Oxygen’s Agile Adoption
January 2004 - October 2007

Executive Summary

This experience report summarizes Oxygen Media’s agile adoption, our history, the benefits we derived and suggestions for other groups introducing similar practices.

Our agile practice encompasses four years during which our team grew, we expanded our portfolio from marketing, ad sales and broadcast operations to consumer facing windows and web applications across a range of technologies: Microsoft Win Forms, ASP.NET, SQL Data Analysis Services (OLAP), Vista compatible Windows Presentation Foundation, XAML, open source .NET MVC frameworks and Ruby on Rails.

With agile practices, we became a performing, self-managing team recognized within the industry for both our software and our contribution to practices and platforms.

The team attracted and retained top talent, increased code quality, breadth of knowledge and value delivered. Oxygen Media software has been recognized within Oxygen, NBCU, industry peers, Microsoft and user communities for its invention, simplicity, and effectiveness.

We provide eight broad recommendations. The most important of which is to understand that Agile adoption is about organizational change. It will surface false assumptions, inefficiencies, and dysfunction in the organization. Success comes from acknowledging and removing them.

Background

The Oxygen software development team began adopting agile software engineering practices, Extreme Programming (XP), in January 2004.

Before this, performance was hit and miss. Success relied on individuals. We had few shared practices. Our goal in going "Agile" was to consistently perform across projects.

“Agile” declares a set of common values and supportive practices. It fosters collaboration with customers and shared ownership within project teams.

In time, the team became proficient in core XP practices:

Planning
- User stories
- Iterative development

- Tracked velocity
- Daily stand-up meetings
- Regular retrospectives with continuous improvement

Designing
- Simple system metaphor
- Use of development spikes
- Refactoring

Coding
- Onsite customer
- Pair programming with switching
- Test driven development (TDD)
- Continuous integration
- Collective ownership
- Sustainable pace

Testing
- Extensive unit test coverage
- Bugs are resolved within the iteration
- Acceptance testing by the onsite customer

Within a year the team's performance was more consistent and visible. We were measuring our velocity and reliably delivering on our 30 day iteration goals.

We discovered our project management practices had become a bottleneck. We were hitting idle periods within and around projects because of a failure to efficiently describe and prioritize work.

We introduced Scrum as a management framework on top of XP. It provided practices for organizing and prioritizing work. It helped us define roles and responsibilities.

We clarified our expectations of internal clients and achieved more efficient interactions overall. We created mechanism for reporting progress and costs to senior management.

In Q1 06, the team's practices were evaluated by an Agile Coach, Jason Lewis. Among his findings:

> Oxygen Media's Agile software development process overall rates above average and is better then the benchmark team. The benchmark did have considerably more Agile experience, but less time together as a team.
>
> In the evaluation of practices, the team was overall: 1) well above average to outstanding in the adaptive learning practices, 2) Above average in Sprint practices and 3) Average in planning practices. High points for the team's individual practices were the retrospective and use of the wall for

iteration tracking. The one low point was the maturity of acceptance testing.

When comparing roles to the benchmark team, the benchmark team had a much better customer role but the team was stronger in the developer and facilitator roles. When comparing the team's adoption of the practice's versus the benchmark the team was generally more effective. Iteration tracking was one key area the benchmark team was better, however, the team was much stronger in the all the adaptive learning practices.

After the audit, we pre-staged our iteration planning with a subset of the team reducing our planning overhead, we reduced our iterations to two weeks and formally planned releases syncing the business and the development team with material 3-6 month business goals.

We added discipline to our acceptance testing. Product owner (proxies) described acceptance tests in a narrative script exercised first by the team then by the product owner (proxies).

We never automated acceptance tests for rich windows applications or systems tied to large, volatile back end data stores. But by Q3 2007, the team was using automated acceptance tests on it's web applications achieving 30% coverage on our last project.

The most drastic improvement however was in the customer role. Scrum defines the responsibilities of the product owner. In our case, that role was divided into two individuals.

The product owner, is an empowered single authority for prioritizing business value at the feature level. They are usually are executive level and work in the business unit "funding" the work. They also have working knowledge of the system to be built. Product owners participate in planning and review, and are available for ad hoc questions within iterations.

The product owner proxy is a member of the development staff who acts as onsite proxy for the product owner. This person assists in authoring user stories and maintaining a product backlog, meets regularly with the product owner, and acts in their place to broker decisions within the development team during iterations.

By Q2 2007, the team had product owner proxies for both our IT and our consumer facing work. Product owners included the VP of Broadcast Operations, VP of Ad Sales Traffic, our CTO, and our CEO.

Members of the team have written and presented on aspects of this transition and of our evolving concept of "Collective Product Ownership" and it's place in innovation and delivering value to end users.

Team Composition

By Q2 2007, the team was being asked to brainstorm and develop consumer web sites in addition to continuing work on Ript, babynamer.com, and internal solutions for Ad Sales and Broadcast Operations.

Our team had the following staff:

Development Team:
- 1 Lead,
- 1 Coach,
- 6 Developers,
- 1 User Experience Designer

Product Team:
- 3 Product Owner/Proxies: (Ript, Babynamer, IT Projects)

Scrum Team:
- 1 Scrum Master

We describe how this organization functioned in detail in our experience report for the "Agile 2007" Conference.

Ript: Innovation and Collective Product Ownership, Ken H. Judy & Ilio Krumins-Beens, Agile 2007

Benefits

Qualitative Measures

Given our size and focus, measures of success are largely qualitative:

Team Cohesion

The Gallup organization uses the Q12 assessment as a semi-annual employee engagement index. It is composed of twelve questions. The more agreement the higher the correlation to retention, customer loyalty, safety records, productivity, and profitability.

We administered the "Gallup Q12" twice.

Early in Agile Adoption responses were 70% agreement, 30% disagreement.

Late in Agile Adoption responses were 80% agreement, 15% neutral, 4% disagreement.

The most improvement was in daily opportunities “to do my best” and daily feedback on performance and expectations.

By the Q1 2007, a re-assessment would have resulted in better results as camaraderie and connection to mission fell into place with additions to the team and the Ript project. Rather than re-take the Q12, we elected to be the only department to execute a 360° performance review further evidence of employee engagement.

Test Coverage/Code Quality

XP practices enforced methodical unit test coverage, mutually arrived at coding conventions, and real-time code inspection by multiple members of the team. The team went from no unit test practice to comprehensive coverage over the business logic layer of each application. A user story, test-driven approach to development has been shown to reduce defects in final testing by 40%.

XP and Scrum practices force conversations between the development team and product owner that incentivize all to build quality into the software rather than allowing technical debt to accumulate and relying on downstream QA process to fix the application.

In 2006-2007 there were no business impacting failures of internally authored software.

Reduced Risk

While any team has experts, “Agile” practices reduced our reliance on “specialists”. The entire team was capable of working on and maintaining any aspect of the code base. We passed the “bus test”; despite our small size, no project was at risk if any given member of the team became unavailable.

Leadership

Our team raised our skills and began contributing to our field. We write, present and teach at conferences on topics of scrum, XP and platform as well as contributing to open source projects and developer knowledge bases.

Recruiting and Retention

After establishing “Agile” practices we recruited skilled candidates from higher paying positions who desired to work in our culture and with our practices. We received inquiries from noted contributors in the Rails and .NET communities from

as far away as South America and Europe. Despite the reputation of our team and market demand we retained staff.

An additional benefit is that pairing provided an inherent and efficient on-boarding process for new hires. Developers joined the team provided immediate contribution rather than requiring a gradual ramp-up.

Workplace Diversity

A 2006 paper by McDowell, Werner, Bullock and Fernald found that pair programming practice "may help increase female representation in the field."

Agile values and practices support a collaborative, empowering and sustainable work place. Such environments support diversity and take advantage of the breadth of experience each worker represents.

Pair programming improves student retention, confidence, and program quality, Communications of the ACM, Volume 49 , Issue 8 (August 2006)

Client Satisfaction

> "Working with the agile Software Development team has been rewarding on many levels…it's a team that celebrates creativity, organization, listening, feedback, openness, honesty…and is proof positive that a great process results in great product. I look forward to our very regular meetings (I even readjust my travel schedule as much as possible to not miss anything) and am never disappointed. They are an engaging and engaged group of individuals." - Gerry Laybourne, CEO

> "[SES saved] half a head in on air scheduling and a full head in my team." - Michele Taylor, VP Broadcast Operations

> "The sales reports written by our development team are the guiding-light to all of our ad sales decisions. Gabriel has a vast wealth of information about our sales activity but no good reporting and our in-house reporting team enables us to divine meaning from the mountain of data." - Dan LeRoy, VP Traffic Operations

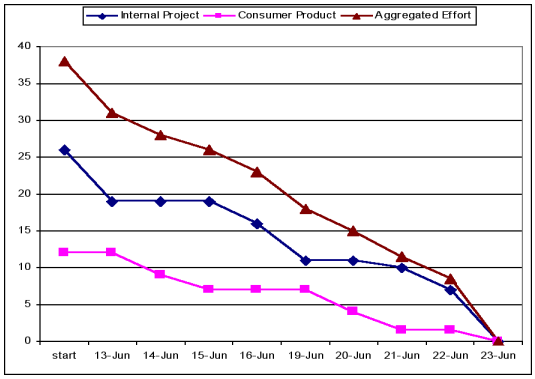
> "We also use Gabriel to traffic our primary logs for all three of our broadcast networks but I have heard about your software technology to traffic secondary events. We currently do that through manual operators but I'd like to understand how you do that more sometime and how it works…" - Gwynne McConkey – SVP, Operations, IS & IT Lifetime Entertainment

> “Given the complexities of driving secondary on-air events that includes the combined limitations of automation, graphic and traffic systems I believe Oxygen has created a solution that has proven to be much more capable than most systems than I've worked with.” - Griffin Moore – CMC Broadcast Center

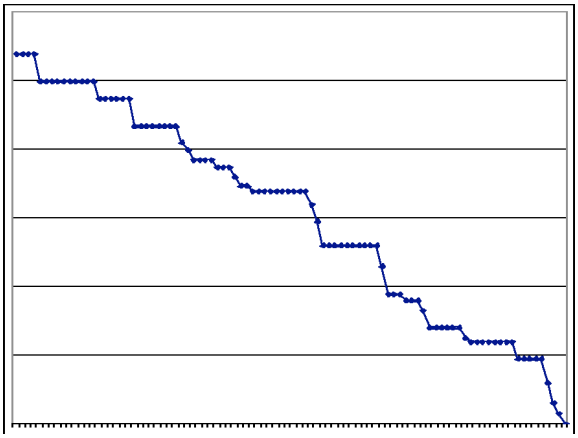
Productivity, Frequent Delivery, Adaptability

Throughout 2006-2007 our team of 3-8 developers balanced two simultaneous lines of work on diverse projects built in Microsoft Windows Forms, ASP.NET to SQL Data Analysis Services Data Warehouses, Vista compatible Windows Presentation Foundation and XAML, open source .NET MVC frameworks and Ruby on Rails.. Including a rich windows application built on beta Microsoft Technology, Ript.

The team completed eight IT and three consumer projects while doubling head count from 5 to 10 (+2 contractors). We initiated our consumer product initiative and achieved our first release on Ript with a six month allocation of effectively 1.5 - 2.5 developers.



Iteration Burn Down



Ript Usability Release Burn Down

Invention/Innovation

Agile practices evolved from Lean management and associated knowledge creation theory. In this, it shares ancestry with Six Sigma.

Agile is based on empirical not plan-driven process control. It is closer to lean product development than lean industrial manufacturing.

Lean product development models sustained innovation as a process of knowledge creation and conversion within an organization that acquires and shares learning in an cycles within and across teams and up and down from leadership.

Agile fosters true joint work which is the only form of workplace collegiality that advances organizational change and innovation.

Using Agile Practices to Spark Innovation, Ken H. Judy & Ilio Krumins-Beens, Hawaii International Conference on System Sciences 2007

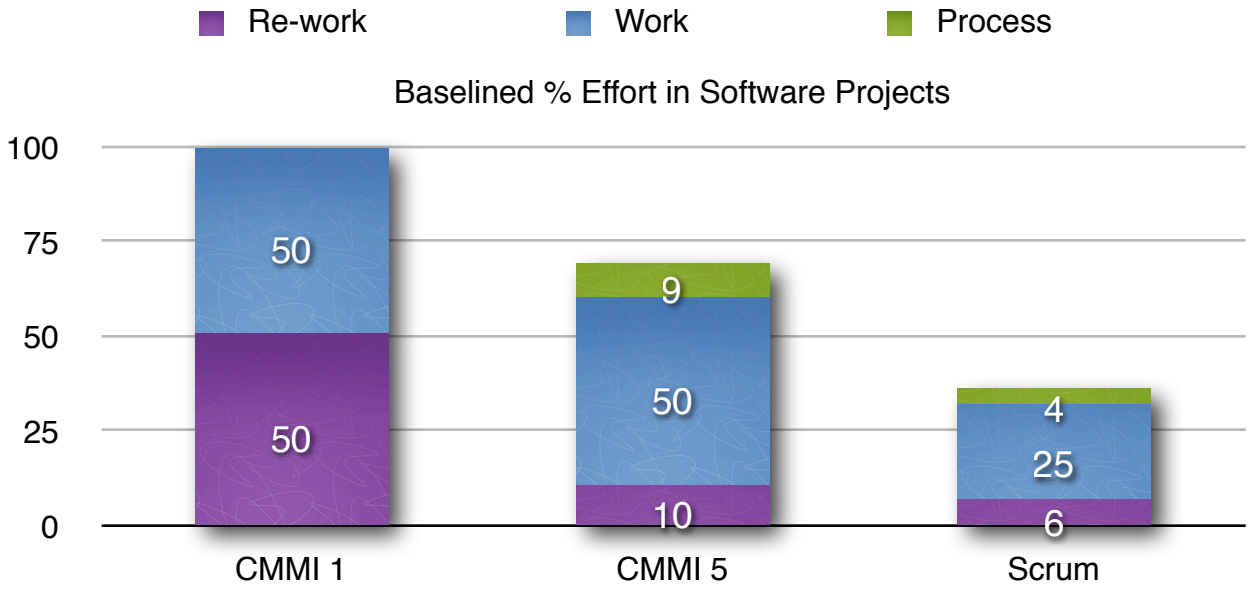
Unbounded Collaboration and Collective Product Ownership, Ken H. Judy & Ilio Krumins-Beens, Hawaii International Conference on System Sciences 2008 (Best Paper Nominee)

Our consumer product, Ript™, was recognized for its design and implementation by Microsoft's platform and developer evangelist team as well as by the WPF team. It achieved high ratings in usability testing with end users (avg rating 8 of 10) and showed potential to deliver on its revenue targets.

Quantitative Measures

Jeff Sutherland, the co-creator of the Scrum process, authored a paper on Scrum methods introduced into a highly efficient CMMI Level 5 software development organization, a government contractor to the European Union.

Because CMMI requires detailed metrics capture and analysis, this provided a unique opportunity to measure the effects of Agile introduction.



CMMI and Scrum Productivity Gain

Having achieved 30% efficiency gains in waterfall practices moving from CMMI level 1 to CMMI level 5, the company achieved an additional 50% efficiency from CMMI 5 waterfall by introducing Scrum. These results apply across a wide range of software projects from large to small, web to embedded systems.

As a result, this company bids out contracts based on these assumptions. Projects mandating the ISO standard, plan-driven waterfall process are bid at an estimate price. **The same project is bid at half that price if Scrum/Agile processes are allowed**.

The company is able to retain it's CMMI Level 5 standing and practice Scrum/Agile. They provide thorough documentation at the end of the delivery cycle rather than up-front. They build the spec iteratively in the form of as-builds culled from the tracking mechanisms inherent in their agile engineering practices. This imposes overhead largely on support staff and management not on development teams.

Scrum and CMMI Level 5, Sutherland, Jakobsen, Johnson, Hawaii International Conference on System Sciences 2008

How we introduced "Agile" practices

Understand the commitment

Agile adoption doesn't fix an organization. It rallies all parties around delivering business value and describes practices that highlight waste and dysfunction.

Eliminating that waste and dysfunction is the hard work that changes an organization.

Start with a motivated group of individuals

Obtain buy in by starting with a team of volunteers. Use their success to establish a process for rolling practices out to the organization.

Build in-house expertise

Identify staff with experience in Agile practices. Build an in house team whose goal is bringing about the agile adoption. Provide training to staff that join new agile teams.

Every member of the Oxygen team received Scrum training early on to give them an introduction to the values and expectations of Agile.

Acquire in-house expertise

Hire experienced coaches for both agile development and agile project management practices. Oxygen sought out developers with XP experience.

Use outside coaching

At the start, bring in experienced practitioners to train your trainers and help form a vision around agile adoption.

Bring experts in on a regular basis to audit your agile practices and to introduce new ones.

The Oxygen team had an audit performed by an experienced XP coach as well as retaining experts to coach on release and business planning.

Provide committed executive support

Empower teams to self-organize around the work at hand.

Empower product owners and hold them accountable to delivering measurable business value.

Remove obstacles escalated by the teams.

Reward and incentivize collaborative performance.

Constantly inspect and adapt. Confront and eliminate waste.

Regularly retrospect at iterations and milestones, identify high-priority impediments to performance, and remove them.

February 5, 2008

Ken H. Judy
VP, Software Development, Oxygen Media
Certified Scrum Practitioner
IEEE Certified Software Development Professional

Ilio Krumins-Beens
Product Owner, Oxygen Media
Trained Scrum Master
PMI Project Management Professional

Salim Divakaran
Scrum Master, Oxygen Media
Scrum Practitioner Certification Pending