 150 countries communicate using 2G and 3G mobile data, video, and speech calls.

Our heritage was classic waterfall development: hundreds of people time-sliced into several two- to three-year overlapping projects. We had two product releases per year — with a system engineering phase based on requirements from a product management organization, a design and development phase, and a six- to nine-month manual integration and testing phase. During strategy sessions, we gradually realized that the potential for significant improvements to this way of working was highly questionable and, consequently, we had to try something fundamentally different.

Our biggest challenges were product quality, an inability to deliver what our customers needed, and keeping up with the exponential data traffic growth following the introduction of the iPhone.

Initial Inspiration

We had heard about a product development unit in our company located just outside Helsinki, Finland, that saw very promising initial results in terms of lead times and quality from working in an Agile manner.[1] We sent several delegations to Finland to learn more. They all

found inspiration from what they saw: few, if any, slide decks; real teams talking about delivering value to customers several times per month; confident and relaxed managers who clearly believed in what they were doing; and information radiators giving teams fast feedback on quality based on continuous integration of automated test cases.

We asked for the "secret recipe," expecting a document or a slide deck of hundreds of pages; instead, we received just one page with the advice to focus on three things initially: continuous integration, continuous integration, and continuous integration — most probably because almost all of our six- to nine-month release testing at the time was manual.

We were also advised to read the book *Scaling Lean & Agile Development*[2] for principled, yet pragmatic, guidance on what to try and what to avoid, and several leadership teams started reading circles. In addition, we were offered internal coaching and mentoring on Agile and Lean at scale that we happily accepted.

What Worked Well Early On

The development organization and the product management organization were very much aligned on needs and direction thanks to strategy work done together over several years. Additionally, the heads of these organizations were curious and eager to learn and improve themselves as well as the organization, illustrating author Frederic Laloux's wise words, "An organization cannot evolve beyond its leadership's stage of development."[3]

We avoided using classic command-and-control, top-down, big-bang change management after a long and heated debate on which approach to use. Since we wanted leaders to be role models and to start acting their way into new thinking, we set up a core team of willing and able formal and informal leaders ("influencers") from all parts of the organization. This core team was meeting openly and transparently one full day per sprint (i.e., once every three weeks) and working according to the prioritized backlog of topics to be resolved, guided by a quantified five-year vision (also known as our long-term key performance indicators [KPIs]) that came from the strategy work done earlier. In addition, we were fortunate to have experienced people from our role model organization in Finland as coaches and mentors from the very beginning of our journey. Finally, our scrum masters, Agile/Lean coaches, and organization coaches self-organized

into a coaching community that collaborated with similar coaching communities in other parts of the company — sharing and learning from each other. Early on, key decisions made by the core team included:

- **A pull-based approach for product discovery** using the Kanban method to visualize early phase studies, as well as one product owner and one product backlog with strict priorities

- **Requirement areas** (collections of customer needs from an outside-in perspective) with 20 or more teams each, enabling:

  o Independent prioritization in backlogs per requirement area

  o Transparent development capability with a set number of teams per requirement area, changeable on a quarterly basis

  o Easier domain competence building for teams

- **Continuous programs** instead of traditional projects for each new release, enabling continuous feedback, learning, and improvement to ways of working in the programs

- **Colocated, semipermanent, end-to-end, cross-functional feature teams,** which, by avoiding handovers and waiting, would improve quality and lead time; teams could choose to use Scrum or Kanban at the team level

- **Gradual ramp-up of cross-functional teams** to enable feedback, learning, and adjustments

- **Heavy investments in continuous integration** with automated, continuous, and fast feedback to the teams, which would improve delivery speed, team learning, and product quality

- **Thorough, hands-on training in Agile/Lean principles and practices** from the beginning, with a two-day Certified ScrumMaster course for all in the organization

- **A more defined coaching stance for managers,** including asking questions rather than providing answers, achieving results through transparency and trust rather than control and micromanagement, and daring to be patient and persevere, rather than going for quick fixes and silver-bullet thinking

A few years into the journey, we involved managers in redefining the expectations of a manager in an Agile

context and, in the next big organizational change, used this redefinition when recruiting managers.

The decision to use requirement areas and feature teams was preceded by a long and heated debate in which several key people argued for using the existing product architecture and corresponding component teams. Our insight was that using component teams would lead to very many dependencies when developing new features, impacting many components, and that managing these dependencies would slow down development significantly.

Other important, external prerequisites that were in place before the journey started included:

- **Expectations, support, and concrete targets** for change and improvements from the head of the business unit who was also a member of the senior executive team.

- **A global leadership model** and corresponding training of all leaders since the late 1990s based on situational leadership (i.e., that managers are expected to adjust their behavior to each individual's needs in the context of the situation at hand).[4] This model was complemented in the mid-2000s by hands-on coaching training based on the GROW model and later on additional coaching training based on David Rock's book *Quiet Leadership*.[5]

- **A global product development doctrine** evolving since the mid-2000s and a corresponding training program for senior technical leaders based on self-assessments, followed by extensive sharing and learning in teamwork exercises focusing on the current needs of participants and their organizations. The doctrine was formulated as 12 principles for large-scale, world-class product development and was influenced by internal experiences and external inspiration, such as Lean product development and Agile software development.

Mental Leap 1: From Methods and Tools to Principles and Mindset

Initially, we started practicing the methods and using the tools — Scrum, Kanban, and continuous integration — by the book and with help from internal and external coaches. Then, thanks to:

- Our initial focus on our needs and the direction we wanted to go,

- Inspiration from thought leaders like Mary and Tom Poppendieck[6] and Don Reinertsen,[7] and,

- A company culture of always thinking for ourselves,

… we moved to trying to apply the principles of flow, visualizing our work for transparency, and having a mindset of continuous learning and experimentation.

Point to the Destination and Explain Why

We set up the following long-term (five years into the future) KPIs based on our needs to improve quality, lead time, value delivery, and employee engagement along with feedback that our original vision statement was too abstract; we needed to complement it with something more concrete. Within five years, we would:

- Quadruple value to exceed customer expectations

- Halve time to market to be more responsive to customer needs

- Improve quality tenfold to secure customer trust

- Have at least 75% of our people fully motivated and engaged

We decided on the KPIs before we knew how to achieve these rather aggressive targets. The key learnings for us were that it is good to have long-term KPIs, aiming five years ahead and not only for the next quarter, and that a balanced set of targets would help us avoid suboptimization. The decision on and usage of KPIs was by no means easy due to the misuse of targets. We found them to be useful, however, to show progress to ourselves internally in our organization, providing us with additional energy and engagement, as well as to our external stakeholders, who valued the regularity and transparency. Having a set of long-term KPIs set five years into the future also gave the message that this is not a quick fix that will be ready in a quarter or two.

One important example of how we used these long-term KPIs to influence the culture was a three-month design stop. Quality measurements showed that our product quality was getting worse, so we asked all teams to stop developing new features and instead fix outstanding internal faults and resolve customer trouble tickets to remove technical debt. Ultimately, this symbolic, yet concrete, decision proved to be a turning point — letting everyone in the organization understand that quality was truly important and with management finally comprehending that better quality

results in higher speed, rather than that quality is mere talk and traded for speed. Moreover, the design stop proved the importance of having a head of product development and a head of product management who trusted each other and *dared to take risks together* in a corporate business climate, where inability to deliver short-term results could lead to demotion.

From Large Batches to Smaller Batches

Traditionally, large batches rule since economies of scale provide a cost advantage. A key insight for us was that small batches deliver better quality and shorter lead times thanks to faster feedback loops and, hence, quicker learning and ongoing adjustments. This strategy delivers more value to customers, and the organization gains a value advantage.

We struggled quite a lot to turn theoretical understanding into practical use. We offered several "elephant carpaccio exercises"[8] to get the point across in a concrete way, and we also included this exercise in the product development training mentioned above. As time passed, we found our own concrete example of the value of slicing, as described below and shown in Figure 1:

- We originally planned a feature for two teams.

- We split the feature into two parts after a dialogue involving the cross-functional development team, the product owner, the customer unit, and the customers.

- After this, we split the sub-features into subparts.

- One team could do subparts 1 and 2 in half the time compared to the original feature lead time, and it turned out that customers did not need subparts 3 and 4; hence, they were not developed. Moreover, it seemed that subparts 5 and 6 were crucial to one customer, whereas subparts 7 and 8 were not important to any customers, so they were not developed. In fact, subparts 5 and 6 were so vital that the product owner decided to reprioritize a team from another feature to this feature, hence adding one more team so that subparts 5 and 6 could be developed twice as fast as would otherwise be possible.

> *We struggled quite a lot to turn theoretical understanding into practical use.*

In summary, smaller slices meant that we could deliver customer value faster (and get paid faster!), and the early splitting into slices meant that we could get customer feedback and change direction quickly. Plus, we now had a success story that we could share and get people excited about!

Another example of the beauty of smaller batches is to avoid big-bang change initiatives from the top and to overcome resistance to change by doing small trials that can easily roll back if they don't offer any improvement (i.e., go for experiments that are safe to try). One example is how we gradually ramped up the cross-functional teams, starting with one team.
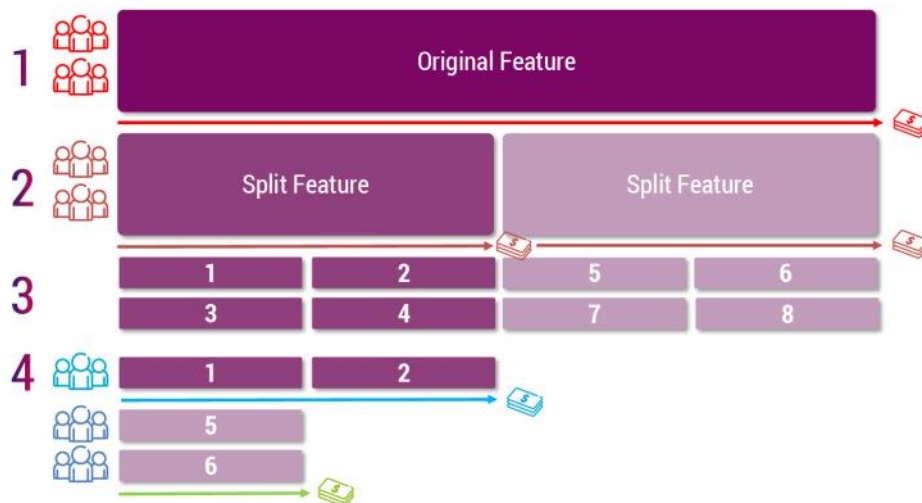

Figure 1 — Slicing a feature to secure faster value delivery and quicker feedback and learning.

From Local Suboptimization to Global Awareness

Something we found extremely valuable to secure global awareness and understanding was a visualization room with a board showing all features and stages of development — from customer need to solution delivered and in operation (see Figure 2). Stakeholders would meet at least once a week in the room at the visualization board and via videoconference to get features out of a blocked state and to improve the flow by visualizing, managing queues, and removing impediments in the product development process. However, we struggled with the videoconferencing system and keeping the boards synchronized across multiple locations; indeed, we sometimes got totally lost in feature-tracking tools and troubleshooting the videoconferencing system.

The visualization board pictured in Figure 3 shows features as colored sticky notes divided into six horizontal requirement area rows. The vertical columns show different development stages. A pink sticky note on a feature, for example, indicates a block in the progress of the feature. By looking at the board, you see the status of 100+ features at a glance. Here, it's clearly visible that certain development stages (columns) are full of features waiting, indicating a

bottleneck in this step or in the adjacent step. When this happens, it's time to act (e.g., by limiting the number of features ongoing in parallel) — that is, limiting the work in process (WIP).

Most of all, we learned that Agile and Lean are mindsets. Agile is described by four values, defined by 12 principles manifested through an unlimited number of practices. We learned the game — the mindset, the values, the principles, and the practices. Then we played the game using the practices. Finally, based on experiences and insights grounded in a thorough understanding of the mindset and principles, we were able to (re)define the game with our own practices and tools to suit our needs.

Mental Leap 2: From Resource Efficiency to Flow Efficiency

What Is Efficiency?

Toyota expert Niklas Modig, trying to answer the question "What is Lean?" realized that the question is really, "What is efficiency?" and discovered that there are two distinct answers: resource efficiency and flow efficiency.[9]

Figure 2 — Visualization room with key stakeholders and videoconference link to other sites.
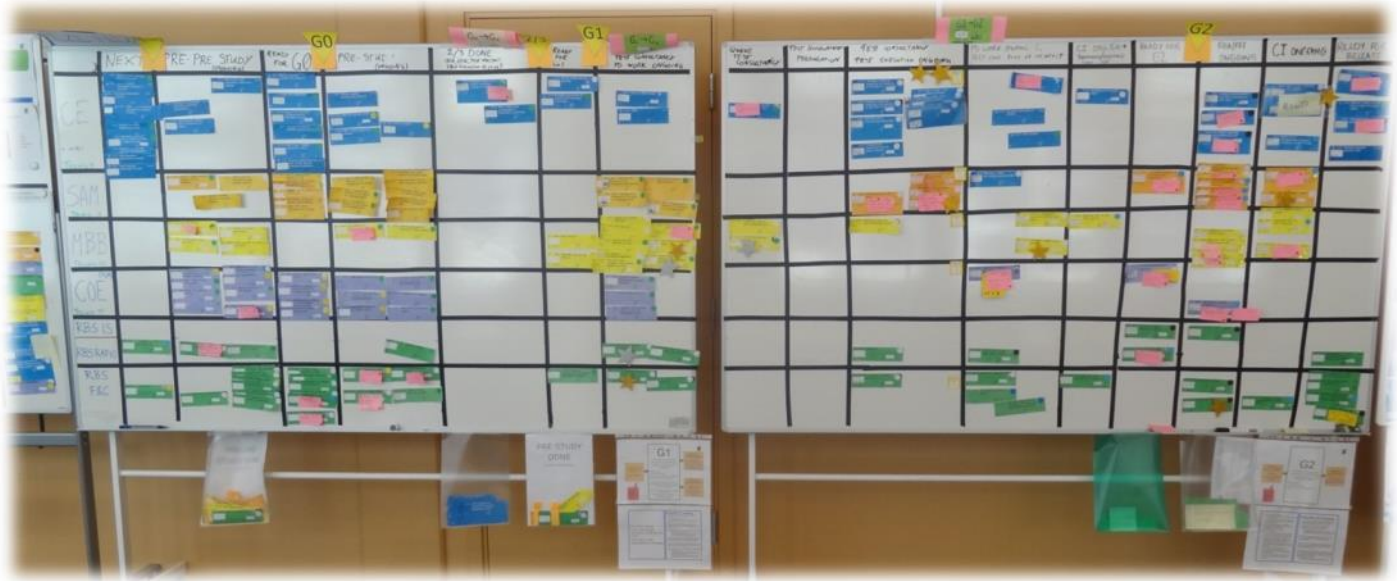
Figure 3 — A visualization board securing global awareness at a glance and hinting what to do.

In Figure 4, we see resource efficiency on the vertical axis and flow efficiency on the horizontal axis. By resource efficiency, we mean keeping people and equipment as fully utilized ("busy") as possible. By flow efficiency, we mean fulfilling customer needs as quickly as possible. We can see one example of high resource efficiency in the upper-left quadrant: the equipment in a steel mill, which makes sense to keep fully utilized since it's extremely expensive. In the lower-right quadrant, we see an example of high flow efficiency and low resource efficiency: an ambulance emergency service where it's crucial to fulfill the needs of the patient as soon as possible since it's often a matter of life or death. In this case, it's not as important to keep the resources busy, and we are willing to trade low resource efficiency for high flow efficiency. In product development, it is equally important to fulfill the needs of your customers faster than the competition does, or you will soon not have any customers left; without customers, it does not matter that you are utilizing your equipment and people very efficiently at a low cost.

In flow thinking, it's crucial to first start working on improving flow efficiency and then improving resource efficiency. Starting with resource efficiency will make it virtually impossible to achieve both resource efficiency and flow efficiency since the higher the resource utilization, the longer the response time or lead time. However, the decision is not a binary choice, one of either resource efficiency or flow efficiency; we need both.
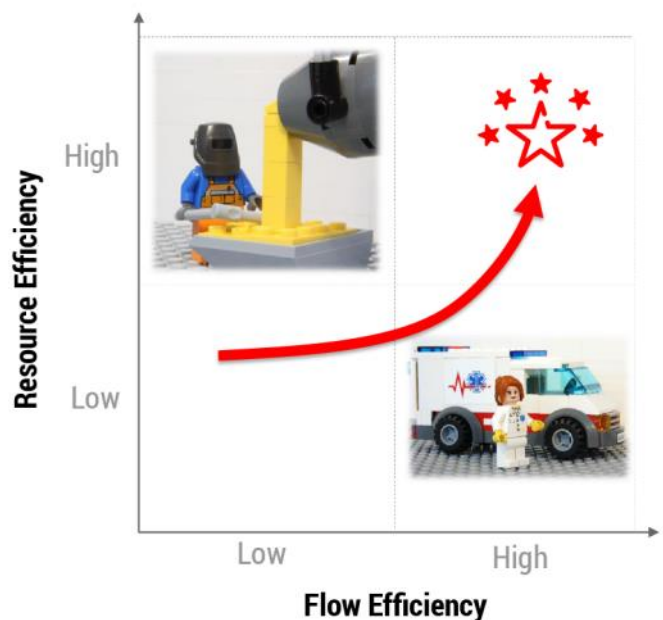
Figure 4 — Examples of high resource efficiency (steel mill) and high flow efficiency (emergency service). *(Disclaimer: LEGO® is a trademark of the LEGO Group, which does not sponsor, authorize, or endorse this article.)*

Storytelling Around Flow in Our Products

We needed a way to explain flow efficiency to our engineers and leaders so they could start to take more balanced actions and make better decisions.

So we looked at traffic jams on a highway: the road is at 100% resource utilization, but the flow efficiency is zero

since all vehicles are at a standstill — the highway has become a parking lot. Moreover, it doesn't help if we add more vehicles or new, better engines for the vehicles.

We also looked at servers. What's the maximum utilization at which you would run your servers? It's around 60%–70%; otherwise, there will be delays in the applications running on the servers, and the delays become exponentially larger the closer we get to 100% utilization. Resource efficiency impedes flow efficiency.

In addition to cars on highways and servers, for greater understanding and inspiration, we also looked at the Toyota Production System, the Scania Truck Production System, and the Swedish Health Care System, all of which are pedagogical examples of prioritizing flow efficiency over resource efficiency. The engineers understood but were still reluctant to use these insights, saying things like, "Our context is special; we're doing creative knowledge work for global, mobile networks, which is very different from manufacturing, healthcare, highways, or servers."

Then, a few of us took Reinertsen's master class on the principles of product development flow. He started by asking us why we were there, his point being that our company already knows and uses all the principles of flow in the algorithms in our datacom products every single day — applying them to flows of data packets. This is relevant since the flow objects in product development (e.g., trouble tickets or features) flow in a highly variable environment, just as the flow objects in our products do. Hence, the algorithms in our products could inspire us when we want to improve our processes! Moreover, we prioritize flow efficiency over resource efficiency initially in our products.

So we started storytelling about how the way in which we secure the flow of packets with low latency in our products to achieve both speed and high throughput could provide inspiration for how to secure the flow of features with short lead time and high throughput in our processes. This resonated with almost all the engineers.

Mental Leap 3: From Scattered Experiences to Continuous Innovation

Our heritage was doing large-scale, lessons-learned exercises at the end of our development projects

and root-cause analyses following taskforces at major customer outages (i.e., local ad hoc activities in isolated parts of the enterprise). Our customers were complaining about what they perceived as a lack of corporate memory and the inability to systematically work on improving our ways of working and our products.

Now, what is innovation? Well, we have chosen to define innovation as "value from ideas" so, in addition to having an idea, we also want to prove that it has value (e.g., through a prototype, a trial, or an experiment). With this definition, we can relate innovation to technology, products, ways of working, business models, leadership, strategy, budgeting, recruitment, and so on. However, in the traditional corporate culture, people often think of innovation only as technology and product inventions manifested as patents.

Plan for Innovation

Based on our experience, we suggest trying to plan for innovation by setting aside 30% of time for learning, innovation, and improvements, both in products and ways of working. You need innovation to improve your capabilities and products over time, as shown in Figure 5. It also shows that an idea can come anytime. It is an innovation until it has proven to be valuable, and it requires time to show the value of an idea.

In our case, by having our head of product management and head of product development on stage together in an all-hands meeting asking for 30% of time allocated to innovation, we were really clear on the expectations on the organization. We understood that it, in reality, it would probably not be 30% but rather 5%-10% due to strong customer pull for features. Still, if we had asked for 10%, it would in practice have been close to 0%.

Continuous Innovation Toward the Vision

We have also seen how a long-term, five-year vision — in our case, a vision of half the lead time, four times the value throughput, 10 times the quality, and more than 75% of our people being highly motivated and engaged — helps direct innovation (i.e., experiments, prototypes, trials, and demos) in the desired direction.

We also recommend using one single challenge (instead of a balanced scorecard or objectives and key results [OKRs]) to focus the innovation efforts more toward the area where you have the biggest need currently and to
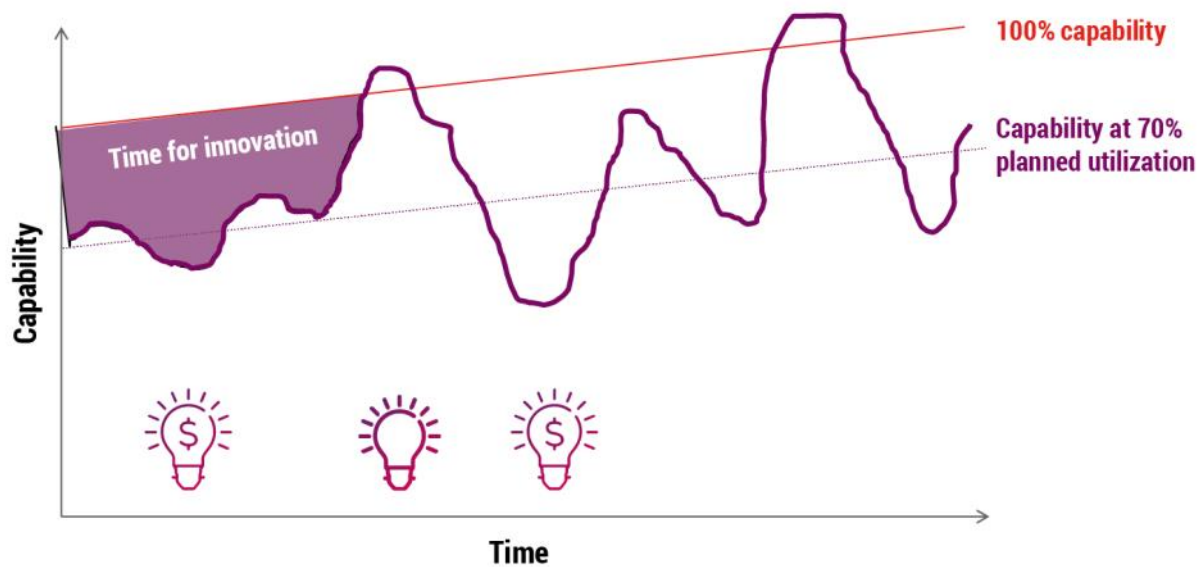
Figure 5 — Planning with less than full utilization creates an environment for innovation.

work with this challenge for one to three quarters, or until you overcome the challenge. Here's an example of a challenge we used: Every sprint deployed to a live network with added customer value and higher quality. Our development teams then used this challenge to come up with experiments, trials, and prototypes that they wanted to try in order to contribute to overcoming the challenge. We tried several different approaches to find out what would work best in our context: a top-down challenge versus bottom-up workshops to identify the challenge, a duration of one-quarter versus several quarters, and a fuzzy challenge versus a clearly defined challenge.

Learning Days

We received feedback that it was difficult to find time for innovation. One example of securing time and space for innovation is our "Learning Day" events, with each Learning Day a full-day, multisite, multitrack, internal conference with content mostly in the form of presentations and workshops from teams, engineers, and leaders, as well as invited external speakers. We run a Learning Day every sprint (i.e., every three weeks). Figure 6 shows an example of a Learning Day.

Most organizations do this at most once or twice a year, if they do it at all. For us, it started with one person having an idea, and it grew into a "self-playing piano," thanks to a wiki page open to all, the regularity of the sprint cadence, and some gentle nudging of people and teams we knew had cool stuff to share.

Remaining Challenges

There are, of course, some areas where we are still struggling, such as the following:

- **Crystal-clear line of sight to customers.** There are still too many organizational layers between teams and customers and almost no interactions between customers and teams.

- **Collaboration with global support functions** (e.g., with HR for speedy recruitment with quality and with finance for more collaborative and flexible budgeting practices).

- **Software "craftership."** Despite several tries, we have yet to see sustainable, self-organized communities of practice working on areas such as clean code and refactoring.

- **Coaches and managers for the future.** Several team/ organization coaches and managers with experiences from this journey left the company or were let go during downsizing, meaning we have lost many people with suitable leadership skills for the future.

- **Battles from within.** How do we continue evolving without becoming a UFO, an alien life-form that other parts of the business want to shoot down?

- **Cross-enterprise sharing.** How do we spread our insights to other parts of the company, where many feel they are too busy to learn and improve?
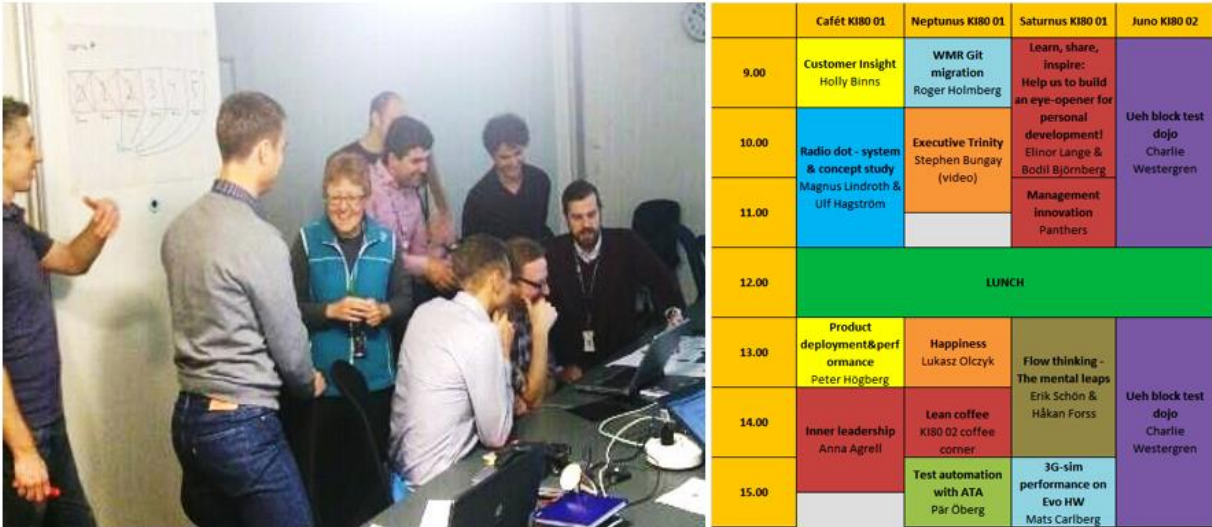
Figure 6 — Learning Day: hands-on workshop (left) and example schedule (right).

	Cafét KI80 01	Neptunus KI80 01	Saturnus KI80 01	Juno KI80 02
9.00	Customer Insight — Holly Binns	WMR Git migration — Roger Holmberg	Learn, share, inspire: Help us to build an eye-opener for personal development! Elinor Lange & Bodil Björnberg	Ueh block test dojo Charlie Westergren
10.00	Radio dot - system & concept study Magnus Lindroth & Ulf Hagström	Executive Trinity Stephen Bungay (video)		Ueh block test dojo Charlie Westergren
11.00			Management innovation Panthers	
12.00	LUNCH			
13.00	Product deployment&performance Peter Högberg	Happiness Lukasz Olczyk	Flow thinking - The mental leaps Erik Schön & Håkan Forss	Ueh block test dojo Charlie Westergren
14.00	Inner leadership Anna Agrell	Lean coffee KI80 02 coffee corner		Ueh block test dojo Charlie Westergren
15.00		Test automation with ATA Pär Öberg	3G-sim performance on Evo HW Mats Carlberg	

Conclusion

Over five years, our large-scale Lean/Agile transformation journey resulted in a quadrupling of value throughput, a doubling of speed, a tenfold increase in quality, and happier, more engaged people who are more innovative. We struggled during the journey and are still struggling in some areas. But our three mental leaps propelled us to continuous improvement. What will be your next mental leap?

Acknowledgements

References

[1]Paasivaara, Maria, Benjamin Behm, Casper Lassenius, and Minna Hallikainen. "Large-Scale Agile Transformation at Ericsson: A Case Study." *Empirical Software Engineering*, Vol. 23, No. 5, October 2018.

[2]Larman, Craig, and Bas Vodde. *Scaling Lean & Agile Development*: *Thinking and Organizational Tools for Large-Scale Scrum*. Addison-Wesley Professional, 2008.

[3]Laloux, Frederic. *Reinventing Organizations*. Nelson Parker, 2014.

[4]Hersey, Paul, and Kenneth H. Blanchard. *Management of Organizational Behaviour*. Prentice Hall, 1977.

[5]Rock, David. *Quiet Leadership: Six Steps to Transforming Performance at Work*. HarperBusiness, 2007.

[6]Poppendieck, Mary, and Tom Poppendieck. *The Lean Mindset*: *Ask the Right Questions.* Addison-Wesley Professional, 2013.

[7]Reinertsen, Donald G. *The Principles of Product Development Flow*: *Second Generation Lean Product Development*. Celeritas Publishing, 2009.

[8]Kniberg, Henrik, and Alistair Cockburn. "Elephant Carpaccio Exercise Facilitation Guide." 11 July 2013.

[9]Modig, Niklas, and Par Ahlstrom. *This Is Lean: Resolving the Efficiency Paradox*. Rheologica Publishing, 2012.

Hacker turned software researcher turned system engineer; manager turned leader turned navigator, Erik Schön has developed people, products, and, organizations for 20+ years in complex, global R&D environments. Recently, as one of the navigators of a large-scale Agile/Lean journey, he helped 40+ development teams become more, faster, better, and happier in a 2,000+-person product development unit in 10 locations. Currently, he heads Erlang Solutions Nordic, evolving the open source Erlang/Elixir/OTP software development ecosystem. He can be reached at erik.schon@erlang-solutions.com.

LabScrum: A Case Study for Agility in Academic Research Labs

by Lisa May and Tamara Runyon

There are many challenges in the work of academic research labs, such as a lack of established planning process, competing commitments requiring frequent task switching, and long delays in decisions. Silos of information create opacity of knowledge, and the individual nature of much of the work can create a sense of demotivating isolation.

So we were curious to see whether an Agile-based project management approach could provide value in the face of these challenges. The Scrum framework seemed like a good place to start — lightweight, yet with more frequent and shorter feedback loops than typical research management. As an experiment, we implemented Scrum with one lab in the Center for Translational Neuroscience at the University of Oregon. We continue to adapt and evolve our method as the implementation spreads to other labs and we work with an ever-growing number of scientists.

We named our adaptation LabScrum to reflect the customization for the academic research lab context. The adaptation from Scrum to LabScrum took a great deal of consideration and experimentation given significant differences in goals, constraints, and environment between industry and academia.

Using LabScrum, lab personnel are seeing increased productivity and increased visibility of short-, medium-, and long-term planning and goals. Personnel are also benefiting from improved graduate student training, increased information sharing/collaboration, better social support, and a positive lab culture.

LabScrum: An Introduction

LabScrum is an evolution of Scrum. The purpose for implementing what has become LabScrum was to increase scientific output as measured by published papers, to share knowledge, and to create visibility for faculty members and trainees conducting research. (*Note: there are many different types of trainees, including post-doctoral fellows, undergraduate students, graduate*

students, and research assistants. Regardless of type, we've used the generic term "trainee" throughout for simplicity.) It is interesting to note that as practices evolved, different labs have adopted different patterns for implementing Scrum.

LabScrum has spread rapidly in an ever-growing number of research labs at the University of Oregon. Currently, it is successfully being used at the university's Center for Translational Neuroscience labs, as well as in other biology, psychology, and human physiology research labs. LabScrum evolved from experiments we ran on the application of Scrum in research environments, adapting our approach from a more formal Scrum implementation to something that stays true to the values and principles of the Agile Manifesto, is based on the Scrum framework, and fits the needs of academic research labs.

There are several traditional issues occurring in research environments that we targeted to resolve with the use of LabScrum, including:

- Difficulty in prioritizing competing projects

- Lack of systems providing structured planning and vision

- Competing commitments and roles (e.g., teaching, research, and clinical work)

- Frequent task switching

In addition, dependencies on overburdened faculty mean long delays for decisions, silos of information, and little to no collaboration or knowledge sharing across lab personnel.

The LabScrum adaptation utilizes all the official Scrum events: sprint planning, daily scrum, sprint review, and retrospective. We included the Agile practices of product backlog refinement and release planning as well. These events and practices are implemented in ways that work best for each unique research environment.

The Scrum roles of product owner, scrum master, and development team are adapted, while staying true to the spirit of the purpose of these roles. Defining metrics that provide value in this context is a continuing work in progress (WIP).

> *The Scrum roles of product owner, scrum master, and development team are adapted, while staying true to the spirit of the purpose of these roles.*

Context of Academic Research Environment

To understand our journey to LabScrum, it helps to understand the context of the research environment. Academic scientific research has two key missions. The first is the production of knowledge. In this perspective, manuscripts published in peer-reviewed journals are the main product. The second is to provide training. In this perspective, scientists are the main product. From either perspective, our work exists without the time, scope, or budget constraints that drive projects in a traditional business environment. This lack of constraints creates unique issues that most organizations do not need to solve.

The academic work environment has qualitatively different controls than those in the software industry where Scrum originated. Typically, there is no profit motive and often very little reporting or budget oversight. For example, there is seldom any financial auditing and no oversight of deliverables or measuring of productivity. Success is measured in terms of papers published and grants awarded, both of which take many years to complete.

Nonexistent Project Management Practices

A key difference between the traditional business environment and the research environment that factors into our use of LabScrum is the absence of a tradition (or culture) of project management in academic scientific research. Due to this lack of focus on best practices and standardized processes, each lab must "reinvent the wheel," leaving most labs' default process to be inefficient and ineffective. Often, this

process consists of weekly one-on-one meetings, typically an hour in length, between the faculty member and trainee, focused on individual WIP. In addition, there may be a weekly lab meeting with all lab personnel to provide feedback on more finished work (e.g., a conference poster, grant application, manuscript, or conference talk). By default, the faculty member is often the only one who possesses the big-picture view of the lab's work; this means that he or she *accidentally* owns the responsibility of repeating information in individual meetings, while all others remain dependent on the faculty member to identify and communicate areas of potential collaboration or knowledge sharing.

Opacity and Few Feedback Loops

Typically, a trainee's main venue for learning is a one-on-one relationship with his or her faculty mentor. In this structure, it is difficult to know what others are working on and nearly impossible to identify opportunities for collaboration without the direct involvement of the faculty member. Weekly lab meetings provide limited opportunities for feedback when deliverables are in a near-finished state. In a culture of not working in a shared space (e.g., working at home, working in coffee shops, working in offices in separate locations), these meetings are often the only time that lab members see one another face-to-face. During this meeting, trainees receive feedback from their mentors and then go back to struggling in isolation.

Inefficiencies, Delays, and Dependencies

The default system described above is inefficient, causing delays by creating dependence on the faculty member for feedback. Trainees often experience delays in their work, waiting a week or more to receive feedback from the faculty mentor in one-on-one meetings. As one faculty member stated, "I have 10 trainees needing feedback from me. I'm always going to be the bottleneck."

LabScrum in Action

Since there is no global process for project management, each lab using LabScrum has made an independent decision to do so. The interest in LabScrum has spread by word of mouth among trainees, formal discussions with faculty at events, and even via pleas for help on Twitter.

Overall, we emphasize flexibility and experimentation in creating successful LabScrum implementations for individual labs and have found that hosting a kickoff event with a needs-assessment and brainstorming session helps a lab's transition to Agile. This allows lab personnel to customize their LabScrum implementation and begin the process of self-organization. This process results in diversity in LabScrum implementation, while maintaining consistency in core practices.

In the following sections, we describe LabScrum's adaptation from traditional Scrum. First, we review the *events* of sprint length, sprint planning, the "daily" scrum, sprint review, and sprint retrospective in the LabScrum context. Following that, we examine LabScrum's *roles* of product owner, development team, and scrum master, and then move onto the application of Agile *practices* in LabScrum, including product backlog, refinement/roadmapping, release planning, and career/training plans.

LabScrum Events

Sprint Length

Sprint lengths vary between one and two weeks depending on the context. Two-week sprints tend to work well in the research environment. One week is generally too short a time period for making meaningful progress on research while juggling other commitments such as teaching or taking classes; however, one-week sprints can be helpful when an individual engages in focused work to meet an impending deadline, such as a dissertation or grant submission. Occasionally, someone working in a lab that uses two-week sprints has expressed a preference for one-week sprints. In these cases, we've been able to meet this need by having that person "split a sprint" and identify sprint goals for week one and week two separately. Overall, sprints are generally two weeks in length and are kept consistent within a particular lab.

Sprint Planning

Sprint goals are identified and communicated during sprint planning, although there is diversity across labs in execution. Sprint goals are typically organized by individuals since much of the work is independent, and the outcome of sprint planning is usually a series of sprint goals written next to each team member's name on a whiteboard. For example, someone might

list completion of a manuscript draft or analysis as a sprint goal. (It is common for individuals to have three or four sprint goals for a two-week sprint.) Sprint plans are often created independently prior to the sprint planning meeting and then adapted based on group discussion of strategy and logistics. This valuable feedback increases success in meeting sprint goals. Most labs spend 30-60 minutes per sprint on planning.

The "Daily" Scrum

Most lab members juggle time commitments from other roles in addition to their research work, which challenges the feasibility of true daily scrums. Instead, these scrums are more like *scrum of scrums* and tend to work better when held two or three times a week. The 15-minute timebox and stand-up method are utilized, but we've found that discussion on progress toward sprint goals is more effective than the traditional "What did you do yesterday? What will you do today? Are there any impediments in your way?" approach since the answers to "What did you do yesterday?" may be work that is irrelevant to research like "grading exams." Discussion of progress toward sprint goals is therefore more relevant to the work and is a key method of sharing knowledge across lab personnel. Many labs post sprint goals on a whiteboard in a shared lab space and refer to it during the scrum. Having sprint goals visible and directly acknowledging them in the scrum encourages a greater focus on priorities, which can otherwise get lost in the midst of many competing time commitments.

Sprint Review

A day or two before the end of the sprint, lab members identify product that could benefit from review and create an agenda. Not every sprint goal creates work appropriate for review, so a lab might review and give feedback on one to three items during a 90-minute meeting. This is a critical venue for getting feedback on work product and often results in lively discussion that can stretch to fill up a great deal of time when not enforcing a timebox.

Sprint Retrospective

Constraints on schedule and roles can make sprint retrospectives challenging to arrange, but this meeting has value on multiple levels. In our observations, the sprint retrospective:

- **Helps labs identify improvements to their shared process.** For example, in one retrospective, a lab realized that impromptu meetings in the shared lab space were causing distractions and decided to repurpose storage space for impromptu meetings.

- **Helps lab personnel identify improvements to their own independent processes.** Here, a lab member realized that being more assertive was a necessary trait when working with a collaborator at another university.

- **Helps lab personnel provide each other social support, which is critical for surviving the pressures of academia.** For example, an exhausted new parent received encouragement, validation, and offers to babysit.

> *LabScrum practitioners organize the product backlog, refinement/roadmapping, and release planning either around a project or an individual.*

LabScrum Roles

Roles in academic research do not directly parallel the Agile worlds' scrum master, product owner, or development team members. Thus, we have loosely translated the Scrum roles to a lab roles paradigm. Each is subject to adaptation and can be altered based on changing context from lab to lab.

Product Owner

The faculty member acts as a chief product owner on large lab-wide projects, while trainees act as product owners on their own individual research projects. The faculty member serves as a chief product owner for these individual projects, setting direction, with budget authority and advisory capacity, but not final say in design decisions.

Development Team

While a trainee may be product owner for major portions of a project, responsible for idea generation, design, and planning, this person may also act as a development team member for execution of the work. Many trainees utilize the efforts of undergraduate and other volunteers to help collect data and then handle the analysis and writing portions independently.

Scrum Master

We see more variation and adaptation with the scrum master role. One lab has a scrum master who splits time about 50/50 with other work, but most labs do not. In other labs, the faculty member takes on scrum master duties, or lab members self-organize to fill this need. For example, one lab has a sheet posted where lab members can sign up to facilitate the sprint retrospective.

Agile Practices in LabScrum

Product Backlog, Refinement/Roadmapping, and Release Planning

LabScrum practitioners organize the product backlog, refinement/roadmapping, and release planning either around a project or an individual. They formulate these techniques on an as-needed basis to facilitate long-term strategizing regarding when and what work to prioritize. For example, a student's three-year plan for her dissertation work helped her identify that a faculty member who was providing important statistical training was going to be gone for sabbatical the following year. This knowledge allowed the student to strategize a means of obtaining the necessary training before the faculty member left, thus avoiding a six-month delay to her work.

We've defined different levels of planning according to the planning horizons. Planning for more than a year in the future roughly equates with *high-level product roadmapping*; planning within an academic year but further out than two weeks equates with *release planning*; and planning within two weeks correlates with *sprint planning*.

Career/Training Plans

Career/training plans are an element of LabScrum not found in traditional Scrum. They are adjacent and related to the product-focused plans discussed above. These plans encourage lab personnel to consider their research work from the perspective of scientific training, not just scientific output. Trainees articulate and connect their long-term training goals to concrete actions that can be part of a sprint. In many labs, faculty and individual trainees meet quarterly to discuss and update training plans.

LabScrum Benefits

The response to LabScrum has been overwhelmingly positive. Faculty report dramatic time savings, as evidenced by these statements:

- "Turns out there was a project meeting every week I didn't need!"

- "I turn work around in two or three days instead of in two weeks."

- "There's 10 more hours in my week I have free to work that I was spending in trainee meetings."

An environment of group learning, shared problem solving, and social support combat isolation and ignorance. LabScrum labs have been described as "ridiculously happy," and trainees report that rapid feedback increases productivity. They obtain feedback in scrums two or three times per week instead of just once weekly in one-on-one meetings, and receive formal feedback from faculty (e.g., manuscript draft edits) more quickly due to open time not spent in one-on-one meetings. Long-term release plans and training plans allow individuals to identify time-oriented goals, and ultimately, manuscripts spend less time stuck in unending revisions due to the ability to communicate time goals and prioritize progress.

LabScrum fosters better work habits, helps separate planning from execution, and increases the use of effective planning strategies. Improved and shared documentation increases institutional knowledge and reduces rework. A structure for making competing time commitments visible helps prioritize work. Increased visibility of WIP increases productivity not only through problem solving, but also through shared knowledge. For example, lab mates have shared new software discoveries as evidenced by this overheard statement: "There's an (statistical programming) R package for that! You don't have to do it by hand!"

Moreover, a focus on colocation has made huge impacts on the efficiency and effectiveness of work. A shift from a culture of individual work toward a culture of work in a shared space has been a major benefit from adopting the LabScrum framework. This shift required the creation of appealing work environments, with labs purchasing comfortable furniture and making snacks available. One faculty member shifted entirely to working in the shared lab space with the trainees, which has yielded several positive results. For instance, increased communication; instead of having to compose an email

and wait for days to get an answer, one can quickly check in verbally, and/or schedule a time for a lengthier discussion. In addition, the value of overhearing conversations in this shared workspace is not to be discounted. Listening creates shared knowledge of the lab's work and surfaces colleagues' novel ideas, potential collaborations, and social support.

> *An environment of group learning, shared problem solving, and social support combat isolation and ignorance.*

LabScrum Challenges and Insights

An enlightening moment in our experiments occurred when we realized that the Scrum events more closely mapped to a scrum of scrums across different projects rather than a traditional single scrum team with planning, review, and retrospectives within a single project. Each trainee led his or her own "projects" with independent goals. Once this realization occurred, we were able to better adapt the events and roles to reality.

Another major aha moment was introducing release planning and making those plans visible. Long-term plans in particular have been a benefit of LabScrum. In some cases, just creating long-term plans is highly innovative. In other cases, increasing the visibility of plans has allowed people to seek strategic input from stakeholders, thereby increasing their ability to inspect and adapt. A key realization was the importance of the development of detailed backlogs before the creation of release plans. This shift increased the utility of the planning process by identifying gaps in knowledge and potential barriers early to avoid delays.

In addition, one interesting development surfaces when considering metrics. Specifically, considering when metrics create useful knowledge versus when metrics increase reporting load unnecessarily. Since research scientists operate almost exclusively as independent decision agents, metrics for reporting to oversight are not necessary. Valued metrics of success come in the form of papers published (particularly the quality of the journal and how often other scientists cite the paper) and funding (in the form of federal research grants) awarded.

With the lack of deadlines in most research environments, there is no strong incentive to predict when work will be complete, diminishing the usefulness of traditional Scrum metrics such as velocity. Another reason we've found the metric of velocity and/or capacity to lack utility is the ever-changing landscape of external commitments. Instead of being dedicated to a specific team or a project as traditional scrum team members would be, researchers are always juggling time commitments from multiple roles outside. The ebb and flow of those commitments makes calculation of capacity inconsequential.

Interestingly enough, however, calculating story points (the Agile estimate of overall effort) does still have value, but we see the value largely in the self-reflection stimulated by the act of assigning points. Moreover, the Agile estimating technique of "T-shirt sizing" product backlog items is quite useful for individual strategy and decision making but not for oversight or forecasting future delivery.

Ongoing Adaptation

LabScrum continues to adapt and change to meet the evolving needs of lab personnel. Areas of ongoing interest include developing best practices around managing roles, the potential utility of user stories, productive discussion of impediments, and defining useful metrics. We work to understand how the funding structure could support dedicated project management staff or how to work within the funding structure to create LabScrum roles beyond the ad hoc manner in which they occur today.

To date, user stories have not been adopted in LabScrum; instead, work is currently framed as tasks in the product backlog. User perspectives such as research participants and journal editors/reviewers exist, but researchers report that the time commitment to create user stories for individual work is overly burdensome as it provides little if any added value.

Lastly, there is ongoing discussion among labs about how to structure productive conversation around impediments. In this culture of individualized work and juggling many commitments, common

impediments such as task switching and finding time to complete work are systemic in nature. To address this, we experimented with a new approach of reporting on needs that the lab can meet. This is a subtle shift in perspective. For example, instead of someone focusing on days of work lost because of a sick child as an (unresolvable) barrier, he or she might report on the need for help reprioritizing tasks because of lost days of work.

Conclusion

Over the past two years, upon learning and adapting our Agile practices, we have discovered deep benefits from working with the LabScrum framework. The most notable benefits include reduced time in meetings, increased knowledge sharing, better problem solving through increased collaboration, and more effective work prioritization. Many of our lab personnel have also reported increased quality of life using these practices.

By adopting structured feedback loops, utilizing the events in the Scrum framework, and adapting Scrum roles, LabScrum brings great benefits without vastly increasing overhead. The faculty member enjoys increased time to mentor students, and the added visibility helps with better feedback and collaboration for the individual trainees. In a nutshell, participants report that LabScrum supports increased productivity in the lifeblood of academic research labs: writing grants, completing necessary research, and publishing papers.

Lisa May is Assistant Director of Operations for the Center for Translational Neuroscience at the University of Oregon, where she developed LabScrum as part of her Agile coaching. Previously, her research focused on the brain chemistry by which psychology impacts pain perception. Dr. May earned a PhD in social neuroscience from the University of Oregon. She can be reached at lmay@uoregon.edu.

Tamara Runyon is a Scrum trainer and enterprise Agile coach with more than 30 years' experience in business management and software development. Ms. Runyon is founder and President of Advanced Project Management, Inc. (APMI), as well as Applied Scrum LLC, an Agile consulting business and division of APMI focused on helping organizations adopt and scale Agile practices. She can be reached at tamara@appliedscrum.com.

Agile HR: The New Way to Design Employee Experience

by Zuzana Šochová

The more organizations shift toward Agile, the more they need to redesign how they work with employees, how they search for new employees, and how they nurture employees' development and careers. This article highlights the different HR functions in an Agile organization and explains the fundamental shift HR needs to make to support agility. Indeed, in an Agile organization, HR must shift its focus to the overall employee experience, choosing an employee-centric approach over the governance role that traditional HR departments often hold.

Recruiting

Knowledge and skills are no longer the key factors organizations seek. Agile organizations build on collaboration, encourage innovation, and need high flexibility. Experience is applicable only to a certain extent. More important than being an expert with a deep but narrow specialization are having an open mind, being able to learn, and collaborating with others to deal with complexity and unpredictability. If you don't believe this is the case, take a look at your own career. Are you still working in the same specialization as when you started your career? Most people have changed careers more than once, and the pace of change is increasing.

Should you still care about hiring experts with a particular specialization? Not really, as they create silos and prevent your organization from changing business direction. Agile organizations need people who are ready to learn and to inspect and adapt. People who are not afraid to take on responsibility and conduct experiments. People who are not inflexible, who do not say, "we have always done it this way," and who are ready to change their way of working in accordance with business needs.

In this environment, it is very hard to create a traditional job description based on skills and experience, as those may soon be irrelevant. A new advertisement for an open position might instead say:

We are looking for an enthusiastic, flexible, and open-minded person who is ready to take on responsibility and collaborate with others on achieving value. We are a team-oriented organization with a flat structure, which will support you in your personal growth. Join our team for a day to experience our culture. Together we can [achieve the vision].

> *Skills are easier to learn than a mindset.*

This is quite different than a traditional ad, right? When we tried it, no recruiting company was ready to support our needs. They wanted to know how many years of Java experience the position required and asked for a description of the position for which we were hiring. Yet, for us, it didn't matter whether we were looking for developers or a new CEO; thus, there was quite a mismatch between our needs and the recruiter's objective.

Eventually we realized that hiring new graduates is the easiest solution for most team positions. New graduates are flexible, have ideals, and are eager to learn. We had only to create a team-learning environment based on pairs and teamwork that allows new hires to catch on fast. We realized that learning is easier than unlearning old habits, so, very often, getting fresh graduates up to speed was easier than hiring senior-level employees with individualistic habits that can create more harm than help in a team environment. This is a difficult message to hear for anyone who believes that years of experience count and should result in a higher salary. That may be the case if you are working in government, but it is not necessarily so in the Agile space, as recruiters for Agile companies may not care at all about years of traditional company experience.

Unfortunately, we had a similar experience with executive search companies. It didn't matter how "big name" the recruiting company was; they often had no idea of what Agile was, and they weren't helpful in assessing candidates, or in finding relevant candidates.

If you start looking for a leader with executive experience with an Agile mindset, you learn very quickly that such people are hard to find. Most executives are used to acting as directive managers within traditional hierarchical organizations. Again, it's often easier to grow leaders from within your organization than to hire them externally.

> *Hiring is more about creating relationships than assessing skills.*

So if we can't rely on experience and skills or count years worked, how shall we decide whether the person is the right match? It's the same as in any other relationship: we start "dating." In this case, it's about acquiring personal experience and starting to build a relationship together with the candidate to give both sides an opportunity to learn whether they are compatible on a cultural level. For that reason, we always invite some team members to the interview to describe a typical day, to see whether the description resonates with candidates, and to talk about the candidate's dreams and visions. All this is to determine whether there is a match. Once a candidate "passes," we extend that prospect an invitation to lunch with the team. Informal conversation is critically important to learn about each other.

Finally, it's good practice to offer candidates the opportunity to spend a day at the company to try it out and see how it will feel once work commences. You don't begin a relationship based on what someone writes or says about himself or herself. You are together because of who the other person is. It is similar with hiring — mindset and culture are difficult to change. Skills, however, can be learned.

Evaluations and Performance Reviews

Once you have hired the right person for the team, it's time to start thinking about evaluations and performance reviews. In traditional organizations, this is pretty simple. Each employee has assigned tasks, and each task is evaluated and linked to a particular key performance indicator (KPI). In Agile organizations,

it's not that simple, as multiple people collaborate on the same task. Even if you try to set some KPIs at the beginning of the year, they mostly become irrelevant somewhere along the way, leaving nothing to evaluate at the end of the year.

The simplest practice used in Agile environments is to set team goals instead of individual ones. The risk still exists that the goals will become obsolete over the course of the year, but at least you are supporting the team collaborative culture. A slightly better option is to break the year cadence and create shorter goals. After all, there is nothing magical about a year cadence when we deliver product regularly. A good practice is to let teams design their own goals, but you need a high level of trust to be able to move in this direction.

A sufficient step on your Agile journey is to replace evaluations with coaching conversations focused on employee development. As with every coaching conversation, it is not about the coach but about the "coachee" (the employee, in this case), and the conversation should focus on raising the employee's awareness of his or her abilities and potential. When done well, a coaching conversation can skyrocket a person's performance. The downside, unfortunately, is that not many managers are good coaches, which is a limiting factor in most organizations.

If you are ready to be truly Agile, consider running regular, frequent retrospectives instead of any form of evaluation. Together with radical transparency, retrospectives will create enough clarity about performance toward the sprint goal, the product vision, and the entire organizational purpose to allow people to adapt in a very efficient way. These are a simple and powerful tool. We refer not only to a team retrospective, which provides powerful peer feedback, but also an overall retrospective at the multiple-team level, as, for example, in LeSS (Large-Scale Scrum), and an organizational retrospective, which can be facilitated by Agile HR using, for example, the World Café Method[1] or Open Space Technology.[2] These retrospectives will engage employees in solving team, cross-team, and organizational issues and increase their motivation to devise creative and innovative solutions to better deliver value and achieve the organizational purpose.

The frequent retrospective cadence provides regular feedback that results in fast change and small improvements on a day-to-day basis and prevents the big

disappointments and surprises of traditional performance reviews, which often create demotivation and stress. Issues are resolved more quickly, before they become too big and poison the team or department, and people get help to work on those issues early, ideally from their peers. Your organization might not be ready for this approach tomorrow, as a culture of transparency and trust has not yet been established, but you can go step-by-step until no one feels the lack of KPIs, performance reviews, or formal evaluations, and frequent feedback and opportunities for inspection and adaptation become the normal way of working.

At this stage, we often stop using "Agile HR" in favor of "talent development" because the entire focus of the HR organization is changing to support the overall employee journey and, to a fuller extent, employee development. Supporting coaching and mentoring programs and creating an environment for effective peer feedback are just a few ideas of where to start.

Career Path and Salaries

As mentioned previously, positions are not that important in Agile organizations. People collaborate, take on responsibility, and become leaders as needed, not because of some line in a job description. In traditional organizations, it's all about the position. We hire to fill an empty position. A position determines what people do and don't do. A position indicates employees' potential and what future position they might hold if they receive good evaluations and are promoted. And last but not least, the position defines the salary rank.

This whole concept breaks down once the individual position ceases to have much importance and a team environment is created where people self-organize their work and collaborate based on their skills and abilities. Such a shift quite naturally creates a need for fewer positions. For example, one Scrum development team with which I am familiar has no roles, just team members. Indeed, positions can follow the Scrum organizational design: instead of having the software developer, the software tester, and the analyst, you can just have the one position of software engineer, or, simply, team member. Every position potentially creates a silo and gaps, dependencies, and a need for synchronization and handover; in other words, nothing that would help you create high-performing teams.

In More Agile Environments

If the scenario described in the previous paragraph hasn't caused too great a shock, you are ready for step two. When team members work to achieve the same goals, conduct frequent peer reviews, and hold each other accountable to improve their skills, the only reason for positions and career paths is the direct correlation with salary. The solution is obvious: decouple salaries from positions. In that case, you don't need any positions at all, as the team roles are emergent, based on the team's current needs to achieve its goals. Salary can be linked to peer feedback and to an individual's value to the organization as a team member.

> *In more Agile environments, we decouple salaries and positions and make all roles emergent.*

It's a startup mindset. Imagine for a moment that you are not an employee but an entrepreneur, and every day you need to prove that you bring enough value to get paid. Stressful? Maybe. Be aware that every practice like this requires a certain culture and organizational agility. I would not start the Agile journey with it. However, you can make this your next step and be ready to take it when your organization is ready. If you feel you are ready now, here are two possibilities to choose from for getting started.

The first possibility is a sudden and hard change. Give employees two options: either stay because they believe in the change you are making and are ready to take on ownership and the responsibility to succeed and achieve the organizational purpose, or take a leave package of x amount of salary and go. The people who stay are those with the right mindset, and any transformation will be so much easier.

The second possibility is a gradual change. Start with decoupling salaries from positions. Sooner or later, positions become irrelevant and no one would miss them anymore, so you remove them.

You must have courage to choose the first possibility. On the other hand, if you lack that courage and choose the path of gradual change, you will make your journey

long and painful, but that may be the best path for your organization. The choice depends on what you want and where you are. Agile is not about practices, it's about mindset, and there are many ways to achieve it; it all depends on circumstances. The same is true for Agile HR, or talent management, as well as for organizations.

> *Being a leader is not a position, it is a state of mind. Anyone can become a leader.*

In Agile Organizations

The more Agile you become at the organizational level, the more flexible and dynamic your team structure is and the more difficult it is to define positions or roles. And the more Agile the way you work, the higher your need for transparency at every level. In a very Agile organization, everyone can see what everyone else is doing, and anyone can challenge anyone else and give feedback. Any employee can join any initiative, but with the responsibility of directing all effort toward achieving the organizational purpose. As nothing is hidden, in a way, everyone controls everything.

Emergent servant leadership is the key that links everything together and ensures the creation of harmony instead of chaos. Such environments are ready to make all salaries transparent and allow employees to take part in the decision making. To be fair, not many companies have reached this point, so don't feel you have to get there tomorrow. But you can still be inspired.

Leadership, System Coaching, and Large Group Facilitation

Finally, let's take a look at what is required of good Agile HR. Primarily it's about understanding the Agile mindset and having the ability to create an environment where an Agile culture can flourish (see Figure 1). These environments support collaboration, transparency, open peer feedback, trust, team spirit, ownership, empowerment, and responsibility. The more Agile your organization, the greater the need for coaching and facilitation skills. Of critical importance is HR's role in growing coaching and facilitation skills in the organization and supporting individuals and teams with education on coaching and facilitating and then guiding them on their journey.

Another fundamental shift is from management based on decision making and delegation to leadership not created by a position but by a state of mind. Anyone can become a leader. Individuals decide when they are ready to take ownership of and accept responsibility for leading an initiative, team, or product. Peer feedback will create enough self-awareness for leaders to emerge throughout the organization. Very often we speak about emergent leadership as a person's acting as the leader of

Figure 1 — Agile HR.

one initiative while at the same time being a team member on another initiative. As traditional evaluations transform into regular peer feedback and coaching for development, the key goal of leaders is to help other leaders grow, creating an increasing and inevitable need for good coaching and facilitation skills. These are the skills that HR encourages and develops.

HR changes the focus in Agile organizations to the overall employee experience, but this is only the beginning. Good HR acts as an organizational scrum master — or Agile coach, if you like — operating at the third level of the #ScrumMasterWay concept,[3] focusing on the overall system. At this level, HR does not focus on coaching individuals but on coaching teams and organizations as a system, leveraging tools from system coaching, such as Organization and Relationship Systems Coaching.[4] More than team facilitation, Agile HR must have the ability to facilitate large groups with hundreds of people, leveraging tools like World Café and Open Space. Indeed, Agile HR must model Agile leadership, growing the "we culture" and mentoring other leaders to grow into Agile leaders.

> Agile HR = Agile leadership + system coaching + large group facilitation.

References

[1] "World Café Method." The World Café.

[2] "What Is Open Space Technology?" OpenSpaceWorld.org.

[3] #ScrumMasterWay (https://scrummasterway.com/).

[4] "ORSC™ Organization and Relationship Systems Coaching." CRR Global.

Zuzana "Zuzi" Sochová is an independent Agile coach and trainer and a Certified Scrum Trainer with the Scrum Alliance with more than 15 years' experience. She has implemented Agile transformation in many companies and teams around the world. By creating and sustaining Agile leadership, Ms. Sochová believes the worlds of work and life can be happier and more successful. She is the founder of the Czech Agile Association and the AgilePrague *conference, a member of the Scrum Alliance board of directors, and an author of* The Great Scrum Master: #ScrumMasterWay. *She can be reached at zuzi@sochova.com or via Twitter @zuzuzka.*

Do We Need More Transparency?

by Emilio Gutter

Transparency has become a popular issue in recent years — and for good reason. After all, it was only a decade ago when we went through one of the biggest global financial crises, caused by large investment groups and credit rating firms that obscured the real risk of financial products to their clients.[1] More recent examples are the Panama Papers — exposing the dirty secrets of hundreds of world leaders, celebrities, and businessmen — made available to the public thanks to the International Consortium of Investigative Journalists; the Facebook–Cambridge Analytica data scandal, which revealed how large corporations can manipulate and share personal information without user consent; and the mountain of fake news[2] that has affected and continues to influence elections in different parts of the world.

> If everyone in the company is going to participate in decisions, then everyone should have access to the same information.

Despite society's push for transparency, when we move the discussion of transparency outside the public sphere to the domain of individual corporations, there does not appear to be a common understanding of what transparency means, whether it's actually needed, or if it has any benefits at all. In fact, articles from prestigious publications warn about the risks of transparency and promote explicit boundaries to strengthen privacy.[3] These concerns about transparency, however, seem to contradict the values and principles that are the foundation of the Agile philosophy. Concepts such as trust and collaboration, communication and openness, empowerment and self-management are an intrinsic part of the Agile DNA. Indeed, most Agile methodologies have some kind of practice that requires a fair amount of transparency: meetings, ranging from planning meetings to daily stand-ups to retrospectives; Scrum, Kanban boards, and other information radiators; or more technical practices like collective code ownership and pair programming.

A Tale of Success

Back in 2009, along with three partners, I had an idea for a new experiment. Frustrated with classic company management, we wanted to do something different. Inspired by the Agile Manifesto principle, "The best architectures, requirements, and designs emerge from self-organizing teams," we asked ourselves whether we could extend the principle to *the best products and services are delivered by self-organizing **companies***. Yes, we decided to use the Agile values and principles to manage not just a small team, but the entire company. To support self-organization throughout the whole company, we established a fully transparent information system, as we believe it is not possible to make good decisions without good information. If everyone in the company is going to participate in decisions, then everyone should have access to the same information. (The exceptions are apprentices and new hires, who must go through a trial period of three months before obtaining access to all information.)

A decade later, our small experiment had turned into a profitable company with more than 70 employees and clients worldwide. An external visitor could come by our office on any day, pick any random employee, and ask him or her for the salary of any other employee (regardless of the seniority of that employee or whether the employee was one of the founders), the profit margin of any client, or how much money was spent last month on any given purchase. That employee may not know the answer by heart, but it would take him or her less than 10 seconds to find the exact figure.

We strongly believe our self-organizing model based in full transparency has been key to our success. Let's take a look at each of the main positive results our company has been able to achieve thanks to this model:

- **Low turnover.** Our average turnover rate is four to five times lower than the industry average. This is a key feature for a knowledge-based organization, as our main asset is our intellectual resources: our people. When an employee leaves, our clients are affected, and our company loses value. Moreover,

a low turnover rate has allowed us to maintain a higher level of technical quality than our competition. Our clients notice the advantages and, as a result, most of them stay with us over a long period of time.

- **Sustained profitability.** Throughout all these years, there has not been a single quarter when we haven't been profitable. Every employee understands profit margins and what affects them. All team members are aware of the profit margin of the project they are working on and are able to analyze variations when they happen. We also do not spend time or money on useless things. Moreover, HR directors do not decide on employee benefits; rather, employees can present an idea and, if it makes sense and the expense for the benefit is reasonable, then it is approved. As an example, an employee recently proposed an extension to paid paternity leave. The employee explained the reasons and how much money it would cost the company. It was a sound idea at a reasonable cost, so the change was approved. Everyone knows every cent not spent wisely will affect the annual bonus, so nobody throws money away carelessly.

- **Pay equality.** Thanks to several recent movements (the most well-known of which may be #MeToo), many companies have started to pay attention to various inequalities within their organizations. At our company, not only is everyone's salary disclosed to everyone else, but discussions of salary raises are open, and anyone can share an opinion about whether to raise someone else's salary. This environment makes it very difficult, given a similar role, skills, and experience, for a man's salary to be higher than a woman's. There is still a long way to go to achieve equality in all areas, especially in the software industry where significant cultural barriers discourage women's participation.[4] Nevertheless, I truly believe open salaries are an important first step to solving the problem of inequality.

- **Strong commitment.** One of the most common complaints I hear from managers and company owners is that employees are not "committed" to their work. "How to motivate your employees" is a must-have discussion in every management course I've seen. Our company does not feel the pain of that problem. We don't have managers; instead, we have leaders (which is a role, not a position, that anyone can take in different situations). Therefore, we don't have to pay managers to ensure that workers are doing their jobs. Our people always work to the best of their capabilities, without anyone having to prod them.

Challenges

Transparency can create some side effects that might raise concerns and bring new challenges to many companies. In my experience, I have found the following challenges and accompanying solutions:

- **Loss of power and control.** Knowledge is power. Therefore, distributing knowledge is the fastest way to distribute power. A good example is the invention of the printing machine, which enabled the massive distribution of books, and which remains one of the biggest revolutions in history. At the same time, by sharing power, you lose some control over other people. If I have a great idea and I want things to happen without being able to impose control, then I have to convince other people of the value of my idea. This requires more time and skills, including leadership and mentoring. For example, if I think we should hire a software developer to meet a specific client need and want to offer a higher-than-usual salary to convince that person to take the job, I first need to secure the consent of the rest of the organization. As everyone will immediately know about the offer, to make this move without that consent would create conflict and lead to an even bigger problem.

> *One of the most common complaints from managers and company owners is that employees are not "committed" to their work.*

- **Slow decision-making process.** Given that decisions are open and visible to scrutiny, they should be substantiated and communicated properly to avoid potential misunderstandings. This requires more time and preparation, which can slow down existing processes. The interest and level of participation in the decision-making process are very likely to increase. As everyone is aware of what is going on, more people want to get involved; it would be counterproductive to deny them the opportunity to share their thoughts and ideas. With this changed decision-making process,[5] the time needed to make decisions increases and new practices, such as consent decision making or the advice process,[6] need to be incorporated. The good news is that total time from the start of the decision process until decision execution may very likely not increase significantly or might even start to decrease. With more people

informed and involved in decision making, there is greater alignment toward the goal and fewer misunderstandings down the road.

- **Frictional costs.** A decline in the number of staff members is another potential consequence of introducing transparency and self-organization. For example, when everybody has access to salary information, then a group of interested people, rather than an HR specialist, can conduct salary negotiations with employees or new hires. Empowered teams start to make their own decisions and people start building new cross-functional teams to make things happen outside their everyday duties. A recent example at our company was the need to lease a new office space and refurbish it. A self-managed team of a few employees assumed the responsibility, inquired about prices, hired contractors, bought equipment, and took all the necessary steps, within the limits of a preapproved budget, to deliver the new office space ready for use.

The downside of this type of transparency is that the loss of specialized staff can introduce new frictional costs associated with learning and making mistakes. Depending on how tolerant the company culture is to errors and rework, and how good the organization is at sharing lessons learned and avoiding repeated mistakes, these costs may be minimal or could be much larger.

> *The business world is not separate from the broader society; changes in society influence organizations and vice versa.*

Increasing Transparency in Existing Organizations

Increasing transparency has a direct effect on the organization's culture. If the culture is not prepared for transparency, then the consequences may be catastrophic. In a toxic culture, any attempt by upper management to increase transparency will be seen as a means to increase control over lower-level employees. Successful transparency requires cultural values, such as trust, respect, honesty, and courage, as well as a strong belief in collaboration over competition; consolidated habits around continuous learning and improvement; and constant communication with timely

feedback. When all or some of these cultural aspects are lacking, the cultural transformation effort will be harder and will take longer than would otherwise be the case.

Whatever the situation, given that each company is unique and human reactions are not predictable, the transformation effort toward a more transparent company should be done incrementally. Plan a small experiment and carry it out in a controlled environment, inspect the results, learn from them, and start over again. An interesting approach is to start applying Modern Agile[7] principles in small teams. One of its principles, "Make Safety a Prerequisite," is fundamental to remove existing fears around sharing information, expressing ideas, and asking tough questions.

A simple and safe experiment for almost every company is to implement company-wide weekly stand-up meetings (depending on company size, this might involve a few or many different teams or areas). It's important to keep these short and to have a single representative from each team or group speak to share with the rest something interesting that happened during the last week. This could be an achievement, something new that was learned (e.g., a new technology), or something funny that happened to someone on the team. After a few stand-ups, when everyone is feeling safer about sharing, start sharing mistakes, errors, or big failures, along with any lessons learned.[8]

Final Thoughts

The business world is not separate from the broader society; changes in society influence organizations and vice versa. We cannot expect society to change unless we first change ourselves and our workplace, the place we go every day and where we spend much of our time. Corporations are having a growing influence on the rest of society. What we consume, read, watch, write, and share ultimately affects what we think and how we act. Many of the abuses committed recently by large corporations against their employees, consumers, and sometimes the public in general could not have happened in a fully transparent organization sustained by Agile values. Company owners and C-level executives can choose to lead organizations that become a better place for humans to collaborate and thrive, transforming how people interact with each other by replacing fear and self-interest with trust, tolerance, and respect. Or, they can focus solely on the accumulation of power, money, and fame.

Do we need more transparency in our organizations? I believe so. And I hope a change toward transparency will bring about many other positive changes that carry over to the rest of society.

References

[1]Crotty, James. "Structural Causes of the Global Financial Crisis: A Critical Assessment of the 'New Financial Architecture.'" *Cambridge Journal of Economics*, Vol. 33, No. 4, 1 July 2009.

[2]Allcott, Hunt, and Matthew Gentzkow. "Social Media and Fake News in the 2016 Election." *Journal of Economic Perspectives*, Vol. 31, No. 2, Spring 2017.

[3]Bernstein, Ethan. "The Transparency Trap." *Harvard Business Review*, October 2014.

[4]One of our female members has written an excellent article about these cultural barriers to women; see: Alonso, Micaela. "Where the Female Developers at?" 10Pines (blog), 16 October 2018.

[5]"Decision Making." Reinventing Organizations wiki.

[6]Laloux, Frederic. *Reinventing Organizations*. Nelson Parker, 2014.

[7]Modern Agile (http://modernagile.org).

[8]A similar idea is behind the Failure Institute's "Fuckup Nights."

Recommended Reading

Beck, Kent. *Extreme Programming Explained: Embrace Change*. Addison-Wesley Professional, 2nd edition, 2004.

Cockburn, Alistair. *Agile Software Development: The Cooperative Game*. Addison-Wesley Professional, 2001.

Foucault, Michel. *Power/Knowledge: Selected Interviews and Other Writings, 1972-1977*. Pantheon Books, 1980.

Wilkinson, Hernan. "Growing an Open, Participative, Horizontal and Based-on-Trust Company." *Agile Alliance Experience Report*, August 2016.

Emilio Gutter is cofounder of 10Pines, an Argentina-based boutique software development company. Inspired by Agile principles, he established a self-organizing management model, with decisions made by consent and where all information (even financials and salaries) is available to everyone. Mr. Gutter has more than 20 years' international experience under multiple roles, including Agile coach/ trainer, technical leader, and software engineer. He has worked with varied-sized companies worldwide, from startups to large corporations, in several countries such as Argentina, Brazil, England, US, France, Romania, and Bulgaria. Mr. Gutter is a frequent speaker at technical and business conferences; organized Agiles 2008, *the first Agile conference in Latin America; and chaired that conference in 2012. More recently, he organized the first Erlang and Elixir conference in Buenos Aires. Besides his work at 10Pines, he is a professor at the University of Buenos Aires in the engineering department, teaching Agile practices to undergraduate students. He can be reached at egutter@10pines.com.*

Beating Production Expectations by Embracing Agility

by Simon Orrell

The Problem

Companies focused on the exploration and production of energy resources face an inherently complex problem: finding and modeling energy-rich formations is fraught with uncertainty. Financing and executing on a portfolio of energy assets comes with public market pressures in the midst of unpredictable performance characteristics within and between formations. A North American energy-producing company with annual revenues of US $6 billion, 2,000 employees, and trading on both the New York Stock Exchange and the Toronto Stock Exchange, was looking to address how best to embrace that uncertainty. It was also seeking to understand how to change its status-quo pattern of finding new solutions only when it was abundantly clear it would be unable to meet its yearly market targets. In short, it was looking for a different approach to conducting its operations organization.

> *Together, with the use of a "breakdown/breakthrough" model, the program far exceeded stakeholder expectations.*

Background

The company's previous approach to the problem had been rooted in maintaining the status quo of siloed disciplines working on initiatives that were not necessarily aligned. Various business units, including Surface/Land, Road and Lease, Development Planning, Drilling and Completions, Facilities, Operations, Geology/Geophysical, and Mid-Stream Marketing, were all working to improve their piece of the puzzle. It was not until relatively late in the year that people would start to think about how to react to progress indicators that didn't appear to be leading to success. In Q2 2018, the company found itself in a familiar situation: it was unable to see how it was going to meet its 2018 commitments to the market.

This same company had, for several years, already experimented with the application of Agile values and principles in the engineering, procurement, and construction of a program of three $800 million natural gas–processing plants. This program had embraced a mindset, roles, events, and artifacts typically associated with Scrum to overcome clashing vendor cultures, misalignment in goals, and significant costs of delay. Together, with the use of a "breakdown/breakthrough" model,[1] the program far exceeded stakeholder expectations. Despite the industry norm for oil and gas upstream megaprojects having an average cost overrun of 34% and an average schedule overrun of 41%,[2] these gas plants came online from 38 to 119 days early and all under budget. This decreased time to value had a significant impact on the economics of the program.

The Approach

To address the organization's problem of not being able to meet its 2018 hydrocarbon volumes and budget targets, leadership agreed to perform more organizational experiments. To that end, I encouraged leadership to focus on the following:

- Creating a multidisciplinary team aligning on outcomes

- Iterative learning and reprioritization based on empirical evidence using techniques such as:

  o Limiting work in progress (WIP)

  o Rhythm of typical Scrum-inspired events

- A mindset of challenging the status quo on everything

We used the approach outlined in John P. Kotter's XLR8[3] to guide us in creating a multidisciplinary "network" team that would complement the existing hierarchy. This dual operating system enables rapid innovation and development of new ideas while still allowing the organization to optimize its day-to-day operations.

The Experiments

Creating a Multidisciplinary Team Aligning on Outcomes

A sense of urgency within the organization already existed. A significant shortfall of hydrocarbon liquids to date indicated that the company would likely miss its commitments to the market. Leadership agreed to form a multidisciplinary team to focus on aligned outcomes that would serve to solve the problem at hand. This team consisted of at least one representative from each of Surface/Land, Road and Lease, Development Planning, Drilling and Completions, Facilities, Operations, Geology/Geophysical, and Mid-Stream Marketing. These representatives were to be dedicated to this multidisciplinary network team, which meant the organization had to discover how best to backfill their roles on their respective functional teams. While, initially, leadership had committed to "clearing the decks" for these representative team members, all had varying degrees of difficulty extricating themselves from their hierarchical responsibilities. This took constant vigilance and required leadership intervention with the hierarchy at times.

Aligning the team on outcomes consisted of two key facets: a common understanding of the goal and a way to visualize progress toward that goal. Articulation of the goal was relatively easy, as it could be expressed as a shortfall in hydrocarbon liquid volumes, with recovering from that shortfall and producing projected liquids the definition of success. The organization had not yet grappled with visualizing progress toward that goal. One of the key activities for the team was establishing a dashboard that could be updated daily to provide feedback regarding the outcomes that both the network and the hierarchical teams were accomplishing. Essentially, it provided a way for people to see where they stood in relation to the goal, every day.

Iterative Learning and Reprioritization

To begin addressing the problem at hand, the team created a backlog of ideas that it thought might have an impact on achieving the goal. The team used a mind-mapping technique to both connect the ideas and visually track which were in progress, completed, or to be ignored. The team's product owner was responsible for prioritizing that backlog of ideas. To do that, he asked the team to estimate the potential impact (in terms of liquid volume) of each of the ideas toward achieving the goal. With these scientific wild-ass guesses (SWAGs), the team was able to create what would come to be known as the "salmon ladder" — a representation of the steps/outcomes that would take the team toward the goal.

Iteratively, the team would select a subset of the backlog (based on potential impact and timing) and devise multidisciplinary experiments/plans that would provide more information on what the impact was likely to be for each idea. Then, limiting WIP, the team would execute those plans/experiments and iteratively see the impact of its work, stepping up toward the goal. Sometimes the steps were larger and sometimes smaller than anticipated. Sometimes the experiments or plans would result in a realization that the idea was not worth pursuing. Regardless, the team was able to see both its progress and what was currently the next highest priority (to either execute or learn) based on value.

> While, initially, leadership had committed to "clearing the decks" for these representative team members, all had varying degrees of difficulty extricating themselves from their hierarchical responsibilities.

This approach to solving problems meant the team essentially adopted a rhythm of Scrum-inspired events: backlog refinement/prioritization, iteration planning, daily scrum, iteration review, and iteration retrospective. Refinement focused on understanding the value and prioritization of items near the top of the backlog and ensuring clarity on knowing how the team would define "done" for each. Iteration planning was done asynchronously by team members and then reviewed as a group for alignment. The daily scrum focused on progress and roadblocks to the plan as well as "wins." Far more people than the team members attended the daily scrum, as it became the central mechanism for the "hierarchy" to understand priorities and impediments. The iteration review focused on articulating what had been completed, what had been learned, and what that meant for upcoming priorities. This was also the opportunity to discuss what the team thought needed to change in the hierarchy in order to adopt what had been learned. Finally, the retrospective allowed the team to discuss what was and was not working, whether within the network team or between the network team and the existing hierarchy.

Challenging the Status Quo

For the team, the overall vision had been set, and leadership had committed to enabling the team to meet goals that currently seemed improbable. To do so, the team had also committed to focusing on true "leadership" rather than "management" — where "management" is defined as minimizing risk and maximizing predictability, and "leadership" is defined as enabling the creation of something currently unpredictable. It is the role of leadership to continually enlist recommitment to the vision. Leaders speak in declarations of what is possible and what will be. This is fundamentally different from the language of management, which is the assertion of what is predictable based on historical evidence.

> *Once breakdowns were identified and prioritized as being significant enough to work on immediately, a team member usually volunteered as the owner.*

"Breakdowns" are defined as any gap that exists between what has been committed to and what is currently predictable. Breakdowns should be embraced as opportunities for innovation and extraordinary performance. "Decommitting" is the act of allowing a set of circumstances to distract the team from honoring its commitment. A "breakthrough" is the result of managing a breakdown (avoiding decommitment) so that the gap between what is predictable and what is committed to is eliminated.

Typically, embracing a breakdown/breakthrough model requires the team and leadership to:

- Set a vision

- Enlist team members in the vision

- Identify breakdowns

- Manage the breakdowns to create breakthroughs

Because of how the network team was operating, the vision had already been set (recover from our shortfall), the team had already been enlisted, and a system had already been implemented to identify and surface breakdowns (the Scrum-inspired events). Further, the team had also agreed to a working agreement whereby

any impediment or roadblock it encountered could be worked on by an individual for only 24 hours before being elevated to the team and possibly beyond. This ensured that the team had multiple people thinking about and collaborating on potential resolutions as quickly as possible.

Once breakdowns were identified (whether as impediments to existing plans or unknown approaches to desired goals) and prioritized as being significant enough to work on immediately, a team member usually volunteered as the owner. That person took ownership of the coordination of the planning and activities that needed to occur for the breakdown to be managed to a breakthrough. Typically, this involved coordinating multiple discussions with multiple team members with experience in the intricacies of the breakdown gap. These discussions focused on challenging the status quo but took many forms/combinations, including:

- Business value chain analysis

- 5 Whys

- Socratic method

Two of the most useful questions, as simple as they might seem, turned out to be:

- "What's preventing us from achieving X?"

- "What would have to change in order for us to achieve X?"

The most difficult part of managing breakdowns is avoiding decommitment. There are many organizational, cultural, and personal mechanisms that result in impediments. Part of challenging the status quo involves challenging those mechanisms that have become normal work and communication practices. Some examples for this particular organization included:

- Treating the sending of an email as a transfer of ownership

- Allowing the apparently urgent to distract from the important

- Notifying people of a missed commitment the day the commitment was due

- Communicating apparent commitments without due diligence

- Responding to challenges to the status quo with, "That's not how we work"

In any team member's toolbox, maybe the most important tool to deal with decommitment is courage — the courage to speak out when seeing him- or herself, or other team members, decommitting. Part of modeling leadership includes the ability to speak with honesty and respect about our own propensity to decommit. Modeling this behavior at the highest levels of the organization encourages and enables others to adopt this behavior.

The Results

The outcome for this team was its ability to meet and slightly exceed its hydrocarbon liquid production targets for 2018. In conversations with team members at their end-of-year celebrations, many attributed their accomplishments to the reduction in silos created by the multidisciplinary network team and the iterative and incremental focus on solving valuable problems. Further, the team stayed intact. Both the team and the organization realized that having a network team focused on solving multidisciplinary problems should be the new norm, not just a remedy to a crisis.

The organization has now added second and third network teams, each focused on separate geographical plays. This in turn has resulted in 2019 "stretch targets" being reset *twice* as the three teams continue to exceed the organization's expectations. The three teams meet on iteration boundary days to review their priorities with leadership, both for alignment and awareness.

The Future

The experiments will continue. We'll be borrowing some concepts from Large-Scale Scrum (LeSS) as the three teams hold joint iteration review sessions with leadership to help all teams understand each other's accomplishments, ensure priority alignment with

current corporate goals, and create opportunities for learning across teams. We'll also try team representative retrospectives in an attempt to grapple with organization-wide impediments that show themselves on multiple teams.

Further experiments will center around three areas:

1. Support for other parts of the organization (e.g., Legal)

2. Crowdsourcing of ideas to challenge the status quo

3. Setting up communities of practice for scrum masters to share and learn from each other as they continue this journey

References

[1]Scherr, Allan L., and Michael C. Jensen. "A New Model of Leadership." Harvard Business School NOM Working Paper, No. 920623, 10 September 2006.

[2]Asvadurov, Sergey, et al. "The Art of Project Leadership: Delivering the World's Largest Projects." McKinsey & Company, September 2017.

[3]Kotter, John P. "Accelerate!" *Harvard Business Review*, November 2012.

Simon Orrell became one of three Scrum Alliance Certified Enterprise Coaches in Canada in 2010 and has been working with individuals, teams, and leaders to help transform oil and gas, financial services, manufacturing, and construction businesses. His career in product development and project execution has spanned a multitude of domains, and his background as engineer and recovering project manager provides him with a pragmatic approach to product, team, and organizational evolution. Mr. Orrell specializes in the initiation and ongoing coaching of leadership and Agile principles, values, and techniques within organizations with an emphasis on product and portfolio management techniques for both colocated and distributed teams. He focuses on applying Scrum principles in broader business contexts, and his groundbreaking work on coaching a large energy company to apply Scrum values and principles to construction megaprojects helped produce extraordinary outcomes. Mr. Orrell is a frequent speaker at numerous Agile and project management conferences across North America and Europe. He can be reached at simon@snowdolphin.com.

Cutter Consortium Membership
The Ultimate Access to the Experts

Cutter Consortium Membership opens up multiple avenues to interact with Cutter's experts to brainstorm and gain guidance to transform your organization and boost success.

Like everything business technology, one size does not fit all.

That's why we encourage you to choose the Membership that's right for your organization. Whether you choose Digital Transformation & Innovation, Enterprise-wide, Practice-specific, or CIO Membership you'll see a strong return.

Contact us at +1 781 648 8700 or sales@cutter.com to arrange a sample inquiry call with a Cutter expert and see for yourself how quickly your return on Membership can be realized.

What Do You Get from Cutter Membership?

- Get guidance in leveraging new strategies, emerging technologies, and business management practices to enable digital transformation and boost competitive advantage

- Learn how to mine data to create new products and services and improve customer experience

- Get input on how to reduce expenses through more cost-effective strategies

- Gain insights and get ideas on achieving sustainable innovation, successful change management, and prudent risk management

- Get coaching and insights on leadership and team-building practices that boost productivity

- Discover vendor-agnostic advice to ensure unbiased purchasing decisions

Test Drive Cutter Consortium Today

Have a question? A challenge you're trying to overcome?
Have you reached a bottleneck and need some expert advice?
Find out for yourself how valuable Cutter Membership can be —
request a sample inquiry with a Cutter expert now.

Call +1 781 648 8700 or email sales@cutter.com.

CUTTER CONSORTIUM
●●● *Access to the Experts*

37 Broadway, Suite 1
Arlington, MA 02474 USA
Tel: +1 781 648 8700
Fax: +1 781 648 8707
Web: www.cutter.com
Email: sales@cutter.com

"The information technology revolution continues to change how businesses work and how people work in and with them. Today's demands on software to provide useful, usable, secure, and scalable services stress all aspects of systems development. These include architecture, technology selection, product development, and lifecycle management. Cutter provides comprehensive coverage for these crucial topics — through research reports and customized virtual training delivered globally — helping us to understand needs and solutions that span traditional silos."

— Eric Schoen
Director of Engineering, i2k Connect LLC
Former Chief Software Architect,
Schlumberger

"Thanks to Cutter´s experts, in a short time we have been able to:

- Improve our capabilities, using the contract scorecard for outsourcing
- Gain key insights and practical strategies for responding to challenges we face in building trust and partnerships within the organization and outside our institutional borders.
- Improve our methods for working with people and anticipating the challenges of cultural transformation within our organization.
- Have immediate access to key 'experts' at critical times, who spend quality time with our teams, provide state-of-the-art orientation, and help us implement projects.
- Implement agile, effective, and articulated innovation around a common vision for achieving quality education in Mexico."

— Arturo Cervantes
Director General of Information Systems
for Evaluation Results, National Institute for
Educational Evaluation, Mexico

"I have personally been able to leverage Cutter's services since 1999. Among the attributes that differentiate Cutter from other firms, two remain at the top of my list: **Thought Leadership and Real Value**. Thought Leadership is driven by Cutter's experts. Real Value, driven by 'Access to the experts,' pushes us beyond just understanding the options. We can develop relationships with the experts, and tailor the options they present so that they can be quickly and practically executed within our organization, enabling our Business Technology team to continually improve, engage, and contribute to business growth."

— Doug Mikaelian
VP Business Technology,
Dairy Farmers of America

Business Technology Journal

About Cutter Consortium

Cutter Consortium is a unique, global business technology advisory firm dedicated to helping organizations leverage emerging technologies and the latest business management thinking to achieve competitive advantage and mission success. Through its research, training, executive education, and consulting, Cutter Consortium enables digital transformation.

Cutter Consortium helps clients address the spectrum of challenges technology change brings — from disruption of business models and the sustainable innovation, change management, and leadership a new order demands, to the creation, implementation, and optimization of software and systems that power newly holistic enterprise and business unit strategies.

Cutter Consortium pushes the thinking in the field by fostering debate and collaboration among its global community of thought leaders. Coupled with its famously objective "no ties to vendors" policy, Cutter Consortium's *Access to the Experts* approach delivers cutting-edge, objective information and innovative solutions to its clients worldwide.

For more information, visit www.cutter.com or call us at +1 781 648 8700.