

Executive Summary

This white paper explains how Primavera Systems, a vendor of project portfolio management solutions, turned around its development organization in 2003. In terms of value to the company, the development organization went from having low confidence in its ability to deliver and repeated failure to meet expectations, to being cheered for a release that was the hit of their user conference, with good quality and twice the expected functionality. Bonuses were forthcoming for this release. Magic? No, just leadership, hard work, and using a process that turned the leadership and hard work into results. These are the Agile processes

Primavera, a 21 year-old software company, sells project portfolio management solutions to help firms manage all their projects, programs, and resources. Primavera was thriving, and its growth was leading to increasingly complex client needs; this put a strain on its ability to release a product that pleased its entire customer base. Throughout 2002, the development organization worked overtime to develop release 3.5. As with other projects in the past, the last three months were particularly bad; the developers sacrificed weekends and home life to get the release out with all of the new requirements. The result – a release seen by management as incomplete and three weeks late, and an exhausted team with low morale.

Primavera decided to try the Agile development processes Scrum and XP to fix its problems. Scrum is an overarching process for planning and managing development projects, while XP prescribes individual team practices that help developers, analysts, testers and managers perform at peak efficiency. Though they are often implemented separately, Scrum and XP are even more effective when implemented together. Primavera adopted Scrum first to improve the way it managed product development, then adopted XP practices to upgrade its product quality and then customized the amalgam to suit its own needs.

The result of Primavera's experiment is a highly satisfied customer base, and a highly motivated, energetic development environment. Of equal value, everyone within Primavera now has a process for working together to build the best releases possible, and is aware of, and participates in, the tradeoff decisions involved. People who haven't had a chance to work together in years put their shoulders to making each release a success, from CEO, CTO and VP's to the entire development organization. When the experiment started, Primavera was a very quiet, subdued place to work. It now feels like a vibrant community.



Primavera Looked for Something Better

Bob Schatz came to Primavera a few years ago as the new Vice President of Development. He brought with him experiences from his career at General Electric and then a software start-up. Steeped in the principles of leadership, Schatz made a number of changes to increase the developers' productivity and teamwork; these included new office space, motivational meetings and programs, and realigning the organization. The incremental improvements that he achieved were all very positive, but still inadequate to meet the needs of the company, and its customers.

The turning point, Schatz points out, was the day the development team released a new version of the Primavera product. The team thought it had nailed the release. The team had worked a lot of overtime, had given up many weekends and felt like it had sacrificed a lot for the company. But the company turned out to be so disappointed in the release that the entire development staff was denied their yearly bonuses. That day Schatz resolved that the team would never go down that path again. He knew it was the development approach, not the developers, that were the problem.

Primavera's development process was a traditional waterfall. Requirements came from marketing in the form of a document. The development staff and project managers would then use the document to plan the company's next release. They would break the work into tasks and organize them in a nine- to twelve- month project schedule. Hundreds of tasks were divided among their analysts, documenters, designers, programmers, and testers. Project management was a command-and-control structure where a few people who made decisions were never close enough to the actual work. The complexity of each release increased as the customer base grew. More and more requirements were being changed or added in the middle of release schedules. The relationships between development and the other departments were strained. The releases were not meeting the ever-changing market needs and the time frames between releases were unacceptable to their customers. Just to start a release, marketing and development would have to meet for several months to devise a plan that would optimally employ all of the staff on the release. Unfortunately, the plan went out-of-date the day it was published as the marketplace and customers demanded changes. Change management, or as it was called then, Change Control was a very painful process that often erupted into arguments.

At an offsite directors meeting in 2003, Bob was looking for a better solution to slipping morale and difficulty in responding to the frequent changes. Bob investigated a number of approaches, and encountered the development process called Agile. First described in the Agile Manifesto in February, 2001 (www.Agilealliance.org), Agile processes had been around since the early 1990's. Their reputation was that they were useful in handling complex development in rapidly changing, almost chaotic environments and they focused on creating a sustainable pace for the development team. Bob thought, "sounds like us, let's investigate!" After a search, Bob decided to first investigate Scrum, an Agile process for managing complex projects. He invited Ken Schwaber, one of the authors of Scrum, in to Primavera to discuss Scrum with himself and his management. Afterwards, even the most skeptical person was willing to give Scrum a chance. With



the support of Primavera's CEO, Joel Koppelman and CTO, Dick Faris, Bob brought Scrum and Schwaber in.

Start With Scrum

Bob started by training all of his managers as ScrumMasters, the Scrum equivalent of a project manager. It is the ScrumMaster's role to facilitate collaboration between the marketing department and development teams to build increments of functionality in monthly iterations.

The first iteration, or "Sprint" in Scrum terminology, made significant progress on the primary risk of release 4.0 – integrating two separate software systems. Primavera had decided that a key feature of release 4.0 was the integration of a new workflow and collaboration system into their project management software. This integration was extensive and devilishly difficult.

A review was held at the end of the first Sprint. The team demonstrated some project management portal functionality to management and executives. Since one of the Scrum rules is that only increments of potentially shippable product functionality can be demonstrated at the end of each Sprint, this was solid, tested, documented functionality. Dick Faris, Joel Koppelman and most of the management team were present, along with the product marketing managers and team. As the functionality was demonstrated, everyone became ecstatic. The risk was probably removed, because the entire product worked seamlessly. Frankly, nobody could believe what they were seeing just one month into the development cycle.

This initial success gave Scrum the credibility within Primavera for it to proceed. A starting point had been established – in code – for extending the workflow and collaboration to all part's of Primavera's new release. The team had proven to themselves, and everyone else, that it could build functionality every month! And, most importantly, since everyone could see the functionality working, they could start brainstorming about the most important next steps to take. They could direct the team to build the next most valuable functionality during the next month's Sprint. Primavera and Scrum were on their way.

The best and fastest way to understand the kind of environment that Scrum produced at Primavera is to witness one of their Sprint Reviews. At Primavera, it looks and feels like a science fair. A large room is packed with all ten development teams clustered around demonstration areas, stating their goals and results achieved in the last 30 days. Joining in the discussions and providing feedback, spending 15 minutes with each team are the product owners (marketing people), the stakeholders (project sponsors), and company executives. Every 15 minutes, a loud horn (actually a bicycle horn) signals that it's time for them to move on to the next team. This is the empirical approach in full demonstration mode.

After one such review Joel Capperella, Product Manager at Primavera, said, "It's amazing how much gets done in a month – every month."

Teams Working Together

One of the most visible results of Scrum taking hold at Primavera is the disappearance of cubicles in favor of each team of developers setting up



workstations in a large room where everyone on each team faces each other and rapidly exchanges information with each other. The 90 developers at Primavera formed teams of about 10 people each. On the walls of each team area hangs the vision and mission statement of the team, along with Sprint schedules, and charts that show a series of complete releases, counting down to the final delivery date for the latest version.

Primavera had used a waterfall development process where analysts analyzed, coders coded, designers design, etc. However, under the guidance of Scrum they now employ cross-functional teams. Everyone on a team is responsible for doing whatever they can to build the best increment possible. If this means that the tester needs to help analysts write specifications for the coders, that's what happens. By using cross-functional teams, various aspects of the increment are inspected from all different angles as it is developed, increasing the overall quality.

After the First Release, What Was Still Missing

Once the Scrum fundamentals started to take hold, the development team realized that creating a solid release every 30 days requires different engineering practices. Primavera had been used to testing and fixing their code at the end of each release; But the short cycles of Scrum require continuous testing, continuous integration, and continuous improvement of the code. Since this type of quick release scheduling wasn't the norm at Primavera, eight months after introducing Scrum, Bob Schatz brought in a team from Object Mentor Inc. to implement Extreme Programming (XP) engineering practices and to help improve the engineering infrastructure.

Bob brought Object Mentor in because they wanted to maintain a high standard of software quality during every Sprint. Development in Scrum is very fast; so tight control and feedback are required to keep that speed from creating quality problems. The practices of XP provided Primavera with that control.

Under the rules of XP, a feature is not done until it passes both its unit tests and its acceptance tests. Programmers write the unit tests, and marketing and quality assurance write the acceptance tests. In both cases, the tests are written *before* the code is developed. In both cases the tests are *automated*. These tests become the detailed specification and the completion criteria for each Sprint and each feature. This type of co-operative work between testers, analysts, and coders is known as *test driven development*. It solved a long-standing problem posed by cross-functional teams: how can everyone on the team get involved from the very start. Once a functional test becomes a specification, it is easy for everyone to apply his or her expertise to get the Sprint started.

Another aspect of quality is sound object-oriented design. In Scrum, the drive towards features is so strong that close attention to good design principles is crucial. In the case of Primavera, Object Mentor made sure that the developers understood what good object-oriented design was about, what the principles were, what the patterns were, and how they could best be applied within the framework of an Agile method like Scrum. Object Mentor taught the team how to design solid systems, create clean code, clean up legacy code, and *keep* the design and code clean, for each Sprint.



Failing is Not an Option

Dick Faris, CTO, said, "Agile programming is very different and new. It is a different feel to the way programming happens. Instead of mindlessly cranking out code, the process is one of team dialogue, negotiation around priorities and time and talents. The entire company commits to a 30-day Sprint and delivery of finished, tested software. Maybe it is just one specific piece of functionality but it's the real thing, including **delivery and client review against needs and requirements.**" Faris continued, "Those needs and requirements, by the way, change. That is the strength we saw in the Scrum and XP Agile processes."

Faris also emphasized that the business of making software is difficult. "We can not afford to fail," he said. "We have to produce a product that people want to use. They can describe it 18 months ahead, but what they want could change half way through and that's where problems happen. The way we used to handle software development and project management was unacceptable. Primavera has 90 developers and it would be embarrassing for a project management company to be late with work and releases."

"Joel Koppelman and I started this company in 1983," Faris said. "Primavera was not the first to do project management. We were the first to bring it to a PC. Both Joel and I spend a lot of time with customers. Our ultimate goal is to rid the world of project failure and we bring this vision and our products to companies that don't practice project management."

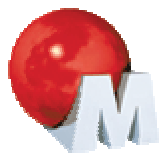
Both Koppelman and Faris participate in the Sprint Review sessions conducted every 30 days with all of the different project teams, all gathered in one area for a two hours. "We would hate to miss the Sprint Review session because that's where we get to see everything early on and we can always make a big impact," Koppelman said. "Over the years I have listened to customers, walked a million miles in their shoes and really listened for what the customers are thinking," he continued. "This process enables us to tell the developers first hand those customer thoughts and see it turn into reality within a couple of months."

Visible Signs of Change

Jennifer Coyle, Primavera Project Manager and Certified Scrum Master, emphasized, "Because they are working together, towards a common goal, and communicating constantly, our teams get their work done during normal business hours, go home happy and come back the next day, refreshed and eager to see each other and jump into the challenges and decisions they will get to make that day."

"What we like the most about Scrum are frequent inspections to say this is what customers want," said Coyle. She emphasized that it was not easy but went a lot faster than everyone imagined at first. Scrum's insistence on delivering complete increments of business value in 30-day learning cycles helped the teams learn rapidly. It forces teams to test and integrate experiments and encourages them to release them into production.

Bob Schatz spends a lot of time listening to his staff and watching out for their well-being. The impact of Agile development can be summed up by the reaction of one of



the test engineers coming to him a few months after they started; "I used to work a lot of nights and weekends. I'd go home and be so exhausted that I couldn't talk to my family. My kids didn't respond well to this, but I just kept doing it. Because you had the vision to do something different, my life has changed, Thank you so much for doing this...". Bob says "When you implement changes in an organization, you can only hope it has some positive effect. Then if someone comes to you with tears of joy and tells you that it's changed their life, well that's the grand prize."

"We pull in a lot of feedback from all of our customers and look for the similarities across conversations with resource managers, functional managers, program managers, and executives," said Michael Shomberg, Primavera Vice President of Marketing. "These methodologies are very empowering. Decisions are driven down to where knowledge is applied. Decisions are better and communication back to the customers is real and exciting. There are no over-promises or expectations that run the risk of disappointment, because the customer sees on the screen what they had in their head – or better. That's the wow we want to experience, with our customers and everyone in our company."

Today

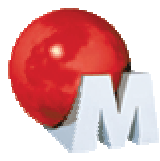
Nine months after Schatz, Faris, and Koppelman agreed to give Scrum a try, Primavera was able to produce a major release. The release was a knockout; it contained the functionality of two releases in one release, it was completed without anyone working overtime, and it was the result of a team environment in which marketing, product managers, developers, quality assurance, and executives worked together.

Starting on the next new release, Primavera has been working five months with Object Mentor and the developers are embracing test-driven development at both the unit and system level, and they are building a fully automated suite of tests. Quality assurance used to be at the back-end of the process. Now the test teams are at the front of the process writing the tests that the programmers make work. The teams are defining their Sprints in terms of automated tests rather than written requirements.

After embracing Agile methods, many at Primavera said, "We always loved working here and now we can say we love our work. We feel like we own the work rather than thinking this is work we are doing for the company." The team wears t-shirts that say it all...Develop with Heart; Deliver with Pride.

Primavera's IT Product

Primavera has found the approach so valuable that it has brought Agile Project Management into its core IT products. This helps Primavera's customers realize the same benefits that it realized, improves and tests the product by having Primavera drink its own champagne. It also significantly helps the Agile community because it provides a product for Project Management Organizations (PMO) and Senior Management managing a project portfolio of Agile and non-Agile projects, and tracking the progress of both in a similar fashion. With the combination of team collaboration spaces and enterprise information radiators, Agile is now able to live and grow in a traditional PM environment.



Primavera has been working with a number of their key customers that have already begun some projects using Agile development techniques. The Primavera team has become a valuable source for expertise and advice for others who may be wondering what to expect from a transition to Agile methods. Agile has gone mainstream.

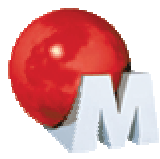
Why It's So Hard

It might seem from this Primavera's experience that everyone should follow in their footsteps. Just implement some Agile processes and solid releases will come out, no problem. Not only that, but everyone will enjoy working with each other, overtime will be eliminated, customers will be pleased, the competition will be vanquished, and bonuses will flow. Well, sorry to tell you, but there really is no Silver Bullet.

The process described in this story is a process of change. Change within the engineering organization, in the tools employed, practices followed, sociology of interaction, workspace, and ongoing collaboration in creating increments. Change between the engineering organization and marketing organization in how work is planned and changes are handled. Change in the way progress is perceived, as it is tracked by requirements rather than tasks. Change in the way deviance from schedules is perceived, since deviance isn't an enemy but something to plan around. Change in the reporting systems that track releases, and the way that all of the other organizations within Primavera synchronize their work with that of engineering. During one of the Sprint reviews, it was noted that it was about time that engineering replan every month; every other organization at Primavera already did. Looking at release plans more as forecasts allows people to consider adjustments due to various circumstances and make better decisions about moving forward. Bob likes to say "It's hard to move forward at 90 miles per hour when you're looking in the rear view mirror". You have to deal with where you are, not why you're not where you thought you'd be. Nobody can change history, and in software development, history is no real indication of what will happen in the future. So don't spend too much time worrying about it.

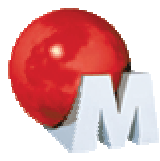
The hardest change was the backsliding. Letting others figure out how to work together rather than telling them how to work together is a major shift in management philosophy. This shift required Schatz to continuously observe and coach his staff to adopt the self-management approach. Everyone chipping in to do the best that can be done, with the team taking credit rather than individuals competing with each other takes a major shift in thinking. Everyone understanding and seeing that the product at the end of each Sprint is only as good as everyone's cooperative effort is also a humbling experience in a profession where people try to be heroes. All of these changes are gradual. And, just when they appear to have taken place, some stress occurs and everyone reverts to form. Regaining the progress each time takes time, much less noticing it.

Ken Schwaber and Bob Schatz were having lunch during one particularly hard day. They were both really discouraged because it seemed that the problems would never end. Then they realized that, indeed, the problems would never end. The core of Agile processes is unearthing problems so that they can be solved; and in a changing, complex development environment, these problems would only end when changes ended, and that was never. So, Bob looked at Ken and said, "Well, even



though it continues to be hard, we are making progress. It was worse than this before, we just never saw it.”

Making the decision to adopt Agile is a commitment to change the culture. Someone has to be willing to be the champion and take risks in order to improve the organization. It must start with a vision, a realization of the pain, and a willingness to question everything in your development process. In the end, we’re all just trying to find better ways to build software and create an industry that considers the well-being of the talented people in it.



The Main Characters of this Story

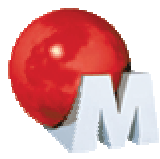
Bob Schatz is the Vice President of Engineering at Primavera. More than that, he is the most stubborn leader that anyone has ever seen. His determination to improve the life of his engineers while building better product has led to Primavera being a better place for everyone, not just the engineers, to work. Never settling for an easy answer, Bob is now getting a degree in Organizational Dynamics at the University of Pennsylvania. He knows that he did something really good. Now he wants to know why it worked.

Ken Schwaber created the Scrum process with Jeff Sutherland in the early 1990s to help organizations struggling with complex development projects. Their collaboration was based on friendship. Jeff's work with the initial ideas of Scrum and Ken's work with methodologies turned into several years of talking, collaborating and researching until they presented a paper describing Scrum at OOPSLA'96. Seven years later, which included more than 20 engagements to help software development companies change to Scrum, Ken and 16 others of like mind got together in Snowbird Utah in February 2001, discovered what they had in common and committed it to writing in the Agile Manifesto. <http://Agilemanifesto.org/> They founded the Agile Alliance, a nonprofit organization dedicated to the creation of Agile software, and serves as the chairman of the board. Email: ken.schwaber@verizon.net

Robert C. Martin has been a software professional since 1970. He is the founder of Object Mentor Inc., a firm of highly experienced software professionals that offers process improvement consulting, object-oriented software design consulting, training, and development services to major corporations around the world. In 1998 Object Mentor embraced, and helped to define, the Agile development method of [Extreme Programming](#). In 2000, Object Mentor took this to the next step by helping to form the [Agile Alliance](#), a group of professionals dedicated to the advocacy of lean and emergent software methods. Bob served as the first Chairman of the Agile Alliance. He called the Snowbird Meeting in 2001 and with Ken Schwaber and 15 other forward-thinking software developers and consultants, produced the Agile Manifesto. Email: unclebob@objectmentor.com

Primavera Systems, Inc. has specialized in project, portfolio and resource management solutions since Joel M. Koppelman, CEO, and Dick Faris, CTO founded it in 1983. Primavera product names reflect the types of projects that its customers manage: information technology, engineering, construction, maintenance and turnaround, new product development, and professional services. www.primavera.com A growing and successful software company, Primavera has not experienced layoffs and recently expanded into new offices in Bala Cynwyd, PA. Many contributed to this article with interviews and by inviting the writers into their offices, their team rooms, their daily work spaces and conversations. Especially useful and focused comments came from the following people at Primavera: Joel Koppelman, CEO; Dick Faris, CTO; Bob Schatz, Vice President-Development; Michael Shomberg, Vice President-Marketing; Jennifer Coyle, Project Manager and Scrum Master; and Joel Capperella, Product Manager.

If you have specific questions about the topics covered in this white paper or if you would like general information about Scrum project management or Extreme



Programming and Agile processes for software development, please contact Object Mentor at 847-775-8172 or unclebob@objectmentor.com or Ken.schwaber@verizon.net. If you want to talk to the man who took the lumps and made it happen, contact Bob Schatz at Bschatz@primavera.com. Just don't take too much of his time; he's busy doing the next great thing.

Resources

Ken Schwaber, *Agile Project Management with Scrum*, Microsoft Press, 2004. This high-level reference describes how to use Scrum to manage complex technology projects in detail, combining expert insights with examples and case studies based on Scrum. Emphasizing practice over theory, this book explores every aspect of using Scrum, focusing on driving projects for maximum return on investment.

Robert C. Martin, *Agile Software Development: Principles, Patterns, and Practices* Prentice Hall, 2002. This book is for the software developer in the trenches. It describes Agile Software Development from a practitioner's point of view. Through dozens of case studies it describes and teaches the principles of object oriented software design, the patterns that embody those principles, and the practices that enable individuals and teams to use those principles.

Ken Schwaber maintains a website about Scrum at www.controlchaos.com

The **Agile Alliance**, a non-profit organization that supports individuals and organizations that use [Agile approaches](#) to develop software. Driven by the simple priorities articulated in the [Manifesto for Agile Software Development](#), Agile development approaches deliver value to organizations and end users faster and with higher quality. <http://www.Agilealliance.org/home>

© July 2004, Object Mentor, Inc. and Advanced Development Methods. This white paper is for informational purposes only. The information contained in this document represents the current view of Robert C. Martin, Object Mentor, Inc. and Ken Schwaber, Advanced Development Methods on the items discussed as of the date of this publication. Object Mentor, Inc. and Advanced Development Methods reserves the right to make improvements in this document and the services it describes at any time, without notice or obligation.