Agile In The Bathtub

Developing and producing bathtubs the agile way

Gaetano Mazzanti
@mgaewsj
Gama-Tech
Fano, Italy
gaetano.mazzanti@gmail.com

***Abstract*— This is a report about the introduction of Agile/Lean methodologies at an Italian company producing luxury hydromassage bathtubs and showers. Initially focused on product development and portfolio management, the Agile/Lean transition has been extended to other departments of the company. This has been a significant experience in combining Agile and Lean in a non-software context.**

Keywords: agile, lean, Kanban, product development, portfolio management, coaching.

I. Introduction

Agile methodologies [1] originated in the 90s in the software development domain to address the shortcomings and poor results of traditional methodologies. During the following decade the integration of Agile and Lean values and principles gave birth to new approaches like Lean Software Development [2] and Kanban [3].

While Lean has already been used (and misused) in various contexts other than software, Agile has been applied mainly to software development.

A growing number of companies are reportedly applying Agile practices in non-software contexts. This is happening in commercial and government sectors. Agile is not limited to product development but it is spreading to services, maintenance, marketing, sales, etc. Unfortunately the focus is often on practices and on the mechanics, ignoring the values and principles that drive those practices.

This paper reports on a coaching experience introducing Agile ***and*** Lean approaches at a company producing luxury hydromassage bathtubs and showers. The specific focus of this paper will be on the benefits and challenges in combining Agile and Lean in a non-software context and on how coaching product engineers differs from coaching software developers.

II. Context

In 2009, Teuco Guzzini, a traditional top-down organization using a phase-gate process with rigidly defined procedures, went through a downsizing and a redefinition of their business model. They moved from 100% internal design and production processes to extensive collaboration with external partners. They radically changed their process for devising new products and selecting which ones should go to the market. The number of engineers was reduced and they had to switch from "just" designing and building products to also coordinating and managing external partners.

In May 2011 the company asked for help. They did not ask for any specific prepackaged Lean or Agile 'method' but just for a solution to their problems. The outstanding issues were typical: too long a time to market and low quality. Most of the blame for this was attributed to engineers being untrustworthy and ineffective, especially in managing external partners. The author's role in this experience has been that of an external agile coach. I operated alone for the most part, but received some valuable help from another coach during the training and active learning phases. Previously, I spent many years as a manager and executive in the software and machinery industry where I had the opportunity to drive and to participate in the introduction of Agile and Lean methodologies into software development and manufacturing. As a coach I had previously been involved in large-scale agile transitions and in helping small and medium sized organizations to embrace Agile. What follows is a recounting of my first experience as a coach in applying Agile and Lean in a non-software context.

III. Assessment

The first step of this endeavor was an assessment to provide to the company an external objective evaluation of its current process and to verify any mismatch between perceived and real problems. Based on previous experience, I chose to complement the usual interviews (with top executives, middle managers and engineers) with other activities like Draw The Process (with engineers) and a Premortem Retrospective (with managers). The reason for separating managers from engineers in the last two activities is to catch major differences in the perception of problems.

978-0-7695-4804-3/12 $26.00 © 2012 IEEE
DOI 10.1109/Agile.2012.27

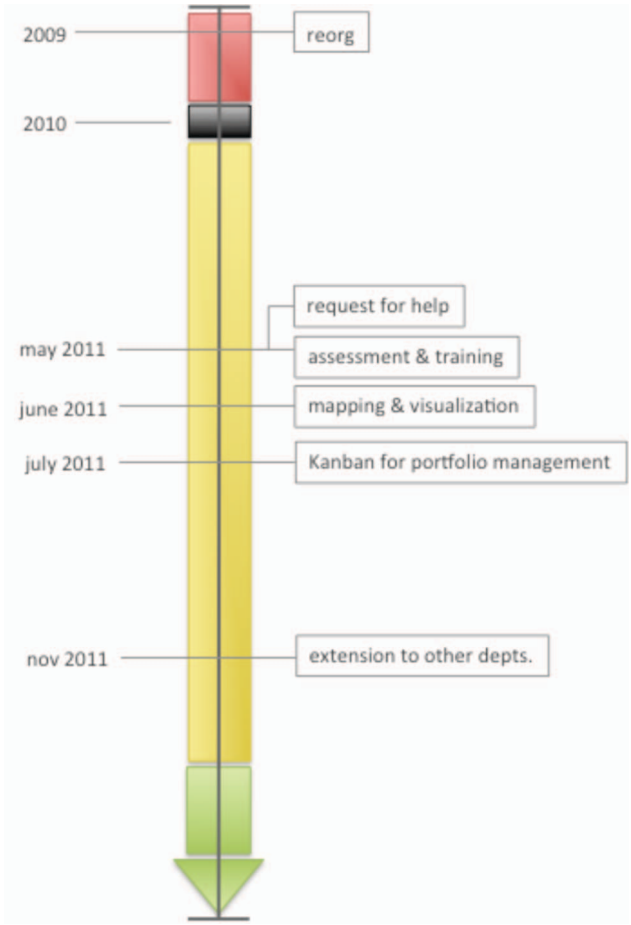

Fig.1 timeline of the transition

A. Interviews

Interviews were conducted with representatives from most departments with different levels of responsibility. The goal was to get an understanding of the company process and to identify major dysfunctions and impediments. The outcome was a variegated picture whose key elements were:

- A general lack of transparency and effective communication. No shared vision.
- A focus on local efficiency, with each department having its own independent goals.
- Moving targets and constantly changing requirements and priorities. Too many projects, unpredictable timing.
- Concept phase for new products (too) long and producing incomplete or vague output.
- Overburdened and somewhat demotivated engineers.
- Corporate process and procedures, considered too complex and cumbersome, were often ignored or cheated. Note: they were defined before the downsizing and never updated.
- Engineers had no specific skills in managing projects and external partners.
- No interaction between engineers and customers. No structured approach to learn from and about customers.

B. Draw the Process

Draw The Process is an activity in which groups of 6-8 people are asked to draw the process that brings a product from concept to shipment and then identify the most critical and problematic steps. The drawing can be very schematic, just a workflow, or more creative. The goal of this activity is to check if there is a shared understanding of the process and to get a first feeling of where the major issues are.

I had successfully used this activity previously. In this situation the process turned out to be much more complex than the typical software process.

About 30 engineers (no managers) formed 4 groups and produced results that were similar, even if with different artistic and ironic content ☺.

People were then asked to identify the most critical steps through dot-voting. Each person had two colored votes, a red one for the most important issue and a blue vote for the second most important one. A more complete and schematic drawing was used for this activity as shown in the figure below.

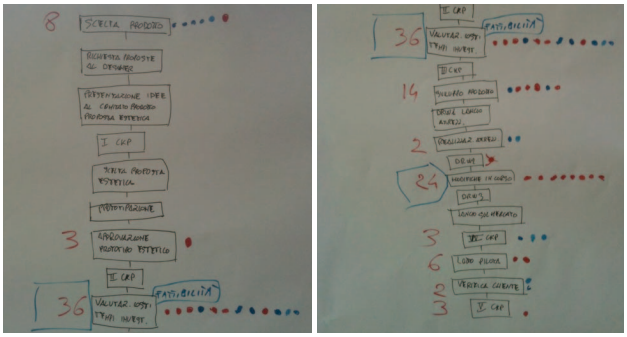

Fig.2 dot-voting most critical process steps

The most critical steps were identified as:

- Feasibility (including determining costs and schedule)
- Late Product Modifications as a result of continuous specification changes
- Product Concept phase too long and not providing clear guidance

The last two items were consistent with the outcome of earlier interviews.

C. Premortem Retrospective

In this type of retrospective, described in [5], the idea is to fast-forward ahead in time, typically 6-12 months, and presume that a project has miserably failed. Key events that 'happened' during the project are positioned along a timeline. Events are then grouped and discussed and finally dot-voted to identify the most important ones.

I had previously used this activity many times. It typically provides valuable and sometimes unexpected

inputs and it often fosters interesting discussions or even arguments among participants.

About ten executives and key middle managers were gathered to hold the premortem retrospective for a 'failed' 12 months project.

The key facts that emerged from this activity were:

- Overburden, multitasking and continuous changes in priorities are the norm (sometimes projects are even 'forgotten' in a dormant state).
- Specs change often, even late in the project.
- Delays in approvals, especially in the initial phase (Concept), and optimistic planning lead to schedule slip -> people urged to stay late, etc.
- Production costs are often higher than expected (e.g. esthetic desires of designers are too difficult to fulfill. This is often discovered too late).

It is worth to note that while overburden is the first item in the list it was not perceived by executives as 'a' problem. Instead they believed engineers were not eager or good enough to perform at 120% capacity.

In the end, executives were really impressed by the effectiveness of this activity. In less then 90 minutes we exposed and shared most of the company issues (from their point of view).

D. Summing it up

The assessment phase took 3 days.

As expected, most of the key learning about dysfunctions emerged during interviews and the premortem, while Draw the Process was more helpful in understanding the real process followed in the company.

Results of the assessment were presented as a slide-based presentation to top executives.

Many of the issues where somewhat known to them beforehand, however some were dismissed irrelevant and some were simply not addressed for the time being. However, management was caught completely off-guard by the lack of transparency and communication reported by many interviewees and also by their demotivation. They were confident these were not issues at all.

IV. Beginning the Transition

The assessment showed that the Concept Phase was longer than the Product Development Phase: the former taking on average 60% of a project duration (equivalent to 9-18 months) while the latter taking the remaining 40% (6-12 months).

The company chose to focus on development, as other initiatives were already in place to shorten the concept phase.

Product Development (PD) in Teuco directly involves four departments: Engineering, Operations, Quality Control, and Test Lab. All of them were involved in active learning about Agile and Lean values and principles. Aspects like the importance of queues and WIP limits or why multitasking should be avoided and how they affect cycle times were introduced through specific activities. This caused some initial misunderstandings and annoyance by people used to being told just what to do or to apply techniques by the book. Specific focus was also put on the importance of interactions, communication and self-organization, concepts that were a bit alien to them.

Engineers involved in managing 'projects' and external suppliers went through specific learning activities focused on the differences between traditional and agile approaches and about handling interactions, conflicts, etc. This phase lasted one week.

The results of the assessment were then shared and discussed with engineers. I naively assumed that after the learning phase they would have comfortably applied the new concepts. Instead it was not easy for them to start dropping old habits e.g. the idea that multitasking was a natural and effective approach to address overload. So we spent additional time on learning activities to better understand demand analysis, WIP, queues, etc. I discussed with the team the need to find a way to 'prove' in an objective way how all these factors were limiting and compromising engineers' capability to perform at their best. And that we also needed to show that this was not simply a Product Development issue but that the whole organization was involved.

Engineers had mixed feelings at this point: they were equally divided between enthusiasm for the possible upcoming (r)evolution and skepticism about the chances to really change anything. This phase took an additional week.

V. Agile and Lean For Product Development

At this point we had a number of goals in front of us:

- Map the real process and visualize all ongoing activities.
- Expose and possibly measure dysfunctions like overburden, bottlenecks, task switching, etc.
- Analyze data and decide what to do (the first in a series of inspect and adapt loops).

We needed quick wins to keep the spirit of the team high and keep executives confident. Running quick experiments, inspecting and adapting is exactly at the core of Agile so that is where we started.

A. Mapping & Visualization

The process had already been roughly mapped in the Draw the Process activity during the assessment, so it was quite straightforward to define it in more detail. Its basic structure has 5 major checkpoints and 3 design reviews:

Concept			Development/ Prototyping			Production	
CKP I	CKP II	CKP III				CKP IV	CKP V
			DRW I	DRW II	DRW III		

Obviously many intermediate activities take place between checkpoints and between design reviews.

Activities up to CKP III include: initial concept, product roadmap, briefing document, first design proposal, CKP I, design mockups, design selection, CKP II, final

design choices, feasibility analysis, costs and schedule, and suppliers selection.

This meant that at CKP III product features, cost and market shipment date have already been set. There are still many unknowns and uncertainty, yet we end up in a fixed price, fixed schedule, fixed scope kind of context. This could be a perfect recipe for failure. However as we have seen above, later changes in requirements are the norm even if no corresponding change in schedule is formally allowed.

As explained in the previous section, our initial focus was on activities following the third checkpoint (CKP III). This is when the real product development efforts start.

Activities after CKP III include: codesign with suppliers, purchasing plan, quality control plan, mold design, DRW 1, mold production, fixtures and gauges production, first prototype, DRW 2, production drawings and BOM, product validation, packaging, final prototype, DRW 3, PLM data loading, CKP IV, materials purchasing, assembly, testing, customer test, CKP V, final modifications, and shipment.

We performed demand analysis and work type analysis. Work can be basically categorized in three project sizes: we creatively labeled them Small, Medium and Large. Small projects last just a few months (from concept to shipment) and involve either simple modifications of previous products or simple products like shower trays. Medium projects are the most common and typically take 9-12 months. Large projects usually involve complete product ranges and can take up to 18-24 months.

The average delay in completion time ranged from 2 months for Small Projects to 6-9 months for Large ones corresponding to a 33%-50% delay.

Each project leader is in charge of multiple projects. This makes sense as projects are typically in different states none of which requires full-time commitment.

Demand is based on a combination of recurring events (e.g. exhibitions), sales opportunities and new product concepts. Sales opportunities are difficult to predict and they typically take priority over any other project. Exhibitions' deadlines are, surprisingly, often ignored and this creates last minute (month-long) rushes and extra-efforts to complete or even create a product.

We agreed that the right thing to do at this time was to create a board where each column represented the states from CKP III to final shipment and where each card represented a project. We initially mapped card colors to project size.

B. Exposing dysfunctions

The figure below shows how the board looked after our first attempt to populate it.

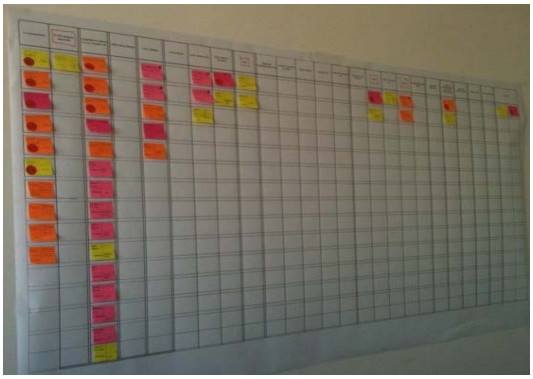

Fig.3 initial mapping

Overcrowding in the first and third column is obvious. Items, that is projects, were piling up either because of delays (waiting for CKP III approval) or because of overloading (in the codesign/purchasing plan phase). This was to be expected, based on the results of the assessment. However, as I had observed in previous contexts, the simple fact of exposing problems on the board triggered immediate action by the management: projects in those two columns were reviewed in terms of business priority, risk and complexity. Some projects were stopped; others were accelerated and moved quickly to CKP III.

We used this board for three weeks just to understand which kind of information we could get from it. We started tracking time information to collect cycle time data.

At this point I introduced some typical Agile ceremonies: we had daily standups in front of the board and we held weekly retrospectives. Initially, engineers found these activities useless and boring. Standups were too long and repetitious as they were used just to report status to the manager. So we changed approach and started focusing only on highlighting problems. It worked, standups became shorter and a number of hidden or 'forgotten' issues were exposed (i.e. a delivery from a supplier being suddenly delayed by another unknowingly conflicting request).

I found a major difference here between product engineers and software developers, at least at this company. Product Engineers were feeling and acting as individuals. They didn't feel they were part of a team. They had their own independent goals and were focused just on them (see discussion about standups above). Standups helped a lot in providing a team-forming glue.

The same happened with retrospectives. They proved effective in discussing and improving the process (i.e. we identified and acted to remove an old corporate procedure, previously taken as a given, that blocked projects for one or two weeks just because four different people had to sign a document) and as a gluing factor for the team.

After the third retrospective we decided to introduce a number of major changes to address some of the key shortcomings exposed by/in the board:

- Overloading was visualized but not prevented.
- Priorities were still 'suggested' by management with no objective selection criteria.
- Project ownership changes during the project life were not visible on the board.

Needless to say the quick effects obtained through visualization and the involvement in standups and retrospectives had an immediate effect on people engagement and morale.

C. *Kanban for Portfolio Management*

Up to this point we did not really use a Kanban system. We were just using a board to visualize what was happening and to expose dysfunctions.

To address the shortcomings listed in previous section we needed to step up and define policies to optimize flow.

It has been quite natural to do this at the portfolio level.

Some of our interconnected goals were:

- Balance demand and capacity
- Define criteria to assign value and priorities to projects based on business importance, cost of delay, due dates, etc.
- Use a pull approach to select and move forward projects
- Identify relevant data to represent on cards and to be used when defining further policies
- Identify appropriate criteria to define and then set WIP limits
- Define policies to rule transitions from one state to another
- Measure cycle time (from CKP III approval to shipment)

After a couple of lively meetings we came out with this board:

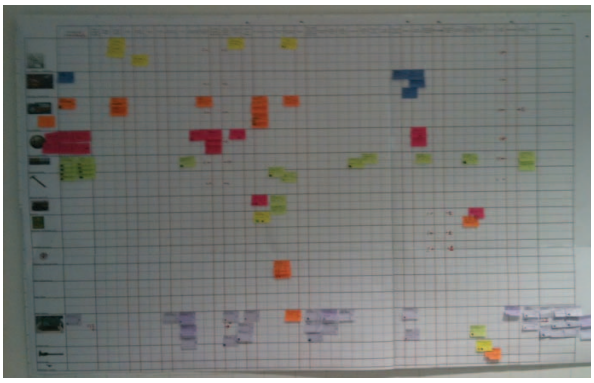

Fig.4 the portfolio board

We introduced specific swim lanes for each project leader and then additional lanes for mold design and production, for final production, and for TestLab activities.

Each project moves from left to right but also along the vertical axis based on the ownership at a given phase. For instance during mold design a project is positioned in the corresponding swim lane. Resources for mold production are scarce so this helps in identifying overloading.

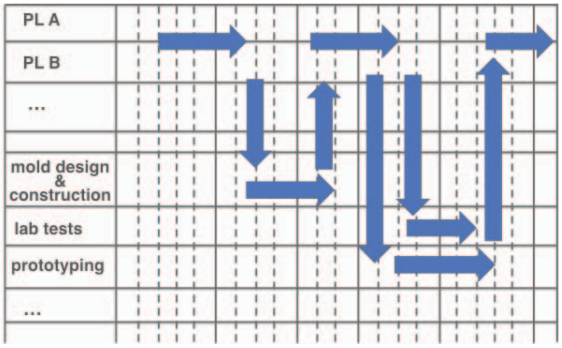

Fig.5 horizontal & vertical flow

As color coding based on project size was not adding any value we decided to use the card color to identify the project leader.

Each card contains the project ID, a description, the due date (if any), the associated cost of delay (not always available) and a number of symbols that show if and which kind of issues or impediments are slowing down the project.

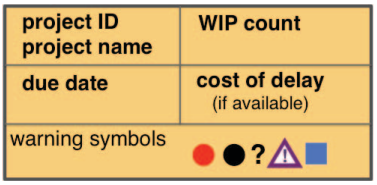

Fig.6 card contents

A special blue square symbol indicates a project that has been stopped due to reached WIP limits. The rationale for this is explained below.

As explained above, each project leader is in charge of multiple projects at a given time. Project characteristics include size, but also cardinality (i.e. how many shower trays for a given product range) and other parameters like machining or forming complexity. We initially assigned WIP = 1 to each project but we soon realized that this was not very meaningful. So we started to weigh how much each project was adding to WIP based on the aforementioned characteristics. Fine-tuning this required a couple of attempts (weeks) but we finally reached consensus. The range of WIP induced by each project is between 0,2 and 2. Project leaders use a planning poker approach whenever a new project passes CKPIII to assign this value.

Each project leader has his own WIP limit that cannot be exceeded. Additional WIP limits have been set for resources (mostly hardware and used also for standard production) like mold design and production, test lab and final prototyping. Some of these limits are not constant but they vary in time as they have to look ahead to take into account standard production needs and to anticipate the load induced by projects in progress.

Whenever WIP limits are reached, a project is tagged with a blue square symbol and it can either remain in a specific buffer or roll back to the 'passed CKP III' state.

When a project leader is available he pulls a project based on available WIP 'space' and cost of delay. This choice is automatic and does not require any meeting or manager involvement.

This is the beauty of Kanban. Visualizing what is going on and exposing dysfunctions makes it much easier to collaboratively define and agree on what to do. Defining policies and optimizing flow is the result of a systemic approach followed by all stakeholders. There is no top-down enforcing of rules and procedures. People can have a significant role in designing their process. This has a tremendous effect on people engagement and motivation.

I also believe that a key ingredient of this transition has been keeping a constant cadence for daily standups and weekly retrospectives to continuously challenge the status quo.

The focus on flow optimization led us to improve or remove some old corporate procedures that were either too complicated or not necessary anymore (i.e. a procedure generating a one week delay just to fill parameters in the PLM system).

Another outcome of retrospectives has been the introduction of design reviews. The main reason for this was a lack of homogeneity of CAD models and drawings in terms of 'style', tolerances, reused components, etc. Reviews are helping in defining a common way to create CAD models and also useful in verifying and validating external partners' work.

At one point we tried to use multiple boards, one for engineers, one for Operations, one for TestLab, etc. and to move stickies around from one to the other. The idea was to replace a very complex board with many simpler ones. It did not work. Reality is complex and people wanted to see it clearly. We switched back to a single board.

In terms of measurements we have chosen to focus just on a few metrics. One is cycle time with some of its components: time spent between various checkpoints and between reviews, differentiating between active time and time spent waiting for 'something' to happen (i.e. an unexpected technical issue, an unavailable machining center, a formal approval, etc.). We are also measuring cumulated WIP by project leader and by project phase to verify if current criteria to assign WIP to projects is reasonable and how WIP variations correlate to (perceived) overburden and to cycle times.

VI. Extending Agile to Other Departments

Following the positive results in Product Development, other departments of the company asked to try a similar approach. The initial requests were focused more on learning specific techniques and practices than values and principles. After agreeing once again that it was not just a matter of tools, top management gave us a chance to train and coach the people involved.

We started in marketing where we successfully implemented an iteration-based approach to the preparation of a product launch at an exhibition (four weekly iterations).

We then moved to sales where we started using the Kanban method to visualize and optimize the flow of sales activities.

Retrospectives have been again a key factor in making this work. We generally followed the five steps approach described in [4]. Activities we used most were (a) Starfish for gathering data and (b) Five Whys to generate insights through root cause analysis.

It's still a bit early to conclude that these new approaches will stick in marketing and sales. They have been surprisingly well received and initial efforts have led to very good results.

Conclusions

A. Achievements

Teuco is still going through its transition to Agile. Actually a transition never ends: the search for continuous improvement should never stop.

After almost one year we have seen major changes and improvements.

The most evident improvement is the significantly reduced overburden of engineers in Product Development and Operations. This has been achieved through demand management and capacity leveling: assigning priorities based on due date and cost of delay and adjusting activities and work to available capacity.

This in turn has led to better predictability for Small and Medium projects both in term of schedule and quality. It's too early for significant data for Large projects but we are confident that the same pattern will apply: cycle times, from CKP III to shipment have dropped consistently on large projects. Our preliminary data show a projected 30% average reduction. This means going to market 2-4 months earlier.

Engineers are more engaged and motivated. We have being tracking their satisfaction using a niko-niko calendar [6]. The number of 'bad days' has dropped from ~40% to ~15%

B. Lessons Learned

This experience presented a number of coaching challenges for the author: applying Agile in a non-software context, combining Agile and Lean, coaching product engineers instead of software developers.

I have been previously involved in Lean adoptions and I had a chance to see that they were focused mostly on the mechanics and practices and they typically failed to embrace Lean values. 'Respect for people' was an ingredient typically missing in those attempts. And this is where Agile played a key role in this endeavor. Ceremonies like standups, reviews and retrospectives induced engagement, commitment, transparency, self-organization, trust, and empowerment in the teams. In the end this has been how 'respect for people' has become a reality and not just hot air.

The non-software aspect of Teuco's products has not been a major issue. I felt initially a bit lost as I could not count on practices like TDD, refactoring, continuous

integration, etc. This however forced me to focus on the flow of activities and on how these activities were producing value for the company's customers, another good match between Agile and Lean as they both focus on value creation. Design reviews turned out to be similar to their counterpart in a software context.

Dealing with product engineers presented some unexpected challenges. They tend to be a group of individuals rather than a team, so I spent significant effort addressing this issue. However, compared to software developers, engineers seem to have a better understanding of the whole product cycle, the impact of changes and mistakes, and the steps required to get a product into the market. On the other hand, I didn't see any particular interest in technical problems, solutions or tools that is typical of software developers. Also, product engineers seemed more willing to wait for the coach to provide solutions. This could be related to how demotivated they were when I arrived. Anyway multiple attempts were required to help them be proactive and act more as a team.

C. *Open Challenges*

A few words about some of the remaining challenges.

There is a natural inclination to rest on laurels. We have seen such a tendency immediately after major achievements (i.e. after setting portfolio WIP limits and reducing the number of projects in process). Retrospectives are a terrific tool to constantly keep questioning the status quo. They should be used constantly.

Lead Time, from Concept to Cash, has improved but it is still too long. Previous attempts to shorten the Concept Phase have failed so we are currently discussing applying the approach described here to the Concept Phase as well.

.

D. *About Waste*

A final note: the word "waste" has never been mentioned in this paper (up to now) and has been rarely pronounced in Teuco. This has been a deliberate choice by me. "Waste" is such an abused and overused term that I consciously avoided using it. Nevertheless we did remove a lot of it.

ACKNOWLEDGMENT

I would like to thank Roberto Onori, Teuco's R&D Director, for his constant help, support and commitment during this endeavor, Mauro Guzzini, Teuco's CEO, for being patient enough to let this effort unroll and bring concrete results and Fabio Fraternale, former Managing Director for initiating all of this.

I would also like to thank Rebecca Wirfs-Brock for being my shepherd in the process of creating this paper. Her insight and hints have been invaluable.

REFERENCES

[1] Agile Manifesto, http://agilemanifesto.org

[2] Mary and Tom Poppendieck, "Lean Software Development", Addison Wesley

[3] David J Anderson, "Kanban", Blue Hole Press

[4] Esther Derby and Diana Larsen, "Agile Retrospectives: Making Good Teams Great", Pragmatic Bookshelf

Article in a blog:

[5] G. Mazzanti, "Premortem retrospectives," http://agileandbeyond.blogspot.it/2011/08/premortem-retrospectives.html

[6] http://guide.agilealliance.org/guide/nikoniko.html