

How to Achieve Lean by Creating Communication Flow and Keep the Capabilities from Lean Startup

Kristoffer Berglind
Faculty of Engineering at Lund University



LUND UNIVERSITY

Abstract

Lean principles are spreading from the manufacturing industry to public sectors and the service industry. Many attempts at implementing lean have failed. The aim of this thesis is to find methods to implement lean in the service industry. Analysis of the correlation between principles in lean production proved that it can be hard to implement lean principles one by one to achieve a lean company. Also, the company's culture is examined as a factor that can inhibit lean principles. Reviewing experiments of implementing lean tools at Telavox, it was apparent that some fundamental principles, named meta principles, were needed to implement lean. A model, with two parts, was created based on what was learned from the experiments. The first is a division named the sandbox which functions by achieving or maintaining the capabilities of a startup company to achieve a higher degree of internal disruptive innovation. The other is a communication tool called communication flow kanban which is designed to support lean principles. It is stated that if a company uses the communication tool to make decisions and as a framework for communication, the company will be more lean. The communication flow kanban also reduces wastes and fosters a higher degree of incremental innovation.

Preface

This master thesis was written for the institution of *Production and Materials Engineering at the Faculty of Engineering (LTH) at Lund university*. The report constitutes 30 out of 300 total ECTS for the entire degree, and is the final assignment for the master of science in Mechanical Engineering.

The master thesis was written for Telavox, where I was also employed as a supporting manager. This gave me a unique perspective, as I could implement lean principles and give good insight into why the result ended up as it did. The thesis was written over a 11-month period compared to the standard 6-months, which was also valuable as I could evaluate the results over a longer period of time.

I would like to thank the following people:

Professor Jan-Eric-Sthåhl - For his dedication for his students and motivational courses that give knowledges that are applicable in work life from day one.

Daniel Johansson - For the help with coaching when I got stuck.

All the managers and employees at Telavox - for having trust and letting me implement my theories, for the support and for letting me be flexible combining work and studies.

Table of Contents

Abstract	I
Preface	II
Table of Contents	III
1 Introduction	1
2 Structure	2
3 Lean Startup - Creating the Foundation	3
3.1 To Work Effective - Experiment	4
3.2 Build-Measure-Learn Feedback Loop	4
3.2.1 It starts with Assumptions.....	5
3.2.2 Build - Minimum Viable Product	6
3.2.3 Measure.....	6
3.3 Learn and Pivot.....	8
3.4 Growth.....	10
3.4.1 Sticky Engine of Growth.....	10
3.4.2 The Viral Engine of Growth	11
3.4.3 The Paid Engine of Growth.....	12
3.4.4 Adjust the Growth.....	12
3.4.5 Worse Before Better.....	12
4 Lean production	13
4.1 Philosophy	14
4.1.1 Principle 1 - Long-Term Thinking	14
4.2 Processes	15
4.2.1 Principle 2 - One-piece Flow.....	15
4.2.2 Principle 3 - Pull Instead of Push.....	16
4.2.3 Principle 4 - Level out the Workload (Heijunka).....	17
4.2.4 Principle 5 - Build in Quality (Jidoka).....	18
4.2.5 Principle 6 - Standardize.....	21
4.2.6 Principle 7 - Visual Control	23
4.2.7 Principle 8 - Use Reliable, Tested Technology	24
4.3 People and partners	25
4.3.1 Principle 9 - Grow Leaders who Thoroughly Understand.....	25
4.3.2 Principle 10 - Respect, Develop and Challenge Your People and Teams	26
4.3.3 Principle 11 - Respect, Challenge and Help Your Suppliers	28
4.3.4 Principle 12 - Go and See for Yourself.....	28
4.4 Problem Solving.....	29
4.4.1 Principle - 13 Make Decisions Slowly by Consensus.....	29
4.4.2 Principle - 14 Continual Organizational Learning through Kaizen.....	30
5 Lean Software Development	33
5.1 Eliminating Waste and Differences Between Production and Software Development	33
5.1.1 To Find Waste by Value Stream Mapping.....	36
5.2 Amplify Learning	36
5.2.1 Quality Differences Between Production and Development	36
5.2.2 Feedback	37
5.2.3 Iterations.....	38
5.3 Making Decisions as Late as Possible.	39
5.4 Deliver as Fast as Possible.....	39

5.5	Empower the Team	41
5.6	Build Integrity In.....	42
5.7	See The Whole.....	43
6	Areas of Research and Methodology	45
6.1	Calculating Waste.....	45
6.1.1	Time Wasted on Outgoing Phone Calls.....	46
6.1.2	Average Waiting Time for Caller.....	47
6.2	Support Decisions with Lean Tools - 5 Why	48
6.3	Continual Improvements with Stand-up Meetings	48
6.4	Why Developing Lean at a Service Organisation Might Fail .	49
6.5	Conclusion - How to Developing Lean at a Service	
	Organisation.....	49
7	Results	51
7.1	Calculating Waste Time	51
7.1.1	Wasting Time on Data	51
7.2	Support Decisions with Lean Tools - 5 Why	53
7.3	Continual Improvements with Stand-up Meetings	53
7.4	Why Developing Lean at a Service Organisation Might Fail .	54
7.4.1	How Lean Production Principles are Correlated.....	54
7.4.2	Culture	57
8	How to Achieve Lean by Creating Communication Flow	
	and Keep the Capabilities from Lean Startup	58
8.1	Meta principles - How to Develop Lean Principles in a Service	
	Organisation.....	59
8.1.1	Consistency	59
8.1.2	Pull Instead of Push - Empowering	59
8.1.3	Spare Time for Change and Make Changes at the Right Time.....	60
8.1.4	One Change at the Time.....	60
8.2	The Model.....	60
8.2.1	Internal Disruptive Innovation Versus Incremental Innovation.....	61
8.2.2	Telavox Startup Culture to stimulate disruptive Innovation	64
8.2.3	Startup Culture in a Larger Enterprise Company	65
8.3	Disruptive Innovation - The Sandbox	66
8.4	Lean startup at Telavox	67
8.4.1	Build-measure-feedback loop at Telavox	67
8.4.2	Pivots at Telavox.....	68
8.4.3	Engine of Growth at Telavox.....	68
8.5	Capabilities Needed to Create a Sandbox Organisation	69
8.5.1	Scarce Resources	69
8.5.2	The Culture in the Sandbox	70
8.5.3	The People in the Sandbox.....	70
8.5.4	The Sandbox's Organisation	71
8.6	Incremental Innovation.....	72
8.6.1	Growth.....	72
8.7	Communication waste experienced by Advisors at Telavox...	74
8.7.1	Decisions Without Reaching Consensus.....	74
8.7.2	Single Minded Decisions.....	74
8.7.3	Waste Relocation, Suboptimization	75
8.7.4	Lost in the Noise.....	75
8.7.5	Unnecessary Information.....	75

8.7.6	Information in the Wrong Hands and Misinterpreted Information.....	76
8.8	Waste at Telavox with Lean Software Development Principles	76
8.8.1	Extra Features	76
8.8.2	Task Switching.....	76
8.8.3	Extra Processes	76
8.8.4	Partially Done Work.....	77
8.8.5	Unnecessary Movements	77
8.8.6	Defects.....	77
8.8.7	Waiting	78
8.8.8	Unused Employee Creativity	78
8.9	The Communication Flow Kanban.....	78
8.9.1	Describe a Broad Problem in an Epic.....	79
8.9.2	Supporting Lean Principles - Epic	82
8.9.3	Reducing Waste - Epic	85
8.9.4	Suggested Solutions in the Epic.....	85
8.9.5	Supporting Lean Principles – Suggested Solution in the Epic	86
8.9.6	Reducing waste – Suggested Solution in the Epic	87
8.9.7	Reaching a Consensus - Horizontal Decision-Making	87
8.9.8	Supporting Lean principles - Reaching a Consensus.....	88
8.9.9	Reducing Waste - Reaching a Consensus.....	89
8.9.10	Making Priorities	89
8.9.11	Supporting Lean principles - Making Priorities	90
8.9.12	Decision-making	90
8.9.13	Supporting Lean Principles - Decision-making	91
8.9.14	Reducing Waste - Decision-making.....	93
8.9.15	In Progress - Start Developing.....	93
8.9.16	Supporting Lean principles - In Progress.....	94
8.9.17	Reducing Waste - In Progress.....	94
8.9.18	Testing and Education.....	95
8.9.19	Supporting Lean principles - Testing and Education.....	95
8.9.20	Reducing Waste - Testing and Education	95
8.9.21	Standardize	96
8.9.22	Supporting Lean Principles - Standardize.....	96
8.9.23	Done	97
8.10	Other Benefits From the Communication Flow Kanban.....	97
8.11	Evaluation of the Communication Flow Kanban	97
9	Discussion and Conclusion	99
10	Bibliography	102
Appendix		105
10.1	Questions for interview with Filip Olsson.....	105
10.2	Questions for interview with Team Leaders	105
10.2.1	Intervjue 5-why question.....	105
10.2.2	Visualization of organisational development processes	105
10.2.3	Creating Communication Flow	106
10.2.4	Organisational change.....	106
10.3	Calculating Waste – Data.....	107
10.3.1	Overall Results and Data for Delivery.....	107
10.3.2	Data for Team Customer Care	108
10.3.3	Data for Economy.....	109

1 Introduction

Many companies fail to implement lean. A survey conducted by industry week in 2007 found that 74% of the companies trying to implement lean in the manufacturing industry admitted that they didn't succeed in making good progress with lean. (Pay, 2008) The purpose of this master's thesis is to find and review new models in implementing lean in the service industry. Since the service industry differs in many ways from the production industry, the model will take inspiration from a wider range of lean principles that originate from lean production. The original plan was to create a step by step handbook to implement lean. This was later discarded as it proved to be very difficult. Two areas are instead examined. First, lean startup and why a company in growth should maintain the capabilities from the startup years. Second, a communication tool that supports lean principles, reduces wastes and increase incremental innovation.

Malmö, 2016-11-18

2 Structure

This master's thesis starts with a large section of theories. For readers that are well orientated with lean and lean principles this part might feel superfluous. However, one of the problems with implementing lean is that the principles are not understood well enough, therefore this section is necessary for they who are not well oriented with all three schools of lean principles presented in this master thesis. Later chapters in the master's thesis cover the areas of research and methodology for three failed experiments carried out at Telavox. It discusses the methodology into how lean might fail, and how it could be implemented correctly, to ensure success. Next, the results of the three experiments are discussed in terms of why implementing lean failed, (on a more practical level). It then sums up to how to achieve lean using a model that is divided into two parts. Internal disruptive innovation and incremental innovation. To boost internal disruptive innovation it examines a lean startup model from the time when Telavox was a startup company. It also examines what waste was detected during the rapid growth period after the startup era. To keep startup capabilities the solution is a sandbox organisation that are argumentet for. To increase incremental innovation and reduce wastes a tool named communication flow kanban is presented. The thesis evaluates what solutions the model can give when implementing lean, and lastly, the discussion and references are presented.

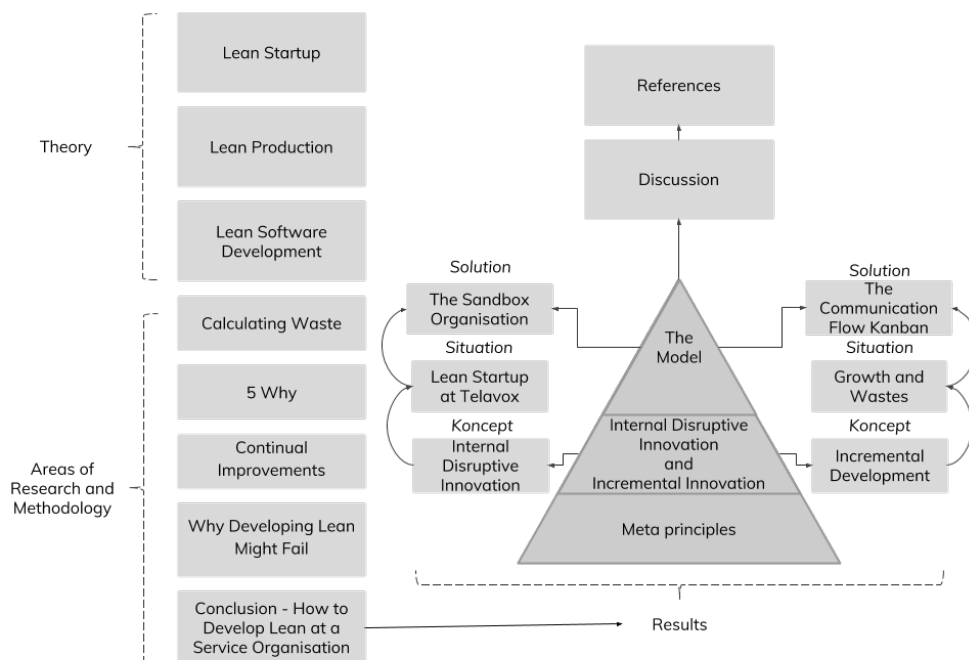


Figure 1 Master Thesis Structure

3 Lean Startup - Creating the Foundation

According to E. Reis the definition of Startup is

“...a human institution designed to create a new product or service under conditions of extreme uncertainty.” (Ries, 2011, p. 27)

There are no clear definitions of a startup, however many business leaders have the same view upon making a business under uncertain conditions. (Robehmed, 2013)

Even bigger companies can use lean startup’s methods, and maybe should. When forming an autonomous division with separate profits and loss targets, they can work outside the corporate culture. This forms an entrepreneurial startup culture with the same strategy as the completely new born companies. (Ries, 2011) Lean startup works well to innovate disruptive innovations in an effective way.

The company, old or new, should have a true north, a vision to guide them in their work. This is the foundation of the company where it thrives to achieve progress. When having a vision they obtain a strategy and therefore the means to achieve the overall goal. The strategy includes the business model, a product roadmap, ideas about potential partners, competitors etc. The product is the end result of the strategy. The product might change with optimizations, the strategy or ‘pivot’ changes less frequently and the overall vision rarely changes.

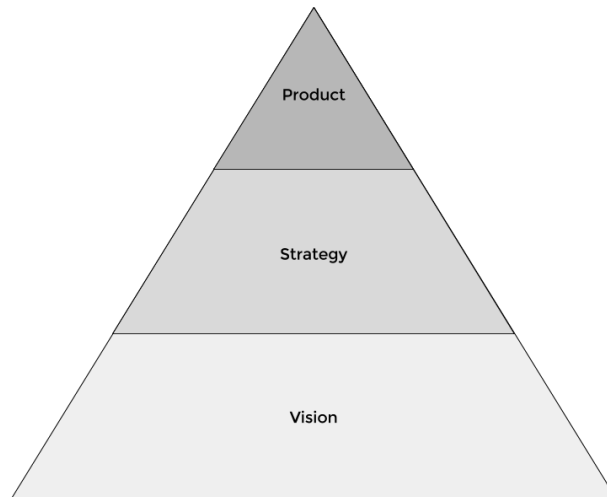


Figure 2 The Startup Triangle

The section that follows will give an understanding of how to find a good strategy to develop a good end product. The thesis will not examine how to obtain a good vision.

Using lean has proven to work for many different companies in different industries, but why use it for a startup? Lean startup doesn't involve all the philosophies that originate from Toyota. However, a startup usually has scarce resources and since the lean philosophy is a way to use the resources as efficient as possible, this lean thinking can be valuable. Lean thinking defines value as providing benefit to the customer, everything else is waste. The lean approach in production is to increase value creating actions and reduce waste actions to produce benefit to the customer. Since a startup won't know for sure who the customers are, the same definition is hard to apply. Lean startup is about learning and finding the right strategy, the right product and the right customer product fit in an efficient way. (Ries, 2011) The learning focus is to understand your customers, their behaviors and their needs. The problem is that customers usually don't know what they need until they've tried it.

Effectiveness is producing the right things, that's what the startup companies have to try and find. Efficiency is how well the things are done, this is the classic lean thinking. When measurement systems are better and it's easier to get data on eg. production facilities, a lot of the focus is towards increasing efficiency. Nowadays since it's easier and cheaper to produce almost anything, the way to get the competitive advantage is shifting from how to produce to what to produce. (Drucker, 2006)

3.1 To Work Effective - Experiment

Startups need to experiment on the vision to discover how to build a sustainable business around that vision. The scientific way to do experiments is to begin with a hypothesis that makes a calculated guess on what is supposed to happen. Then the hypothesis is tested empirically. The experiment should be made on potentially *early adopters* and then spread all the way to *laggards*. Early adopters are more forgiving on the small mistakes, what is more important is their eager to give feedback. A big difference from taking in information through a survey for example, or other market research before the product or concept is tested, is that the customers have tried it in an experiment. They have facts about their behavior that can receive concrete answers about what to change, improve or discard. (Ries, 2011)

3.2 Build-Measure-Learn Feedback Loop

The build-measure-learn feedback loop (Figure 3) is the core of the lean startup model. People with different backgrounds have different special abilities. Engineers might be good at creating products efficiently, managers have great ability to strategize and find models on the whiteboard and entrepreneurs are great at coming up with new ideas and inventions. As individuals they won't get far, and increasing their individual abilities are not

the primary goal, the goal is to minimize the time it takes to get through the build-measure-learn feedback loop. This means they will obtain more feedback and reduce waste by making sure the company is heading on the right path. Each loop increases the knowledge about the customer and how well the product/service fits for them. (Ries, 2011)

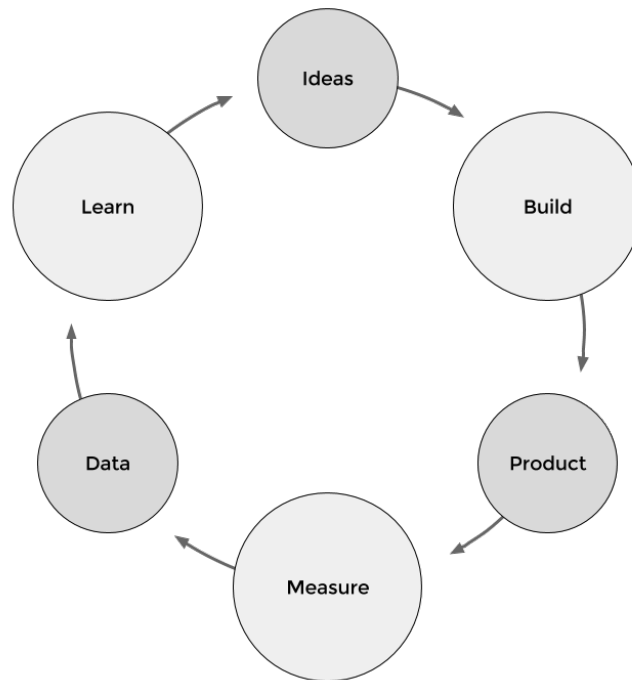


Figure 3 Build-Measure-Feedback loop, Inspiration from: (Ries, 2011).

3.2.1 It starts with Assumptions

The startup should have assumptions, or some hypothesis to test that it all depends on. There are two assumptions that are important to identify, the value hypothesis and the growth hypothesis, which will be examined more closely later.

Business strategy is based on assumptions that can be hard to determine. There are two ways to make the best possible assumptions. The first way is for the managers to receive help to get better assumptions eg. consultants. The other way is to test the assumptions early and pivot or change the assumptions. For a startup the latter is preferred for various reasons:

- The cost of failure is low, they have no or few customers who are affected.
- It's a learning process, what they learned stays in the company and could be valuable in the future.
- They can't afford outside help.

(Ries, 2011)

3.2.2 Build - Minimum Viable Product

When the company knows what to test they should start to build as quickly as possible. One way to do this is to create a Minimum Viable Product (MVP). A MVP is defined as that which is just enough to be able to do a full turn in the build-measure-learn loop. It might lack a lot of features that are essential later on, but this is no concern since it's just about testing the concept. Since a test needs to be made, some extra work might need to be done so that it is possible to measure it's impact on the customers. It's not enough to have a group of internal people giving their opinions, it needs to present it to potential customers to see their reactions. One example of a MVP is Dropbox's video early in the process before they'd released the beta version. The video described the concept of what they intended to develop, the result was the amount of signups as beta users rose from 5000 to 75000 overnight. This was proof that they were heading in the right direction and that the market, or at least the early adopters, was mature enough to accept the product. (Ries, 2011)

For some people it could be hard to create a MVP and have early releases. It is quite the opposite from textbook engineering methods like 6-sigma. It even comes down to pride where it could be hard to show a half developed product to friends, family or colleagues. But in this early state, quality is second hand. E. Reis make this clear with the quote.

"If we do not know who the customer is, we do not know what quality is."
(Ries, 2011, p. 107)

Therefore the MVP needs to focus on finding the value for the right customers. The lack of quality of early releases should rather be seen as learning opportunities. If the customer believes that the MVP has low quality, it's a learning opportunity with attributes the customers cares about.

3.2.3 Measure

Measuring should prove if the developing process is going in the right direction i.e. if the product is improving. Developing something that no one is using is the biggest kind of waste so it wouldn't matter if it's on time or on budget. The measurement should be quantified and not given vanity metrics. The measurement and data will be used to analyze and learn about how well the product works for the customer. If the hypothesis is false, a different course needs to be taken with a new hypotheses to test for the next loop, the company needs to pivot. When making a pivot sooner rather than finding the product unwanted later, the company can minimize the waste of time and resources. (Ries, 2011)

One general principle about lean is constant improvement, and therefore it is important to know what actions render positive or negative results. To be able to do this you need data. For bigger companies this can be complex

analysis with revenue as the bottom line. For startups it is more important to look into usage. One way to do this is with cohort analysis, it looks at the performance or activities from the perspective of a group of people (a cohort). When grouping customers by a defined time-span in an individual's life cycle, they can show specific characteristics, give data to developing features according to the pattern and see how it evolves. It could also be used with cohorts of users of different sets in set-based concurrent engineering, this helps to see which set seems most promising. (Ries, 2011)

A cumulative analysis could show a vanity metrics. For example if the product got more users, spread faster and in total gave higher income, this might seem great. However, if usage decreased compared to the sign-up numbers due to bad a development direction, this could cause the rapid sign-up numbers to be deceptive. Marketing and other methods of spreading the product could be the reason for its early success, not the product itself. (Ries, 2011)

The metrics should consist of the three A's, Actionable, Accessible and Auditable.

- Actionable must demonstrate clear cause and effect. For example the increasing numbers of hits on website could be due to larger groups of employees who use the website. They might frequently visit it to get information and the newest pricing of the products. In this case the metrics should include where the visitors came from. If marketing did a campaign thru an e-mail newsletter they would like to know how many hits this generated on the website, excluding their own personnel. Vanity metrics is dangerous because it reflects directly on people's minds. If numbers go up it's possible that employees would think it was due to their actions, however the cause might be totally different. (Ries, 2011)
- Accessible is about creating reports everyone understands. Reports are made to guide people into making the right decisions, if the report is hard to understand the decisions might be wrong. People might waste time trying to figure out the data rather than using it as feedback for future actions. For example an employee might ask "*what is a website hit?*" It can be hard for one to understand. Instead the phrase "*a person that visiting a website*" could be used. It is so clear it is even possible to imagine a person sitting on a computer surfing on the website. This is one reason why cohort analysis is good, it turns complex analysis to people-based reports. A cohort analysis answers; among the people who used a product under a certain period, how many of them exhibited each of the behaviours that are relevant? Accessible is also about how easy it is to get hold of the report. If it sits on someone's desk it won't create value until someone asks for it. (Ries, 2011)

- Auditable means the metrics should reflect how the real world acts. This can be a problem when a project owner's pet project has proven to be a failure. The project manager can often try to find a scapegoat, or failure in other areas such as the data or the goods etc. If the data is truly auditable there should be no discussions about such things, it should be clear what went wrong, what areas need improving and what's been learned from the project. To make sure the data is auditable, it should be able to be tested. This is done by communicating with the customers to make sure that the data contains facts. Data should be withdrawn from the master data and not from intermediate systems that could be one possible source for errors. (Ries, 2011)

3.3 Learn and Pivot

Even though the phases of the build-measure-learn feedback loop should be executed in the order written, it should be planned in the opposite direction. To be a startup, research needs to be done to find capabilities and a good market-product fit. That's what the startup should look for. So after the build-measure-learn Feedback loop was executed and it proved to be a failure. This since the product/service was not accepted by the market, change is needed with possibilities of new insights and learning. With the findings from the data and the build phase on the earlier feedback loop, the company can make a pivot, a radical change of strategy to hopefully create a better product fit. According to E. Reis the telltale sign for pivot is:

“The decreasing effectiveness of product experiments and the general feeling that product development should be more productive.” (Ries, 2011, p. 164)

A company needs to cross the chasm before the runway ends. The chasm is the gulf between the early market and the late market. The early market are small but have customers that easily adapt to new technology, the late market is larger but take longer time to adapt to new technology. Since the late market is larger it is there where majority of the return on investment can be made. The runway is the time that the company has until it runs out of cash. It could be calculated as the amount of money the company has left divided by the net burn rate. To get a longer runway the company could raise more money or cut costs, but cost savings could decrease the speed of the feedback loop. It is not important to have a long runway, what is more important is the number of pivots the company is able to undertake. If the company made several quick pivots and making sure to learn during each step. On each pivot they tuned the value proposition closer to the demand of the market, but still didn't succeed. At least they didn't spend a lot of time just keeping their head above the surface with a product that was doomed to fail. (Ries, 2011)

There are several different pivots that can be used to find a more suitable strategy: (Ries, 2011)

Zoom-in Pivot

When a part of a product becomes the whole product. Focusing and developing the ‘right’ feature to make it more valuable for a larger consumer base.

Zoom-out pivot

When a single feature doesn’t satisfy the customers needs and the product scope needs to expand so the initial product becomes a feature.

Customer Segment Pivot

When the product solves the problem but not for the initially intended customer. Change of customer base solves the right problem for the right customers.

Customer Need Pivot

Early customer feedback indicates that the customer’s problem is not that important, so the product development might need to take a different direction or develop a totally new product.

Platform Pivot

This refers to change from an applications to an platform or the other way around. An entrepreneur might have a vision to create a platform for future products, but don’t have any good applications that creates value for the customers yet. Then there would be hard to make the customers use the platform.

Business Architecture Pivot

Geoffrey Moore observed that companies generally follow two sales models. High margin and low volume as often the case with Business to Business (B2B), or Low margin and high volume like Business to Customers (B2C). This is applicable and could prove valuable to change depending on the outcome of the product.

Value Capture Pivot

Where the company creates monetizing value in terms of features that could be added or removed at the customers will.

Engine of Growth Pivot

The company can have three different engines of growths: The viral, sticky and paid growth models. The company changes the model to achieve faster or more profitable growth.

Channel Pivot

The sales or distribution channel is the way a company delivers its product or service. The pivot could be either letting someone else distribute the value offer or direct sales.

Technology Pivot

When a company figures out how to solve the same solution by using different technology. This is more common in established businesses. It often has the same value offer to the customer, but saving costs could be made to a superior price and/or performance.

When the company has tuned in their strategy and reached the early adopters it's time to focus on quality. This requires another pivot when trying to reach for another customer segment. A low quality product works as proof of concept but the mainstream customer is more demanding and wants higher quality. Timing the fast phase production development and producing higher quality products could require new metrics and different methods of product development, maybe even a small change of the culture in the company. How to achieve higher quality in products will be examined later with lean production and lean development.

3.4 Growth

Finding value in making products for the customer is one area of the build-measure-learn feedback loop, the other is the growth hypothesis. E. Reis characterize growth with one rule; *New customers come from the actions of past customers*. This can be divided in four key aspects of growth:

1. Word of mouth, recommendations beat any sort of commercial actions.
2. Side effects of product usage. If your neighbor buys new clothes or new luxury boats there is a large possibility that they will be influenced to buy that product too.
3. Advertising is a part of the marginal cost. As long as the marginal cost is lower than marginal revenue, which is the revenue from the customer's life cycle it should be done.
4. Repeated purchases. Some products like groceries are voluntary repurchases or there could be an automatic repurchase like subscriptions.

Startups should not focus on complex metrics and long discussions of what to do when a product is finally developed. All their energy should be invested in finding new customers, serving existing customers better, improve quality or lowering the cost of the product. (Ries, 2011) E. Reis has designed three engines of growth that have a small set of metrics to focus on.

3.4.1 Sticky Engine of Growth

One way to measure growth is to measure the rate of customer acquisitions and the churn rate. The churn rate is the fraction of how many customers in a

certain period that fail to remain engaged in the customer's product. The speed of growth is therefore growth rate minus churn rate and is called the compounding growth rate. (Ries, 2011)

3.4.2 The Viral Engine of Growth

Viral growth comes as a result of person-to-person interaction as a side effect when using the product. To measure viral growth there is an easy single mathematical term - the viral coefficient. The definition is *How many new customers will use a product as a consequence of each new customer who signs up*. The diagram below illustrates this. A company above 1 will have an exponential growth and that is ultimately the goal. If it's below 1 the company will need to put resources in actions like advertising to be able to grow. (Ries, 2011) A viral coefficient of 0.2 means if 100 customers are acquired, 20 would be acquired virally. Out of the 20 acquired 4 more would be acquired and the loop would be closed and others actions to acquire customers would be needed.

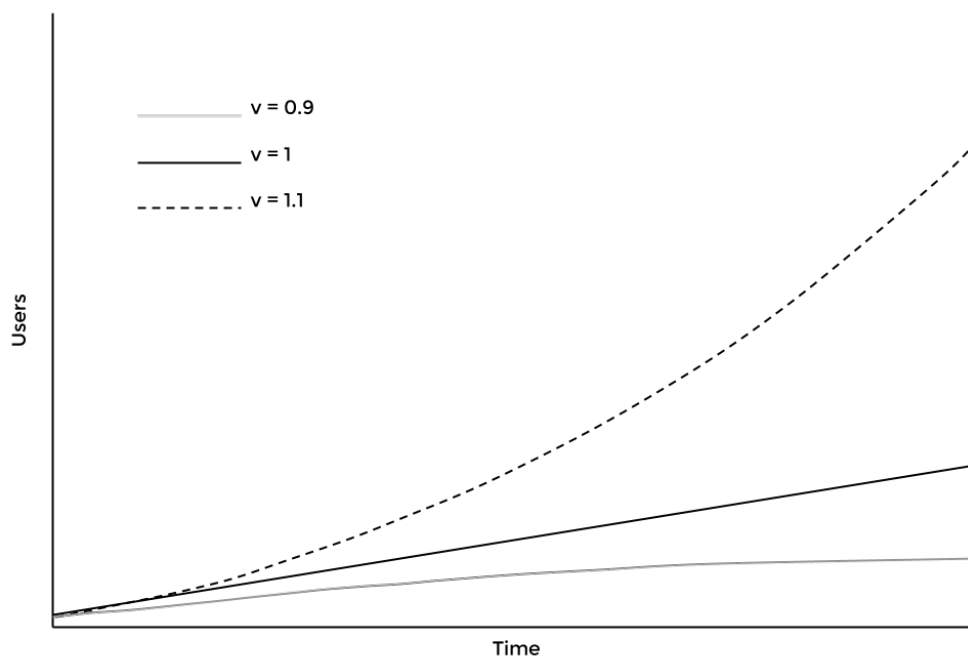


Figure 4 Viral Coefficient, Inspiration from: (Ries, 2011).

Small actions could have largely impacts on the viral coefficient. That is one reason why many products that use viral growth focus on indirect revenues like advertising. This is because even a small charge might be enough threshold for the customer to lower the viral coefficient and drastically decrease growth.

3.4.3 The Paid Engine of Growth

A customer creates revenue through their lifetime as a customer. This value is called Customer Lifetime Value (LTV). To get a customer, it's an acquisition cost. This is either by salaries to sales people and/or cost for advertising for a single customer. This cost is called Cost per Acquisition (CPA). If the LTV is larger than the CPA there is growth, and the margin between will determine how fast the growth is. This is if all the revenue is reinvested in the acquisition of new customers. (Ries, 2011) In some industries they lock in the customers, such as the telecom industry, with subscriptions to make sure that the cost of acquisition will be paid in the end.

3.4.4 Adjust the Growth

The company can grow too fast, which is a paradox of the build-measure-feedback loop that advocates speed. Andon system is a great example. Andon system are used in the production industry and is a way to notify management or maintenance that something is wrong. The andon system can also stop the production line therefore force the problem to be investigated. When this is done the company makes sure that no problem will show up twice. Even though it is a short term cost (the system stops) there are long term profits (it will never stop for the same reason again). If the system wouldn't stop, and the problems are momentarily with work arounds or quick fixes, the problems will stack up later and create a legacy of problems that will create ripples all over the company. To slow the processes and make sure that errors don't show up twice might inhibit growth for the moment, but ensure faster growth in the long term. (Liker, 2004) This could be contradictory to what was earlier said regarding creating MVP's in short loops with applications that could have many bugs. However, this phase is intended to start when the innovators and the early adopters have given their approval of the product, and when the company change/pivot the strategy to be able to cross the chasm.

When the company is sure of their path, they should start asking why, and do this five times like a speed regulator. The core principle of *5 Why* is to truly understand the problem and what happened - the root cause. For every step closer to the root cause the countermeasure will change. (Liker, 2004) Taking two step forward and one step back, keeps control of the growth and development pace and tries to keep legacy problems out of the way.

3.4.5 Worse Before Better

Introducing Lean to existing companies often means things get worse before they get better. There is a switching cost that needs be taken into account. Therefore it is important to set the expectations right. Lean is not the answer to all problems, it's quite the opposite. It's function is to reveal problems and this might be hard in the beginning. If expectations are too high too early, this could cause the company to fall back into it's old ways. (Ries, 2011)

4 Lean production

Lean principles became famous from a set of tools spawned by Toyota such as *just-in-time*, *Kaizen*, *one-piece flow*, *jidoka* and *Heijunka*. However, the success of Toyota goes deeper than just those tools. The tools work due to a deeper business philosophy based on understanding of people and human motivation. Jeffrey K. Liker created a model to visualize the principles, these principles can consist of tools to help achieve operational excellence. The chapter that follows will follow the model and explain the philosophies from the bottom-up.

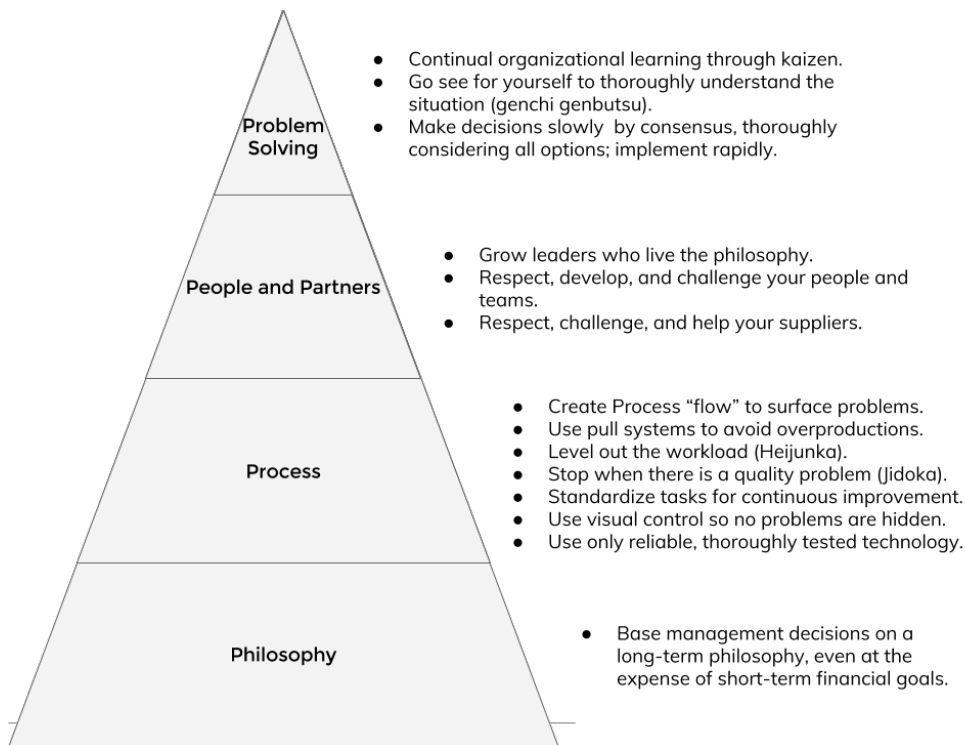


Figure 5 The Lean Triangle, Inspiration from: (Ries, 2011).

Many of the principles in lean derive from a mindset to reduce waste. When examining any process in a production company, the first question should always be “*what does the customer want from this?*” according to Jeffrey K. Liker. This includes internal customers (the next step in the production line/value chain) and the final/external customer. The question defines value, and all that does not create value for the customer is waste. Toyota has defined eight different kind of waste in the production process, later definitions will be made for the software development industry.

1. Overproduction

Production of something where there is no demand/order. In other non-manufacturing industries this could be a line of code that never will be used, or overstaffing on service time when there are no customers with questions.

2. Waiting

When personnel have nothing to do, in production for example waiting on a machine, waiting for more materials etc. In general this could also be waiting for a meeting to start, waiting for new tasks.

3. Unnecessary Transport

When increasing work in progress (WIP) with unnecessarily long transportation. In the service industry it could be the equivalent of taking decisions through a long line of managers.

4. Overprocessing or Incorrect Processing

When the results/product doesn't match the expectation. For example in cases when a tool is not right for the job and therefore increases defects. Overprocessing is when the quality is too high and are not valued by the customers.

5. Excess Inventory

Problems could be hidden when there is no risk of not being able to deliver to a customer. Causes it to become harder to notice problems or possibilities for improvement. More WIP or finished goods increase storage costs, longer lead times, damage goods could be hidden for a longer period of time etc.

6. Unnecessary Movement

When workers make unnecessary movements such as reaching, looking or stacking for parts or tools.

7. Defects

When spending time or money on defects such as re-doing or repairing, scrapping and inspecting.

8. Unused Employee Creativity

When there are losses of human capital. This can happen when a company is not making use of employees ideas for improvement, their skills or not developing their knowledge.

4.1 Philosophy

4.1.1 Principle 1 - Long-Term Thinking

Management decisions should be taken on a long term philosophy, not on short term financial goals. The company has some responsibility to customers, employees, suppliers and society as a whole, and these are regulated by law or unwritten rules. But what is more important is that companies have a reputation/expectations to live up to and with long term definition and clear values, it is easier to get the company's stakeholders to understand and set the

expectations right. Then it's important for the company to live up to the expectations, and to be able to maintain trust. For the employees, they could feel a sense of purpose rather than just serving the company, and in return earn a living. (Liker, 2004)

This is more complicated than it might seem since there are economic forces driving most decisions. For example, a family run business could have more incentives to make more long lasting decisions than a public run business. In a public run business the shareholders would like return on their investments as soon as possible. Family companies could find it easier to develop more loyal staff. Even though they may not be able to hire top consultants, the teams that are created last longer. (Survival of the fittest, 2015)

4.2 Processes

4.2.1 Principle 2 - One-piece Flow

Flow is about going from point A to point B with the least time possible. In production this means to go from raw material to finished goods as efficient as possible. The results of working with this can be better quality, lower cost and shorter delivery time. (Liker, 2004) One-piece flow is in a way the core of many other lean tools and philosophies. It is easy to believe that one-piece flow is just a way to see how the materials are running thru processing steps in an production line. But one-piece flow can be seen in other areas of work as well. For example how information flows in a company. A case of information flow can be when an analyst makes a report with propositions for changes that sits and waits for decisionmakers. The decisionmaker forgot to give feedback so it arrives months after the report was made. The decision was that the report was good and that the proposed changes should be implemented. But due to changing conditions in the company the report might not be viable anymore so the analysts have to do a new report and start over the process. One have flow when someone is placing an order, then that triggers production to get just the right amount of raw materials and start processing without stops with waiting for something else all until the product is reaching the customer.

This can give some benefits.

Quality

A higher degree of flow reduces batch sizes. When batch sizes are reduced it increases possibilities to find defects earlier which otherwise could be hidden in the inventory. If a batch is 100 pieces large, and piece number 10 is faulty, it will produce 90 other faulty products until the error is found and can be corrected. Every operator should therefore inspect their work and fix the problem in the station before going to the next step. This ensure that defects are detected early and that the problems can be solved directly.

Higher Productivity

Depending on how to measure productivity the results could vary. But since one-piece flow visualizes waiting time and quality problems it is easier to solve the problem that over time can increase productivity. Let say that a company measures productivity with Overall Equipment Efficiency (OEE) that is defined as:

$$(1 - \text{Downtime Rate}) \cdot (1 - \text{Production Rate Loss}) \cdot (1 - \text{Scrape Rate})$$

(Ståhl, 2013)

The downtime rate could increase in the beginning due to lower inventory, or increase the production rate loss due to bottlenecks. So the OEE could get lower in the beginning due to one-piece flow. But since the problems is clearly visualized as mentioned before the problems could be fixed and long term increase efficiency. Another problem with measurement is that when people are overproducing parts, and fills up inventories it gives vanity metrics in machine and staff utilization. They are utilized which looks good on paper. But they don't necessary create value for any customer.

More space

When arranging production in cells and keeping less inventory the production site is more compact and this reduces waste of floor space.

Improved safety

Since batches are smaller there is less need for moving large chunks of materials with which could cause eventual injuries.

Reduces cost of inventory

Since a batch binds up inventory, flow can free up capital to invest elsewhere.

(Liker, 2004)

It can be hard to implement one piece flow. There might be high costs when there is a breakdown on a machine that puts a stop on the whole production line. This create incentives to go back to the old way instead of fixing all the problems so that there are less breakdowns instead. Bottlenecks can have large effects on the whole production for the same reason. In some cases investments might be needed to fill gaps in the process that earlier was outsourced. These kind of investment might be too large for some companies. (Liker, 2004)

4.2.2 Principle 3 - Pull Instead of Push

The pull system minimizes inventory while producing products just-in-time and deliver when the customers needs it. The most optimal pull system would be zero inventory, where production starts the moment a customer makes an

order. This can be impossible to achieve since the production wouldn't be the most efficient and the customers expect quick deliveries and don't have time to wait on production. The supermarket provided the inspiration for creating an effective pull system. When a grocery item is taken from the shelf, the empty space signals that that particular grocery item needs to be restocked. The system was created based on visualizing signals, and was given the name kanban. Using the kanban system in the area of production showed the apparent need for a processing station. The station uses a kanban card that is sent back through the value chain and pulls the orders through the system. The card signals when there is a demand up the value chain, specifies what is depleted in terms of products, parts, or inventory and triggers production to start replenishing them. (Liker, 2004)

The push system can be preferred in some instances, it attempts to produce against a projected customer demand. In production for new phone model, the market demand can be extremely high just after release, and can quickly decrease due to the high competition and the short product life cycles. Therefore, it is important to be able to ship high quantities of phones just days before the release date. Factories are designed to produce many phones as quickly as possible against large inventories, and continue to do so until the release date of a newer model closes in. (Larsson, 2015)

4.2.3 Principle 4 - Level out the Workload (Heijunka)

Customer demands vary and can change from week to week. When using pull to reduce inventory, it is important to level out the workload so people and machine are not overburdened by higher customer demand. On the other hand when demand gets lower the people will have a little to do and machines are not operating at it potential. Toyota see the importance of this and uses three terms that works together to level out the workload.

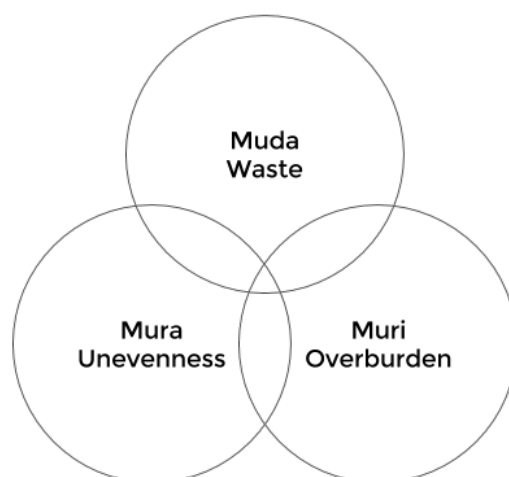


Figure 6 Muda, Mura and Muri, Inspiration from: (Liker, 2004).

Muda - Non-Value-Added

Muda include the eight wastes; Overproduction, waiting, unnecessary transport, overprocessing or incorrect processing, excess inventory, unnecessary movement, defects and unused employee creativity.

Muri - Overburden People or Equipment

Muri is created when you push equipment or overburdened people that could risk their safety. Overburdened equipment could cause breakdowns.

Mura - Unevenness

Mura has dependencies of both muda and mura. If one intend to start using lean and only reduce muda by decreasing inventory with one piece flow - there might be a bottleneck where people have to work very hard. If customer demand increases it might make the system collapse. This is one of the most common examples of not convincingly implementing lean. To be able to achieve success it is important to have a well balanced organization. This is where heijunka comes in, leveling out the workload so there is less risk of muri and making it possible to reduce mura. (Liker, 2004)

To achieve heijunka, all parts of the organisation need to work together. The demand on people, equipment and suppliers should be even and the product mix and volume should be well balanced.

4.2.4 Principle 5 - Build in Quality (Jidoka)

Jidoka comes from the founder of the Toyota group, Sakichi Toyoda, who created an automatic loom. One of the inventions was that when the string broke, the loom stopped. This function made it easier to reset the loom and most importantly fix the problem, so that the defect didn't repeat itself.

Jidoka can also be called autonomation, this is the word for automation with human intelligence. The machine stops when it detects a problem so that the human can investigate why the error occurred and fix it. One approach to visualize when an error occurs is the andon system. Here an operator can push a button to send a signal (light a lamp) and flag a problem so a manager can come for aid. If the problem is severe and can't be fixed at once, the production line stops. Stopping a whole line is an expensive way to make sure that just one product isn't faulty. In long term though it is beneficial, fixing the problems when they occur makes sure the problems won't occur again or create more problems further up in the value stream. (Liker, 2004)

Poka-yoke is a system built to avoid workers making human errors. The concept of the poka-yoke is reducing efforts into finding the unacceptable products, the process should involve a system so that all products are correct without needing to sort out defective parts. There are two types of poka-yoke,

prediction and detection. Prediction recognize a defect before it occurs and detection after it occurs. An example of prediction poka-yoke could be a kind of sensing device on a fixture so that the system wouldn't start if the part isn't inserted correctly. Another example of prediction poka-yoke is an USB drive that you won't fit in the USB port if it's not turned correctly to make sure none electronics in the device could be broken. The detection poka-yoke is a built in system to find errors that already occurred. If the system find error it stops the process. (Inman, 2006)

Built in quality could be used in a service organisation as well. The Andon system is preferably applied in working environments where there is repetitive work and short-cycles. (Liker, 2004) This could be the case in an office environment such as call-centers.

Set-based Design Process

For other lines of work set-based concurrent engineering could be used. When trying to make decisions it can be time consuming and hard to figure out one optimal answer. Set-based concurrent engineering takes many alternatives into consideration and studies them thoroughly. Every option can give insights and learning that should be taken to account in the future. Step by step with the learning the options are converged into a single solution. This is related to jidoka since this slows down the process by doing a more thorough reflection of all options before going forward.

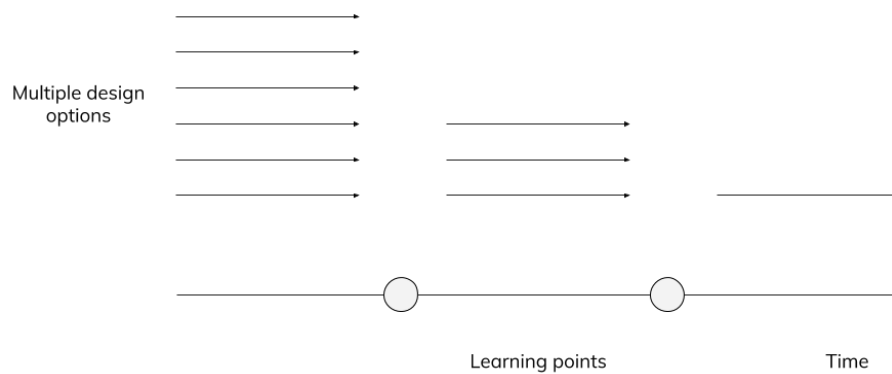


Figure 7 Set-based Design, Inspiration from: (Scaled Agile®, Inc., 2016).

In the set-based approach, a few specifications are made in the beginning of the process, constraints are decided instead of clear specifications on a single options. Constraint-based communication is a powerful tool for communication, and works as a framework for boundaries decisions/development. An example of this can be in trying to schedule a time for a meeting. If everybody communicates the times they are not available, it's easier to find a time that works. Another benefit of constraint-based communication is the possibility to make decisions as late as possible. Dye makers in the auto industry can start to produce the dye at the same time that the design process is made. The further

the design progresses, the more the dye makers can cut out from the original material. So when the design is totally finished, the dye will be ready and functional and therefore decreasing time to market. This also made it possible to decide as late as possible since minor necessary changes in the design wouldn't effect the die which is expensive to build. If decisions of the design were made earlier, and the die were crafted for the initial design. But in the end of the developing process some problem came up that needed the design the change, the whole die would be needed to be scrapped and a new one made. Therefore the decisions should be taken as late as possible so problems that are found can be solved in a flexible way. (Poppendieck & Poppendick, 2003)

Set-based design is closely related to split testing. Split testing can for example be used on website to see what design, colors etc. works best to get the desired result. Split testing in practice can work as following. A website have different variations of designs, the design have small differences as different colours on a button, different text on the heading etc. The different websites are distributed to the visitors, who are not aware of the experiment. Lastly the data is withdrawn that measured their interaction so that an analyzed can be made how it was correlated with the different designs. By using tools that compile the data on the action the users are taking decisions can be made statistically significant on the visitors behavior. (Optimizely)

Point-based Design Process

On the contrary to set-based concurrent engineering the point based design process takes one option and pins it down with a specific and detailed requirement early. This is iterated and adjusted through the process to try to achieve a more optimal solution, however major flaws could be seen late in the process. At this point adjustments can't be made as these would need budgeting and/or time. Another problem with point-based design is that the system designer lack of empirical data to be able to make good decisions. In the set-based approach all the failed/non chosen designs could give valuable data which is deprived in the point-based design process. (Scaled Agile, Inc., 2016)

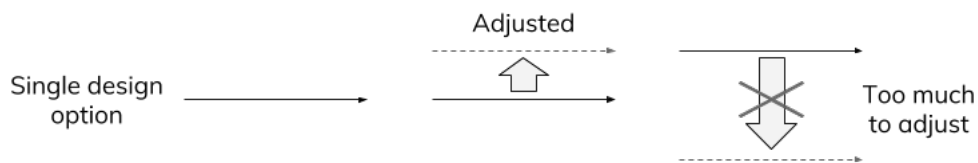


Figure 8 Point-based Design Inspiration from: (Scaled Agile®, Inc., 2016).

Point based design could also be referred to as serial engineering, or in development, the waterfall method where all functions or steps along they way are each designing to a single solution. When the best solution is generated they *throw it over the wall* to the next function.

Point-based process set requirements in the beginning while the set-based processes sets the constraints. To compare the two the method of scheduling meetings can give deeper insights.

Point-based way of scheduling a meeting:

“let’s have a meeting at nine o’clock, can everyone?”

Everyone answers no

“Let’s have the meeting at ten o’clock instead, can everyone?”

Everyone but one answers no

“Let’s have the meeting at eleven o’clock instead, can everyone?”

Everyone answers no

“Let’s have the meeting at twelve o’clock instead, can everyone?”

Everyone answers yes but one who is busy.

“Let’s have the meeting and we take notes for the single person who misses it”

Set-based way of scheduling a meeting

“What times can’t you have a meeting?”

Everyone answers and the staff are busy between ten to four o’clock”

“Okay, let’s schedule a meeting between four and five o’clock”

The example proves that point-based way of scheduling a meeting can be time consuming and might not end up picking the best time since they feel forced to let someone miss it. (Sobek II, Ward, & Liker, 1999)

4.2.5 Principle 6 - Standardize

Standardization in production started in the early 19th century with Frederick Winslow Taylor, the father of Scientific Management. He found a way to increase production, decrease the cost and stimulate the workers with rewards such as higher wages.

Time and Motions Study

Taylor’s way of doing so was by dividing the workforce in two, laborers and management (blue and white collars) and it was the management’s mission to break down the laborers work tasks into smaller elements to reduce waste. All work was supposed to have one way that was best, the manager’s job was to find it and instruct the worker to do it.

Standardize Tasks and Procedure

Some working areas had workers using their own tools, some that were not fit for the purpose. Taylor saw the flaw in this and figured that both the design of the product and tools should be standardized to achieve higher efficiency.

The Task

The workers should have a task fit for their skill level, it was supposed to be specific (quantitative) and challenging for the worker.

The Money Bonus

The worker should be paid 30-100 percent higher when working with scientific management compared to the 'old way', this is in order to be sure that the worker was carrying out orders.

Individualized Work

Taylor advocated that the work should be done individually and therefore also rewarded individually. Group work and rewards were believed to undermine individual productivity. He wrote

“Personal Ambition always have been and will remain a more powerful incentive to exertion than a desire for the general welfare”

(Locke, 1982, p 17)

Management Responsibility for Training

Workers should not be learning from older experienced workers since they might not use the scientific *one best way* solution, but from managers who thoroughly knew the work.

Scientific Selection

This is a scientific way to find the best workers. It was believed that selecting the *first class* worker could increase the productivity by several times greater than the average man.

Shorter Working Hours and More Frequent Rest

Taylor discovered through experimentation that more rest and/or shorter working days in proportion to the work difficulty would increase total productivity.

(Locke, 1982)

This proved to work by the time Taylor released the book *The Principles of Scientific Management* in 1911. It was later realised that when people know they are being monitored, they work to reach their number without concern of quality. Also they might not share their improvements as it would increase their burden. A bureaucratic company can create tall, hierarchical organizations, companies with top-down control, a lot of rules and procedures with a lot of documentations, poor communication, resistance to change and static and inefficient rules and procedures. All if this inhibit companies innovation and increase wastes. (Liker, 2004)

Some degree of standardization is necessary, because if everything is changing from day to day, an improvement could go by as just another variation. Without standardization and stability, there is no way to have

continuous improvement (kaizen). Standardization is also an important factor for built in quality, if there are variations of standard it is easy to find out why the problems occurred. If quality failures are the issue, whether the standard is being maintained or not, there is still room for improvement. (Liker, 2004)

Companies can be describes as two types, either they are bureaucratic with the description above, or they are organic with less rules and structures. Combining this with social structure explains some of the differences between Toyota way and Taylor’s scientific management. (Liker, 2004)

		Social Structure	
Technical Structure	High Bureaucracy	Coercive Bureaucracy <ul style="list-style-type: none"> ● Rigid rule enforcement ● Extensive written rules and procedures ● Hierarchy controls 	Enabling Bureaucracy <ul style="list-style-type: none"> ● Empowered employees ● Rules and procedures as enabling tools ● Hierarchy supports organizational learning
	Low Bureaucracy	Autocratic <ul style="list-style-type: none"> ● Top down control ● Minimum written rules and procedures ● Hierarchy controls 	Organic <ul style="list-style-type: none"> ● Empowered employees ● Minimum rules and procedures ● Little hierarchy
		Coercive	Enabling

Figure 9 Social and Technical Structure, Inspiration from: (Liker, 2004).

Toyota are enabling bureaucracy and Taylor proclaimed for a more coercive approach. For Toyota, the employees are the company’s most valuable asset, they can drive innovation. Taylor, on the other hand, see workers as machines. Toyota do follow Taylor’s scientific management though, but re-worked. They follow the principles at group level and it is based on employee involvement. (Liker, 2004)

4.2.6 Principle 7 - Visual Control

Visual controls function so one can see abnormalities at a glance. One way to do so is by having a clean and tidy working environment, therefore nothing gets hidden in the rubble.

Using 5S - programs is one method to eliminate waste due to lack of order, cleaningness etc.

The 5S's are:

Siere = Sort

Only keep what is needed and dispose what is not, things that are used regularly should be the most accessible.

Sieton = Straighten

Organize so everything has a dedicated place and everything is located at this place.

Seiso = Shine

Everything should be clean.

Seiketsu = Standardize

Creating a system and procedure that sustains the three earlier S's.

Shitsuke = Sustain

Keep self-disciplined, this could be managers doing audits to keep discipline to make sure that everything are as decided.

5S by itself is not a value creating program, it is a support function to detect problems and support flow. It is one tool for visualisation, indeed many of the tools discussed earlier are visual tools, like Kanban, Andon and standardized work. All of these support the ability to see and act upon deviations.

Visual tools work best if they are easy to use, easy to see and to take action. For managers on Toyota this could be an A3-report. All the information for decisions should be able to fit on an A3 sheet of paper. This makes sure that it's only the most important information that is highlighted and it won't be hidden under less important information. (Liker, 2004)

4.2.7 Principle 8 - Use Reliable, Tested Technology

The technology the company acquires should serve the people and the processes, not the other way around. When adopting new technology it is important to do thorough testing. Toyota will only accept new technology if it is directly tested by experiments over a broad cross-section of people. They then analyze the impact of the existing processes to see if there is value added and if it eliminates waste. If everything is according to Toyota's requirements they implement it quickly and try to minimize any negative effects on the transmission from the old to the new. (Liker, 2004)

4.3 People and partners

4.3.1 Principle 9 - Grow Leaders who Thoroughly Understand

Toyota is very restricted when hiring personnel from the outside to top-manager jobs. They see the importance of growing leaders within the company. Companies can find it necessary to change leadership in times of crisis to give a change in direction. This can cause turbulence and unevenness at work on a management level, and is opposite in theory to what eliminating Muri (unevenness) dictates. Bringing in outsiders for leadership during a period of change causes a loss in the ability to build the organization on old achievements and mistakes. Managers are not just managing on a technical level, they are promoters of the culture, and the culture is there to support the working people.

For Toyota it is important to live and understand the culture you are about to lead, therefore the leaders must understand how the work gets done at floor level. This is something that has been a core principle at Toyota since the start. Toyota's leaders, Sakichi Toyoda and Kiichiro Toyoda (founder of Toyota), understand that the work is done by getting your hands dirty.

To define the philosophy of leadership, the following model can be used.

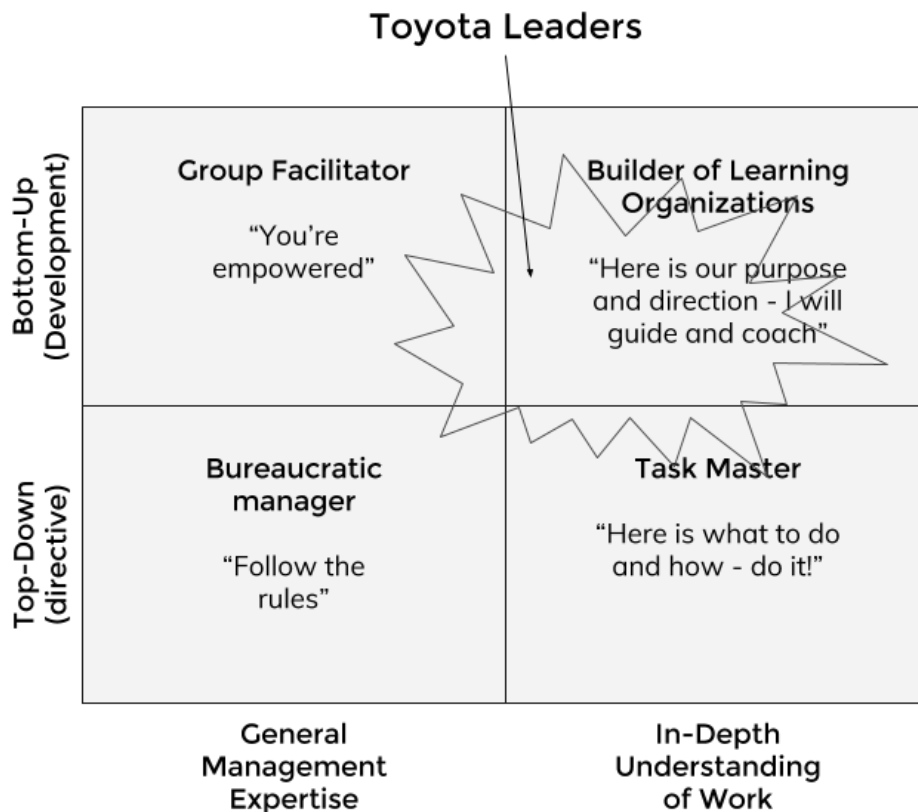


Figure 10 Toyota Philosophy of Leadership, Inspiration from (Liker, 2004).

When a manager uses the bottom-up approach, but still without an in-depth understanding of the work, they work as a **group facilitator**. This type of manager is good with people and might be able to motivate the group to work to a common goal. They are good facilitators, but when it comes to teaching or guiding junior staff in the content of work, they lack knowledge.

Managers with an in-depth understanding of the work and that use top-down directives are called **Task Masters**. They often make use of micromanagement, tell their staff what to do and how to do it.

Toyota's managers have a good in-depth understanding of the work and are good with their staff, using the bottom-up approach to empower them. They are **Builders of Learning Organisations** by developing, mentoring and leading. They do not often give orders but ask questions so that the staff can learn how to figure out their own answers.

4.3.2 Principle 10 - Respect, Develop and Challenge Your People and Teams

In America, Toyota bestowed much wisdom when implementing new facilities. An example of this was in applying the system and culture to support the teams before implement the work teams themselves. One of the managers of these facilities, Ken Elliot said:

"We are not building a warehouse; we are building a culture. This is why we have been so successful as we are". (Liker, 2004, p 187)

He also mentioned that:

"We have only one shot at this to get the culture right". (Liker, 2004, p 187)

When the culture can support the team, the team can support it's individuals in value adding jobs. In other words, a team don't just create value just because it's a team. They can help coordinate the job, create forum for team members to learn from each other and help motivate each other. Teams can also suggest ideas and improvements and be an organic control through positive peer pressure. (Liker, 2004)

If teams are not lead in the right way, or don't have the right support, they can be inefficient. Empowering members of staff too early could lead to those individuals spending too much time in meetings with no clear goals or values to have as a foundation. To create an efficient team, Ken Blanchard has a five-step model to explain the process. This is seen in the Figure 11 (read from right to left).

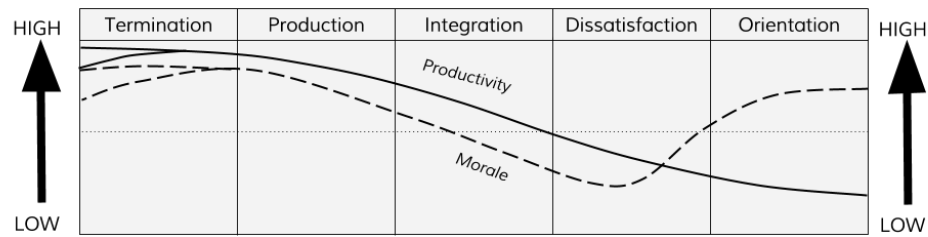


Figure 11 Team Development Stages, Inspiration from (Blanchard International, 2007).

Step 1. Orientation

A team leader sets up the direction and makes sure that the team members understand the mission, rules of engagement and the tools that are available. In this stage the team members could be quite eager to be on the team, and therefore have high expectations, and with these expectations, some anxiety regarding how they will fit into the team.

Step 2. Dissatisfaction

The team gains experience working together, however, team members can start feeling that the expectations are higher than initially expected. This therefore leads to lower morale. The difficulties working together and the problems they experience reaching their goal can lead to frustration and confusion. The leader must have focus to solve the issues and strive forward with strong directions and firm structure.

Step 3. Integration

When the issues in the group gradually resolve, the morale rises and the group find practices that make it easier to work with each other. Task accomplishments and technical skills increase leading to positive feelings. These feelings and the newly found sense of trust in the group is fragile. The group can be reluctant to argue through issues and experience conflicts in fear of ruining the positive mood. This can increase the time in this phase. The team leader must be aware that this is not the most efficient phase and must continually question the team's tendency to agree with each others. One way of doing this is to share the leadership within the group and therefore empower the team members.

Step 4. Production

At this stage, both morale and productivity is high. The primary focus should be performance and the leader should focus on new challenges and growth.

Step 5. Termination

This stage may not occur, but for temporary task forces or ad hoc teams, it might. The team might feel a dip in morale due to sadness or loss, or moral could rise due to expectations of a closing deadline. The focus should be

kept on productivity and high morale while managing closure and celebration. (Blanchard International, 2007)

In many companies, the white-collar members are responsible for the problem solving, productivity, analyses, maintenance, quality assurance etc. In Toyota it's the workers in the shop floor who are in focus for problem solving. Toyota exists to provide value for the customers, and it's the people who are bringing about this value that are at the top of the hierarchy. The managers and leaders 'under' them are there for support. (Liker, 2004)

4.3.3 Principle 11 - Respect, Challenge and Help Your Suppliers

Just as it is important to challenge one's employees to improve, suppliers should be challenged too. It is important that there is mutual respect between business and supplier, and that the supplier is not switched out the moment better prices are found. Instead both parties should teach each other to achieve lower prices long term.

The suppliers can be seen as an extended learning enterprise. This learning development doesn't just have to be of a technical nature, but just as any lean enterprise, learning thru all departments. Encouraging development in the supply staff could in turn help them to help themselves, instead of just telling them what to do.

Outsourcing has to be approached with caution. Core capabilities must not be outsourced, nor should too much dependence be placed on too few suppliers. Toyota is highly aware of this, the following quote from *A Toyota Way* indicates this:

"We strive to decide our own fate. We act with self-reliance, trusting on our own abilities" (Liker, 2004, p. 208)

This is a strong indicator that no key capabilities should be contracted to outside firms.

4.3.4 Principle 12 - Go and See for Yourself

Toyota call this *Genchi Genbutsu*, this interpreted within Toyota as going to the real place where the problem is to see the actual situation for understanding. It is hard to understand what the core problem is in any situation if you don't go and see for yourself, firsthand. Nothing should be taken for granted and when you write up reports, you are the only one who is responsible for that report. If you don't understand the problem yourself, you shouldn't be the one doing the report. (Liker, 2004)

Toyota have a some eye-opening examples of *Genchi genbutsu*. One is when the chief engineer responsible for developing a redesign of a minivan decided to go through all 50 states in america, all 13 provinces and territories in Canada

and even some parts of Mexico. He had a good understanding about the Japanese market but decided to do this since he didn't fully understand the North American. All the driving resulted in a totally new design of the car which ended up a success. (Liker, 2004)

In some cases top managers like the president of Toyota can't find time to be everywhere. He surrounding himself with people he trusts and allowing them to be his eyes and see for him. In these cases it is important to have a system of sharing information so it's easy to read and understand. When this is the case, it stimulates thinking and is a part of how companies can improve their learning practices. (Liker, 2004)

4.4 Problem Solving

4.4.1 Principle - 13 Make Decisions Slowly by Consensus

Toyota advocates the idea of spending considerable time on the planning process. This is by truly understanding everything, as discussed earlier, and getting a consensus before implementations. How you arrived at the decision is important, whether the decision turned out to be correct or not. If the analyses was correct and all the options were taken into consideration, there should be no reprimands.

To take thorough considerations, five elements can be included.

1. Understand the situation truly by looking for yourself (discussed earlier in principle 12 with *genchi genbutsu*) and finding out what is really going on.
2. Understand underlying causes or the core problem by asking why five times (described later in lean startup).
3. Broadly consider different solutions (discussed in principle 5 with set-based concurrent engineering).
4. Build a consensus with all stakeholders.
5. The information sharing and communication needs to be efficient and easy to understand. An A3 report is one tool.

Numbers 1-3 have been or will be discussed in other sections. For understanding No. 4, one can look again to Toyota and dig deeper into the word *nemawashi*. Toyota describes it's literal meaning as *going around the roots, particularly in a sense of digging around the roots of a tree to prepare it for transplant*. (Toyota, 2013) In practice, this means to have less formal meetings with before decision-making with everyone attaining the meeting. This then makes sure that everyone is aligned in the decision to lay the foundation and in the end reach consensus. This involves many people, especially in Japan where decisions are rarely in the hands of one person. Working with *nemawashi* demands that all individuals involved in the decision-making take it seriously. They should put effort into giving feedback so nothing is overlooked. (Liker, 2004)

When a lot of people are involved it is important to have routines and tools for efficient communication. This reduces the risk of misunderstandings and time consuming correspondences, as stated in No. 5 of the above elements. Toyota uses A3-reports where all necessary information should be presented. To be able to fit all information in this relatively small document, staff must learn how to communicate with as few words as possible and use visual aid. The benefits with a document like this is that it's easy to share and people actually have time to read it. (Liker, 2004)

One of the benefits with an A3 report is that meetings can be very efficient. To have an efficient meeting, the following aspects are advised.

Clear Objectives

These should be stated before the meeting, the objectives could be the agenda.

Right People

Only people concerned should be at the meeting, and if a key person can't show up it is better to postpone the meeting.

Prepared Participants

The information should be shared prior to the meeting. The participants should know what to prepare for the meeting and have done it.

Visual Aids

Here the A3 report comes handy as a foundation for discussions.

Focus on Problem Solving

Sharing of information should already be done. The meeting and focus should be about rapidly exchanging opinions to solve a problem.

The meetings starts and ends in time

The colleagues time should be respected, and to reduce risk of attaining late on the next meeting, it should end at time.

(Liker, 2004)

4.4.2 Principle - 14 Continual Organizational Learning through Kaizen

Kaizen is usually described as continual learning. The core of kaizen is more of an attitude for all people, blue collar as well as white collar, for self reflection and a desire to improve. With their collective talents, they work together proactively to achieve regular, incremental improvements of the process. A learning organization does not only increase technical or developmental skills, the organization should also develop good skills on *how to learn*. Standardization is an important factor in being able to transfer new innovative ways of doing things from individuals or teams across the whole organisation. When a new improvement is made it should be implemented and set as a standard. This enables to find new improvements. (Liker, 2004)

With low employee turnover it is easier to achieve higher degrees of learning. This combined with stable work force and slow promotions keeps a good foundation of knowledge to learn from. Learning is about building on the

past and moving forward incrementally. High degrees of personnel turnover and/or low degrees of standardization can lead to old findings getting lost. This means employees must put time and effort into re-inventing methods that have already previously been implemented. (Liker, 2004)

Problems are opportunities for learning. Toyota uses a seven steps method for practical problem solving that include *go to see* (*genchi genbutsu*), *5 why* and *standardize*. These together create *kaizen*. (Liker, 2004)

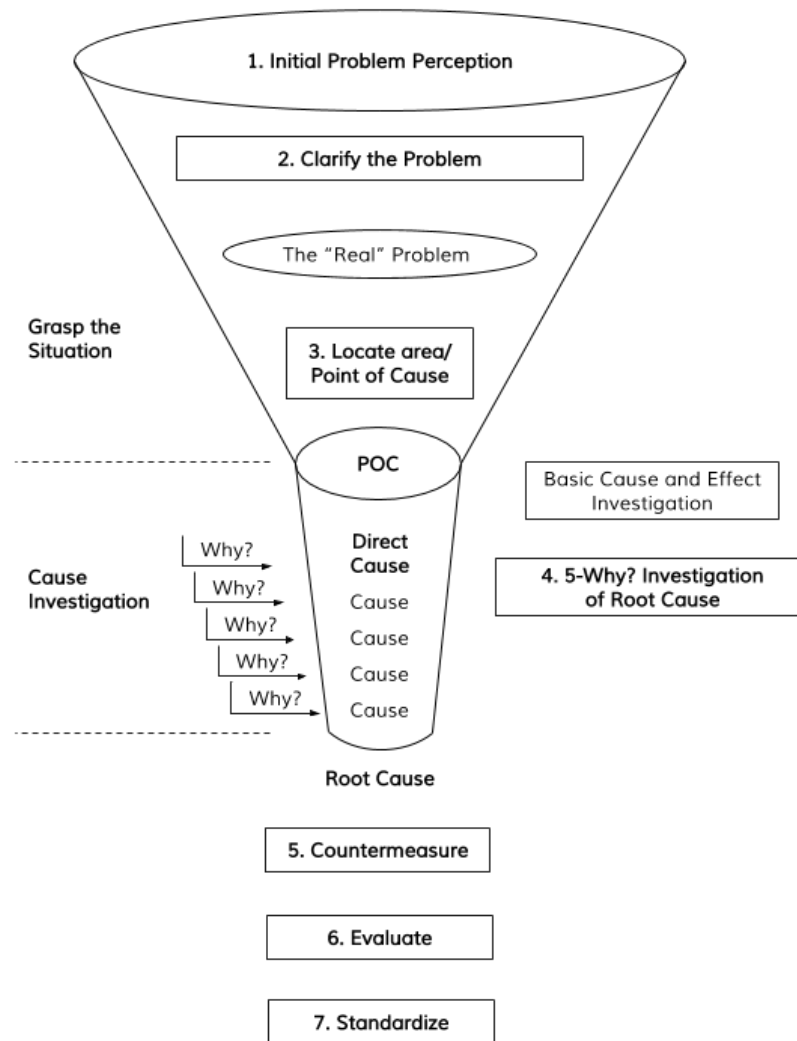


Figure 12 7 Step of Problem Solving, Inspiration from: (Liker, 2004)

The seven steps start by grasping what is the hardest part in the problem solving method. Below it is briefly described, step by step.

1. Initial Problem Perception

A problem can exist in three ways: anything that is a deviation from the standard, the gap between actual conditions and the desired condition, unfilled customer need.

2. Clarify the Problem

In order to clarify the problem you need an open mind and to get hands on experience for the next steps. One needs to break the problem down more specifically. These can be prioritized in a Pareto analysis.

3. Locate Area/Point of Cause

Set a target for improvements and try to find the point of cause (POC). eg. Where is the problem observed?

4. Investigation of Root Cause

By using *5-Why* to figure out the root cause and the factors that generated the problem in the first place, it could become evident that there is more than one root cause and all should be addressed.

5. Develop Countermeasures and execute them

When all root causes are found, countermeasures can be created for every single one of them with the aim of removing them. It can be good to execute the countermeasures one by one to be able to monitor the effectiveness of each.

6. Evaluate

Some countermeasure might fail, this is where the failures are analyzed and modified to get the intended result. There should always be room for improvements, with a good evaluation it's possible to find these improvements for the next problem that needs solving.

7. Standardize

A standard should be understood and set to make sure that what has been learned and the improvements are shared through the whole organisation. This is important so the learning don't get forgotten or become unavailable for further improvements.

(Holland, 2013)

Toyota introduced the lean concept at a time when they were opening facilities in other areas around the world, this caused the cultural difference between the countries to feel significant. One such area was *hansie*, which is translated as reflection, and is an essential part of kaizen. But *hansie* is a mindset and an attitude. In Japan this means that the person feels deeply sorry and admits shame when something goes wrong. They do this too make sure that the error won't occur again. In other cultures it can be harder to admit mistakes to that extent. When one does however, the process of reporting the mistake can create a lot of valuable learning tools for the company, as well, of course, as the mere results of the findings. (Liker, 2004)

The measure results and errors are important to be able to prioritize what actions to take and to what extent the improvements rendered positive results. They should all reflect the same common goal through organization vertically.

To be able to do that, it is important that the goals and measures reflect the objectives. Goals start at the executive level and each department or level develops measurable objectives designed to support the executive-level goal. The goals should not be contradictory between groups or departments in the company. (Liker, 2004) An example of this is when the overall goal is to achieve cost savings within production. The purchasing department therefore try to make savings on tools with less quality while production try to increase the quality and decrease scrap loss. To increase quality they might need increased quality of tools. The departments have clear contradictions.

The goals should be process-oriented, as this leads to positive results. Instead of being result-oriented, for example being focused more on the bottom-line. Being process-oriented typically involves having more patience, and more investment in people and the process, therefore leading to positive results. (Liker, 2004)

5 Lean Software Development

Lean software development focuses on how to administer high quality code to the market, in a quick way, with optimal solutions. Software developers work a lot with problem solving which is why their method can be inspirational for other areas of work. Lean software development has 7 principles as a foundation when working and making decisions.

1. **Eliminate Waste**
Spend time only on what adds real customer value
2. **Amplify Learning**
When you have tough problems, increase feedback.
3. **Decide as Late as Possible**
Keep your options open as long as it is practical, but no longer.
4. **Deliver as Fast as Possible.**
Deliver value to customers as soon as they ask for it.
5. **Empower the Team**
Let the people who add value use their full potential.
6. **Build Integrity In**
Don't try to tack on integrity after the fact - build it in.
7. **See the Whole**
Beware of the temptation to optimize parts at the expense of the whole.

5.1 Eliminating Waste and Differences Between Production and Software Development

Since lean has been so successful within production industries, it's started to spread among other industries. This chapter will explore how this can be translated to fit into software development.

Eliminating waste is a core function of lean, therefore it is necessary to start translating waste into something that makes sense for software developers. Waste is something that does not directly add value as perceived by the customer. A famous and successful method for coding, without giving any opinions, is the waterfall process. It is a linear process where the sequences could look something like following:

1. **Submit request**
2. **Project Approval**
3. **Gather Requirements**
4. **Customer Sign-Off**
5. **Analysis**
6. **Design**
7. **Design Review**
8. **Code**
9. **Test**
10. **Deploy**

(Lotz, 2013)

In this example all of the steps but analysis and coding are actually waste, even though they might be necessary, it does not create direct value to the customer. Translating waste from the production industry to the software development industry can be done according to the table below.

Production Wastes	Software Development Wastes
Overproduction	Extra features
Unnecessary Transport	Task Switching
Overprocessing or incorrect processing	Extra Processes
Excess Inventory	Partially Done Work
Unnecessary Movement	Unnecessary Movement
Defects	Defects
Waiting	Waiting
Unused Employee Creativity	Unused Employee Creativity

Table 1 Lean Production Versus Lean Software Development Wastes, Inspiration from: (Poppendieck & Poppendick, 2003).

Extra Features

When adding extra features into a system, the complexity of the system increases and it can be a potential failure point. Every time new code enters a system it has to be tracked, compiled, integrated and tested, and then it needs maintenance for life. If the code is not needed, it should be considered waste since no one asked for it and no one will use it.

Task Switching

Assigning developers to multiple projects can lead to waste. If a single developer's works on two projects simultaneously and each project take one week of man hours to finish, none of the projects are in production until two weeks later. If switching time is added, it will probably take even longer to finish. Each time a developer switches task there is significant time wasted as they need to get their thoughts together and get into flow of the new task.

Extra Processes

Extra processes can take developing time from the developer. As an example writing reports can be regarded as waste since it might not add value for the customer. If paperwork is done and no one is waiting to read it, there is high risk that it's waste. This can happen when developing for a customer during a change of demand, requiring new customer approved documentation. The waste risk grows higher when the safety-critical system requires documents for traceability.

Partially Done Work

Partially done work can become obsolete. This is due to the fact that the code system that it is intended to interact with might change. Partially done work can also get in the way of other developments that rely on it's completion to function.

Waiting

The developer can end up waiting for many reasons eg. waiting for requirements, aid from staff if the right type of staff is not available or busy in another project, waiting on feedback and reviews etc. The delay that might end up due to waiting leads to more time until customer realizes it's value, and is therefore waste.

Unnecessary Movement

Developers need concentration. When developers need to find an answer by walking around in the office it takes time. But the time to find an answer to the question might be less than the time to regain the focus.

Defects

Bad code is not necessarily defect or waste since it's a part of the process required to test the code. But the time until an error is found is waste. To decrease waste it is essential to find these errors as soon as they occur, test

them immediately, integrate often and release to production as soon as possible. (Poppendieck & Poppendick, 2003)

Unused Employee Creativity

When there are losses of human capital. This can happen when a company is not making use of employees ideas for improvement, their skills or not developing their knowledge. (The same as in Lean Production)

5.1.1 To Find Waste by Value Stream Mapping

A good way to see and analyze waste is value stream mapping, this is used in a variety of industries. It is a chart drawn with process break down from order to delivery, each part is quantified with time and decided if it's waste or value added time.

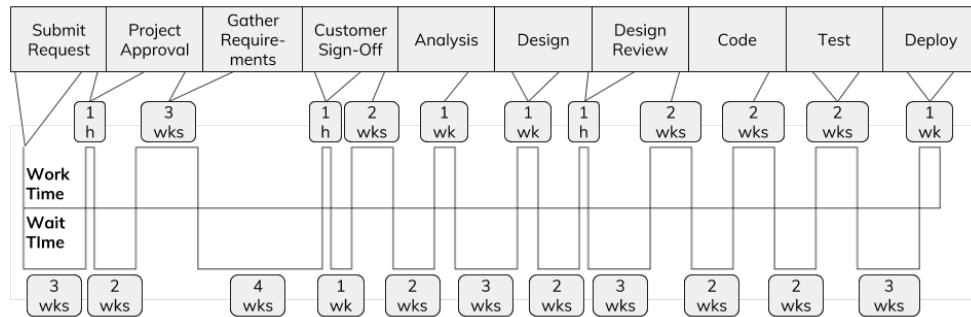


Figure 13 Value Stream Mapping, Inspiration from: (Poppendieck & Poppendick, 2003).

The diagram presents a value stream map for a waterfall project where there are handoffs between every step in the process. The waiting time reduces flow and are in this example a majority of the time in the project before the customer can get their product. The goals should therefore be to find better processes to reduce the waiting time. (Poppendieck & Poppendick, 2003)

5.2 Amplify Learning

Creating good software can't be done effectively with production processes, it is a development process. The differences are that in production, the processes should result in minimal variation, while development processes need high degrees of variation with trial and error. Each variation should give insights as to what the best result might be. Amplifying the learning can lead to better results. (Poppendieck & Poppendick, 2003)

5.2.1 Quality Differences Between Production and Development

One crucial difference between production and development is in the definition of quality. There are no clear definitions of quality and it can change from industry to industry, from company to company and also over time. They all aim for customer satisfaction by meeting or exceeding customers expectations, but how to achieve this varies. One general definition for quality in production can be; *Conformance to requirements specified in the design*. If the product varies, it won't be able to integrate with other parts or products in

the product-line. Customer expectation is only met when a customer receives exactly what is ordered. In the service industry, every customer has a different idea of what quality is (different expectations). Therefore, the quality is defined as such; *Adapting to meet the changing expectations of many different customers*. In the software industry quality should mean the product is easy-to-use and cost-effective. Thus, quality can be defined as, *fitness for use*. (Poppendieck & Poppendick, 2003)

In production variation from the standard is bad, in development and the service industry variation is necessary. In fact, in many cases it is sought after since the customer expectations can be shifting. For example, a theme park would not be very successful if it only had one attraction. To meet customer demand there should be many attractions. Similarly, the development phase produces solutions to unique customer needs and therefore the process can't be standardized. To solve complex problems, a similar scientific method discussed for lean startup can be used: observe the problem, create a hypothesis, experiment to test the hypothesis.

There are generally two ways to do software development. Either to try and get it right the first time, or to have quick feedback loops that stimulate learning and deeper understanding of the problem with rapid try-it, test-it, fix-it cycles. Often the problem is best understood by those in the business with the problem, so it's necessary to have them in the knowledge-generation loop. When working with quick cycles it also can increase learning. Longer cycles create large batches (many features are released at the same time), longer feedback loops and therefore give ineffective learning. (Poppendieck & Poppendick, 2003)

The essential differences between software development and production are concluded in the table below.

Lean Production	Lean Software Development
Quality is conformance to requirements	Quality is fitness for use
Variable results are bad	Variable results are good
Iteration generates waste (called rework)	Iterations generate value

Table 2 Differences between Lean Production and Lean Software Development (Poppendieck & Poppendick, 2003).

5.2.2 Feedback

When the details of a project are determined in the beginning, it is called a deterministic approach. Often in these cases the waterfall model is used, although this model does not usually provide much feedback. The waterfall

method generally has a plan of how and when things should be done. This plan can be hard to change if developers and managers/clients are not working closely together. Problems arise when the solution to the business case differs from what the manager expected in the beginning of the project. The developers should get feedback early, which should ultimately lead to a change of plan early in the process.

Software developers intention should always be for immediate customer product usage. The reason for this is so there is no delay on feedback. The testing could even be done at the customer site, so if there is a problem, it can be resolved directly. (Poppendieck & Poppendick, 2003)

5.2.3 Iterations.

Iterations are closely linked to one piece flow from the production view of lean. An iteration should produce a coherent set of features to a well understood short-term goal. These features are similar to a small batch. This way of working should generate quick feedback for the next iteration. It enforces quality and communication among the developers and makes it easier to allocate developers where they are most needed since the projects are short. Iterations also makes it possible to make decisions based on facts instead of forecasts. When one iteration is done, the group of developers can meet to get synchronized, make decisions and stay aligned with the interest of the customers and users. When the iterations are short and frequent it creates a motivational environment for the developers. They can feed on success and from the feedback the customers give. It give them a sense of accomplishment and become more motivated to satisfy the customer. (Poppendieck & Poppendick, 2003)

If the feedback loops are too long a situation called *Trashing to develop* might occur (not to be mixed-up with trashing on a computer). This means that the feedback is so fast that the first direction is changed before it's even finished. Therefore, one has to go in a totally new direction. (Poppendieck & Poppendick, 2003) Feedback cycles that are too short might increase trashing, so the feedback loops need to be balanced. The project should therefore be locked without input until the feature is done and only then be open for feedback.

One concern about iterations can be that there are no finishing point and clear end goal. Features don't necessarily converge into a solution since there are no stopping point, and it might feel like the development is never done. However, if the developers work on the top priority items first, it's most likely that the business value is optimized. If the scope is negotiable at every step along they way, other priorities can be made. At some time the negotiations about the scope will conclude. *Standish Group made a study* and found that 45 percent of features in a typical system were never used, and 19 percent rarely used. As discussed earlier, this is critical waste as time was spent on non value creating features, and could create legacy problems. When taking the highest priorities first and negotiating along the way, the non value adding features can

be minimized. Customers/clients tend to ask for as much as possible in the beginning of the project. At this time they know they have the developers attention and might think they just have one shot at getting what they want. (Poppendieck & Poppendick, 2003)

5.3 Making Decisions as Late as Possible.

When working with a deterministic approach, all the requirement and decisions are decided in the beginning of the process. Many people prefer this since it quickly reduces the complexity of the problem and there is a fixed design to work on early in the process. This can create costs later in the process, if for example problems appear that were not considered in the beginning. Large problems could also mean that a change of course is necessary and make all the previous work obsolete. When iterating the development with a broader range of solutions, new discoveries are found through the cycles. These problems could be solved when they appear and/or another solution could be chosen from the iterations feedback. (Poppendieck & Poppendick, 2003)

Whatever strategy to chosen for the problem solving process, the process still won't think for itself. It's important to develop staff into making wise decisions, also to empower them to be able to execute them. An example of this is the U.S marines. The marines are drilled for responding to chaotic circumstances where quick decisions have to be made with an open, innovative and knowledge oriented mind. The marines plan, but do not predict. Their plan is thorough and attempts to gather as much information as possible just hours before the mission. This results in them having the latest information and being able to cover the issues rapidly. When engaging in the mission, the organizational structure often collapses, so those in the front lines with the most current information are expected to make the decisions. There are no penalties if they make mistakes, but they believe that their mistakes in the front lines would be less severe than for the distant officers. (Poppendieck & Poppendick, 2003)

To make it easier for the employees to make decisions comfortably, simple rules can be set up to support them. This acts as a framework where the employee can work freely without having to ask the manager about every single detail. (Poppendieck & Poppendick, 2003)

5.4 Deliver as Fast as Possible

There are a couple of reasons why you should deliver as fast as possible.

Increases Customer Satisfaction

Take the service sector as an example, fast deliveries have been a key success factor for online stores. For the customers, it means that decisions can be made as late as possible and the satisfaction for the service is granted earlier. This applies to the

software development industry as well, since the customer receives the value from the software earlier.

WIP is Equal to Risk

The customer can change their mind when working on something, this could make the initial product requirement obsolete. Decreasing WIP by fast delivery therefore reduces this risk.

WIP can Increase Defects.

If the developer is releasing their product in a few large batches, many problems and bugs could be hidden which otherwise would be detected in a live test.

There are in general two ways of putting people to work. Either they receive their orders by a manager or leader, or the system allows them to figure it out by themselves. When the business demands a fast moving development, the latter is preferred. If the staff always have to be told what to do, information travels up and down the chain of command, which is extremely time consuming. This time can also decrease the workers motivation. To prevent this, the system must preempt the next step by visual control or management by sight. This can be done by using a pull system with visualizations as kanban. Kanban can also make it possible to deliver the features when they are needed and enables the Just-In-Time idea. Kanban in software development differs from production. Software development uses a kanban board where all the cards with requested features are posted in a to-do area. Then the developers estimate the time it will take and the customer prioritizes them. The developer can then pick cards to work on and the kanban board can be modified to visualize the process along the way. The work therefore becomes self-directing, pulled from the customers and visible for anyone who is interested in how the work is progressing. (Poppendieck & Poppendick, 2003)

To deliver quickly, you also want to decrease queuing. There are two ways to decrease queuing. The first is to look at how the work arrives, and the second is how the work is processed. Doctors make appointments so that they have a steady flow of people entering the clinic. Restaurants might have early bird tickets with lower prices. This creates incentives for customers to avoid the lunch rush. Both ways attempt to control how the work arrives. Allocating resources or simply working faster also decreases processing time. For developers, a way to achieve steady work flow is small batches produced by smaller units of workers. A good analogy for this is the line in the grocery store. With many cashiers the customers spread out. Then, if a customer needs more time, there is less effect on the queue. Small batches spread out the demand in a steady state of arrivals, therefore, the capacity of the system can be balanced to handle them. (Poppendieck & Poppendick, 2003)

5.5 Empower the Team

Moving the process of decision-making further down the hierarchy and making decisions closer to the customer could empower the workers. This was one of the important differences between Taylor's scientific management and the Toyota production system. The latter is described in *5.3.1 Principle 9 - grow leaders who thoroughly understand* with a bureaucratic and enabling system. Empowering the staff is closely linked to those workers finding more purpose and motivation. Workers need more than just a list of tasks. The work can be intrinsically motivating. This motivation comes from the work itself, pride in workmanship and a sense of helping the customer. The task should have compelling purpose, it should be achievable, challenging and closely linked to the customer. A highly motivated team does not need to be told what to do, they know from the principle of pull. (Poppendieck & Poppendick, 2003)

To stimulate intrinsic motivation four areas should be considered.

Belonging

In many complex processes, a team is needed to work efficiently. The team should have clear goals and be committed to its success. There should be honesty and respect between the members and the team should win or lose as a group. When rewarding one team-member but not the others, competition within the group could be fostered.

Safety

Allow mistakes to be made and don't encourage a zero defect mentality. When mistakes are not allowed to be made, it ruins the initiative that is required for success. Delegating responsibility requires tolerance, those who get the authority are going to want to do their jobs in their own way, and in long term, their mistakes will be lesser than those of the managers.

Competence

People need to believe they are capable of doing a good job. They want to work with something they believe will succeed and have a sense that they are a part of a winning team. Competence comes from knowledge and skill, positive feedback, high standards and meeting a difficult challenge. The leaders have to be supportive and make sure they are on the right path. They must also have sufficient equipment and the capabilities to do a good job.

Progress

This is another example of why quick feedback loops are good. Quick feedback loops give hands on results earlier and stimulates progress. People need to feel that they have accomplished something and their purpose must be reaffirmed. Celebration is a good way to ensure the accomplishment is not forgotten. The celebrations should be balanced according to the scope of the problem. Learning is a central part of lean software development. To achieve learning simply for material monetary means can be useful, but this

can affect the long sense of intrinsic motivation. The key is increasing the value of learning itself with intrinsic motivation.
(Poppendieck & Poppendick, 2003)

5.6 Build Integrity In

In the late 1980s, Kim Clark from Harvard Business School studied the automotive market and found key differences. Even though they were alike, these were related to why some companies made successful cars and others failed. An example of this is with the introduction of the four wheel drive with Honda and Mazda. They had the same functionality but Honda had great success and Mazda was somewhat of a failure. The differences were something that Clark called product integrity. Honda found that four wheel drive on a car with a sporty and progressive image, matched the consumers idea's about the technology. Mazda however released four wheel drive it in a car with designed for the needs of a family. This was a mismatch with the racing steering system. (Clark & Takahiro, 1990)

Product integrity has two dimension. Internal integrity refers to how well the product functions and if the structure works together. In practice, this means that the parts fit smoothly and that the available space is used to it's maximum. Internal integrity is achieved when the development works cross functionally and coordinates well within the company and with its suppliers. There are two ways to achieve internal integrity. One is using existing parts. While this removes a degree of freedom, it reduces the complexity for communications. The other is integrated problem solving to assure excellent technical information flow. Integrated problem solving means:

- Understanding the problem and solving the problem at the same time.
- Information flow is not delayed until all information is available.
- Information is transmitted frequently in small batches instead of at the same time in a large batch.
- Information flow both ways, it is two-directional.
- Information flow should be done with face-to-face communication, preferably not with documents.

(Clark & Takahiro, 1990)

External integrity refers to how well the products performance matches the customer's expectations. Companies with good external integrity consider the customer's values before technical staff make design decisions. The vision of external integrity should be clear and refreshed regularly so the tech staff don't get lost in the technical issues and forget the importance of customer value. Customers usually know what their problems are, but can't describe the solution, so they will know a good solution when they see it. This is another argument for why small batches and iterations with customer insight is important in the development process, and why the *throw it over the wall*

system might not work as well. With every requirement, hand-off information might get lost or misinterpreted. To get a well working customer-development-information flow, some techniques could be used:

- Smaller systems should be developed by teams that have direct contact with those who will judge the system. The iterations should be short and should show the results from each iteration to a broad range of people, making corrections according to their input.
- All people, customers as well as developers, should use the same language and vocabulary to communicate. They should also have models for communication that are easy enough for everyone to understand, especially for large systems. Models are key for creating abstractions and enabling communication on broader topics.
- Large systems could have a master developer, one that represents the customer's interests to the developers. This person should therefore have deep customer understanding as well as excellent technical skills.

(Clark & Takahiro, 1990)

5.7 See The Whole

A system can be described as interdependent and interacting parts that are joined by a purpose. How well the system works is not just the sum of the parts and how good they work individually, but the result of their interactions.

A sub optimization can occur when the sights of the system are too narrow. If the individual parts are considered one by one, there becomes a lack of understanding of the system as a whole. This might occur when targets or measurements are set for different departments that are not aligned. (Poppendieck & Poppendick, 2003) If the software developers have a goal to produce code with minimum defects and testers of the code have a goal to make as many tests as possible. The developers might work to get it right the first time thus leading to larger batches. The testers would have larger batches to test which takes more time and lowers their achievements. If this would be the case the department's goals are incompatible and don't strive for a mutual goal. If the goal however is to avoid defects in production for both teams, there would be more incentive to collaborate.

When measuring workers they try to optimize their performance that they are measured against. If not all their work is measured sub optimized is behavior is likely to appear since workers tends to prioritize their work only to improve their numbers, not what create most value. The rule is *you get what you measure*. So if you can't measure everything that is important, it's better not to measure at all, so you avoid a sub optimized behaviour. (Poppendieck & Poppendick, 2003)

Instead of measuring the number of defects or faults per individual, it should be done per feature. This is due the fact that not many defects are made by the staff themselves, but can be traced to the system or the

procedures the workers work within. In the quality movement in the 1980's, only 20 percent of the defects were within the workers control, the remainder was in the systems under the manager's control. When looking for defects at level of the individual, the focus shifts from the root cause (the system) to smaller, surface issues (the individuals). (Poppendieck & Poppendick, 2003)

Maciej Zaremba wrote a series of reports in the Swedish news magazine Dagens Nyheter about the New Public Management (NPM). He describes the consequences of what happens when healthcare puts a price-list on different treatments. It also discusses the effects that bonus systems give when they used KPI's for waiting times etc. The system for setting price-lists on treatments was so complex that not even the doctors understood it. Hospitals earned more money if they employed a programmer who could manage priorities. With the right priorities according to the price-lists the hospitals earned more money. On the other hand, the hospitals that made priorities according to what the nurses and doctors found most important and didn't followed the price-list got less resources. Since the price-list couldn't take account to all the real life circumstances the doctors argued that a sub optimized behavior was enforced. (Zaremba, Kultur & Nöje, 2013)

Another KPI that gave bonus to the hospitals was waiting time. Patients were not aloud to wait more than a certain amount of time before receiving healthcare. That meant, as the reporter says

“...a person from the eighties with worries about wax in the ear when going abroad had the same priority as a person with heart failure”. (Zaremba, 2013)

Since the hospitals were rewarded with more resources depending on the KPI's, they found a way to work around this. Overall, this made the quality of the experience lower, but the measured waiting time was reduced. Waiting time is the time the patient goes through the hospital doors until the time the patient gets help from a nurse or a doctor. In this case, patients had to wait in the ambulance when they knew that they wouldn't meet the time requirements. When they knew they would meet the time requirements the patient was welcomed to the waiting room and the ambulance could get back to work. (Zaremba, 2013)

NPM was also used in Germany, and provides another example. They managed to decrease the average health care waiting time from 9,7 days to 7,8 days. The doctors use what they call *“Blutige Entlassung”* to achieve these numbers. Translated this means bloody dismissal (or early discharge). The doctors let the patients leave earlier than they should, with the hope that home care would take care of the wounds. In practice, this meant that the patience often came back again, lack of treatment for the wounds caused them to worsen. (Zaremba, 2013)

6 Areas of Research and Methodology

Some insight into Lean startup can be made by looking at the history of how Telavox was initially managed. To get an understanding on how lean production and lean software development principles can be implemented, some experiments to implement lean principles were evaluated. The experiments was directed at what seemed suitable for Telavox's processes at the time. No regard was made for what the best method in implementing a lean mindset would be. The experiments was made on the department for customer support, and was implemented by the author functioning as a supporting manager. The experiments took place before and during a large organisational change within the customer support organisation. The customer support org. contained four different departments; Technical Support, Economy, Delivery and Customer Care. These was merged into one department named Advisors. The concept with the Advisors was that all customers could have their own personal contact person. This person would take care of all the issues that the old departments covered previously. The following chapter will describe the experiments as:

1. Calculating waste
2. Supporting decisions with a lean tool - 5 Why
3. Continual improvements with stand-up meetings
4. Conclusion - Why developing lean at a service organisation might fail
5. Conclusions - How to developing lean at a service organisation

6.1 Calculating Waste

A small project group from the customer care department received a mission to make *cold phone calls*, which is a saying for a phone call made without preparing the customer. This project was put together for the release of a new platform. The platform intended to be more user friendly, the services cheaper and it would have better flexibility. The callers goal was to convert the customer to the new platform without any new subscription bound time, and with no ulterior motives. Since they had the customer on the phone they were also supposed to conduct a survey. This was to make sure that the customer was pleased with Telavox as a contractor of communication solutions. The idea of converting customers was to ensure that the customer had the right pricing and a good value offer. This is important if competitors would try to engage them. The target of converting the customer was set relatively high, suspecting that no one was going to decline such an exceptional offer. However, it was not successful. With an average number of 45.5 phone calls a day, less than 0.5 were converted to the new platform. During discussions for improvement with the staff callers, it became evident that very few phone calls actually reached the right person. With this background information, a method to try and figure out a better way of making outbound phone calls was analyzed.

6.1.1 Time Wasted on Outgoing Phone Calls

When working with customer care, there are tasks where the employee is obliged to make outgoing phone calls. The aim in this case was to see what portion of the employee's work time was regarded as waste. To be able to do that, a definition had to be made regarding what kind of phone calls were value-adding and non-value adding (waste). The definition of value adding time was; *sharing information so that the customer could get a better experience from Telavox or if Telavox got information to be able and help to improve the customer's business*. This meant that all incoming phone calls were regarded as value adding since the customer needed to share information to receive help and have a better experience. The definition of outgoing phone calls that were non-value adding was:

- The intended recipient wasn't available and someone else answered.
- The person receiving the call didn't have time to talk and the caller had to call again another day.
- The recipient didn't answer the phone call and the caller hung up.
- The phone call was redirected to voicemail.

To measure the waste of outgoing phone calls, two employees from three different departments were analyzed: Economy, Delivery and a team from Customer Care. They were chosen so that there was a range of different people with different working backgrounds, and the different departments had a widespread set of reasons to administer outgoing phone calls.

Economy handled issues regarding

- Sending invoice reminders
- Escalations to debt collection agencies
- Handling errands regarding bankruptcies
- Transfers of subscriptions
- Invoice corrections
- Internal support by answering questions

Delivery make sure all the companies IT-solutions gets operational by

- Setting up Private Branch Exchange (PBX) services
- Porting numbers
- Information gathering about the customer
- Make all the settings of services using the information from the customer
- Quality assurance

The team from Customer Care did a temporary project involving:

- Making proactive phone calls to customers and making sure they were pleased with their solutions and if not, solving the customer's issues.
- Market surveys with the aim to achieve a Net Promoter Score (NPS).
- Giving customers an offer for a new platform and lower pricing.

Personnel from the different departments were asked what the minimal time was to have a value adding phone call using earlier mentioned definition. The time was concluded and averaged to be around 60 seconds for all departments.

Data from the phone calls was collected from the Telavox database with information on the following:

- Length of the phone calls.
- The type of the phone calls in three categories. Missed, outbound - or inbound phone calls.
- Date.

The data was categorized by length in groups of 0 seconds (unanswered), and ranging from 30, 60, 90, 120, 180, 240, 360, 480, 600 to 600+ seconds. In these categories the number of phone calls were counted and their average length summarized. Since data received on the length of the the phone call didn't include the waiting time for the caller, an average waiting time had to be compiled.

6.1.2 Average Waiting Time for Caller

For every phone call it takes some time to be routed through servers and telecom operators. When the phone call is connected to the receiver, it takes some time for the receiver to pick up. This time was added to the minimum of 60 seconds, the amount of seconds the employees figured was enough to be able to deliver a value-adding phone call. To measure the pick-up time, data from 3394 phone calls, from random customers, was withdrawn from Telavox's internal log. The data consisted of connected time and durational time of the phone call. Duration time is the time from routing to Telavox to the end of the phone call. Connected time is the time between when the phone call was answered and when it ended. When connected time was subtracted from the duration time, the requested data is compiled - the waiting time for the telephonist.

If the customer doesn't answer the phone at all, this is also considered waste. The average waiting time it took for the caller to hang up was determined with interviews with the callers from customer support. It was determined that the average time was about 20 seconds.

When the average waiting time was added to the length of the phone call after it had been categorized, it was possible to analyze the waste time of phone calls from the different departments. To see the extent of waste comparing to the total working time, the personnel working hours were calculated for the same period they could be working actively with phone calls.

6.2 Support Decisions with Lean Tools - 5 Why

When there was an organizational change to create the advisor department, three team leaders were internally recruited. All of them had worked in different departments (Technical Support, Customer Care and Delivery). Before the organizational change they were working as specialists. The Advisor department was supposed to manage all three roles and the employees would assume a more general role. This meant knowledge was diffused between the specialist. A head of advisors was recruited externally with no knowledge in any area, the head of advisor was supposed to support the team leaders, follow up goals and make sure the department was running smoothly.

To help the head of advisors, a supporting manager was available with deeper understanding on earlier processes, the working culture and the people in the organisation. A framework for the Advisor organisation was created with vague goals and some values beforehand. These were partly supplied by the supporting manager. The team leaders and the head of advisors were about to develop more tangible goals, more clear values and a new working processes. Naturally many questions and problems arose, these couldn't be anticipated. The supporting manager used the 5-why system to find the root cause. The head of advisors used a version of 5-why by *boomeranging* the question (sending it back to the questionnaire). The basic concept about 5-why was presented to all of the team leaders. This was to ensure that they knew the supporting manager was going to use it, and that even small issues were not about to be taken lightly. The 5-why system was mostly used as a tool for coaching so that the team leaders would find the root cause of their problems. In the longer term, it was thought that the team leaders would start to use it by themselves to solve their issues.

A couple of month after they had been working with 5-why, the team leaders were interviewed to see how they liked being confronted with 5-why and boomerang questions.

6.3 Continual Improvements with Stand-up Meetings

Stand-up meetings were implemented to support the organisational change. The stand up meeting was once a week in front of a white board for all the team leaders. The white board contained fixed subjects to attend to. The board was located in a central area close to each team and clearly visual. The board intended to be updated regularly as ideas came up. The subjects on the board were:

Top of Mind

Top of mind could also be called a focus area. It was meant to be a few areas that needed the team's attention. It could be to figure out solutions to problems or develop processes to further increase efficiency.

Hidden Failure

This was to enhance transparency through the organisation. If someone did a mistake they were supposed to write it down so no one else would repeat it. It could also anticipate the need for further improvements to reduce risk of human errors.

Weekly Tasks

The idea was to set clear goals on what was to be achieved every week. The team leader was supposed to set the goal with support of the manager. There was a follow up on the previous week's goals and new goals were set. This was supposed to make the priorities in the team and between the teams clearer.

Happynings

A wordplay on fun things that would happen during the week. It could be after works, birthdays etc. The goal was to always have something fun happening each week, if there were none, something should be organised. The idea was to make sure that everyone was leaving the meeting in a good mood.

The stand-up meetings with the subjects top of mind, hidden failures, weekly tasks and happynings were tested for a period of six weeks. Afterwards, when the head of advisors started it was reduced to just the subject - weekly tasks. The stand up meetings were reviewed with interviews.

6.4 Why Developing Lean at a Service Organisation Might Fail

A dependency model was created to see how the lean principles from Toyota related to each other. The results proved that they had strong internal dependencies and it was therefore hard to create a step by step handbook to achieve lean. Therefore, the plan to figure out how to achieve lean in a service organisation had to make a pivot. The new idea was to use the results from the other attempts as a foundation to create a model of how to achieve lean.

6.5 Conclusion - How to Developing Lean at a Service Organisation

The conclusion of this master's thesis will examine the company and its products/service life cycle with different aspects of lean. It looks into how it can be used for effective innovation or efficient production. It then explores whether or not companies that begin as startups can use theories from lean startup. It will examine if existing companies should use startup principles to enhance what will be named *internal disruptive innovation*. Over time, as a

company grows, it might find theories of lean software development and/or lean productions useful, an idea also explored further. This type of development is called incremental innovation.

These ideas about product's life cycles were examined with theories about the Innovation S-curve and diffusion of innovation. To get a better understanding of the startup years from Telavox, one of the co-founders was interviewed. The intent of the interview was to see similarities and differences between Telavox as a successful startup, and lean startup principles which work in theory. This can work as insight or inspiration for other companies that intends to create a startup. In an existing enterprise using methods to create internal disruptive innovation, a proposition was made to create a division called the sandbox. Examinations were implemented based on literature studies, and some inspiration came from other companies on what types of capabilities a sandbox organisation should have.

With the analysis of lean theories and with the failed tests to implement lean tools at Telavox, it was stated that communication was one of the fundamental areas of research. Communication was believed to enable the company to achieve better innovation in terms of quality, effectiveness and efficiency. Good communication was also stated to be a core principle in enabling organisational changes. With the life cycle of Telavox, so far it was possible to create a communication model that was tested and evaluated in Telavox processes. To create the model, the different aspects was analyzed.

- The challenges with efficiency losses due to a company's growth and the more complex communication channels.
- Communication waste at Telavox
- Waste at Telavox according to lean development principles

The model took inspiration from one piece flow and kanban. It is therefore called communication flow kanban. It is explained as a 9 step process. It argues why the model can solve many of the communication issues a company can experience. This is done during each step, and explains what wastes it can reduce and what lean production principles and lean development principles support each step.

7 Results

7.1 Calculating Waste Time

The results from the phone call analyze on non value adding time was as following:

Delivery

- Waste time on outgoing phone calls - 7.52%
- Number of waste phone calls - 56%

Team customer Care

- Waste time on outgoing phone calls - 18.42%
- Number of waste phone calls - 69%

Economy

- Waste time on outgoing phone calls - 62.17%
- Number of waste phone calls - 61%

Overall

- Waste time on outgoing phone calls - 11.17%
- Number of waste phone calls - 61%

The caller showed frustration at how few customers seemed able to talk. However the calculations proved that the amount of time spent on trying to reach the customer was 18.42% of the total time trying to reach the customers. The project group from customer care have a higher degree of wasted phone calls than the other departments. The reason for this is likely to be because they were cold phone calls. Economy had a high degree of waste time on phone calls. The data indicates (see appendix for detailed data) that the amount of time spent on longer phone calls was significantly shorter for Economy compared to the other departments. In general, their errands are shorter. This was confirmed by the personnel from Economy. The amount of total waste time spent by making outgoing phone calls relative to all working time was less than 1%.

7.1.1 Wasting Time on Data

The data was withdrawn to get tangible results to be able to improve the efficiency on the work when doing outgoing phone calls. However, no real plans for improvements were found with help from the data. When the data was finally compiled, the reorganisation was set in motion and with that the project was relocated to a whole new Advisor department. The strategy to give customers the offer had made a pivot. Instead of a project group, their personal advisor would convert the customer to the new platform instead. They would do so after they had established a more personal relationship with the customer and established a deeper trust. The time to withdraw the raw data and to process it took a week. The paradox is that the entire amount is considered

waste in the sense of creating customer value and a better working environment for the caller. But from an analytical standpoint there are some things to gain/learn, such as:

- The number of people picking up the phone is low so there might be better methods to contact the customers beforehand, ie. with text messages or email.
- The amount of time spent contacting people was relatively low, but is demanding on the caller since it's frustrating to get a low percentage of successful phone calls.

According to lean principles (see See the Whole - Lean Development), the main failure of the calculation of waste was that it tracks the worker's duties, but had little or no effect on customer satisfaction. In this case the workers could do nothing to affect the results and should therefore not be measured in that way. If they were measured with KPI's such as number of successful phone calls there could be a risk that there would be sub-optimizations. To increase the results and achieve higher KPI without increasing the value for the customer the caller could have:

- Let the phone ring until it connected to voicemail. This way it is logged as an answered phone call.
- Call the main number to the company instead of the responsible person and ask to be connected. Those phone calls are also logged as an answered phone call.

These methods would be less effective, but would show greater results.

When the project started, the supporting the manager did make some phone calls to establish routines for the personnel. Except the routines the phone calls done by the manager aided in the decision to not continue the project any further, but scale it with the advisor project instead. The supporting manager believed that the customers unwillingness to take the offer and convert to the new platform was due to lack of trust of someone giving them the offer on the phone. When the personnel making the phone calls later confirmed that it was hard to get the customer to accept the offer, it was easier to quickly change the strategy. This was an example of Genchi Genbutsu - go and see for yourself. If the supporting manager wouldn't have hands on experience of contacting customers there might be a risk that the competence of the personnel was the blame for the low convert rate. Not the process of contacting them by phone.

Despite everything, an understanding on how to calculate waste for a service organisation with callers was established. This type of waste calculation didn't prove to have any true value for either the company or the customers. It was an typical example when a white collar starts to try to create lean in an organisation and fail. Just as the supporting manager started in this project, by trying to get tangible KPI by putting quantitative data on the working processes. And with the data later see areas of improvements and progress by

doing follow ups on the data. However, the data in this case won't change the way people answers their phone, and is not a good measurement since employees are not able to affect the outcome

7.2 Support Decisions with Lean Tools - 5 Why

5 why functioned fairly well as a managing tool used by the supporting manager to find the root cause of the problems. But the results from the interviews provided information that the awareness among the team leaders of 5 why was low. This implies that the team leaders had difficulty to truly understand the concept of 5 why. Even though they spent a couple of months of going through the concept and frequently using it, they didn't recognize why it was used. The feedback showed it was frustrating to be confronted with 5 why or boomeranged questions. One of the team leaders mentioned that when they had a question, it was because they didn't know the answer. Just to get the question boomeranged back, didn't help. This can however indicate why 5 why can be successful. When someone asked "*There is a problem, how should I solve this?*" and gets boomeranged back with "*how do you think you can solve it?*" frustration arises. If the manager instead would have responded with "*let's sit down and do a 5 why analysis, why is there a problem...?*" it creates means for discussion and works better for coaching.

Due to the radical organisational changes the team leaders were working under high stress. In the interviews all of them stated that when they just wanted a quick response, it was because they did not have time for a full analysis.

7.3 Continual Improvements with Stand-up Meetings

Team leaders often showed up late for the meetings and the whiteboard didn't receive a single update with information about hidden failures. This was taken as a sign that the board didn't work as intended and was not appreciated as a tool aiding the changing progress. When the head of advisors later decided to change the board to just contain weekly tasks, the supporting manager didn't see any purpose in questioning the decision.

Even then, the people tended to be late for the meetings and the mood at the meetings was not very positive. The team leaders said in the interviews that the tasks gave more pressure than motivation. Since the team leaders were stressed, they didn't take time to figure out reasonable goals, instead they came up with them on the spot. This could be one reason why the goals weren't taken seriously and since they weren't, they didn't reach them. When next week's stand-up meeting came, they spent time defending themselves as to why they didn't reach their goals, and the mood at the meeting dropped lower. A downward spiral was created. After a couple of stand up meetings a suggestion came up that the meetings should be one by one with the head of

advisor. They felt it took their time to listen to other team leaders defending themselves. Another reason to suggest one by one meetings could be that they didn't like their colleagues think they did a bad job. The suggestion of taking the meeting one by one was however neglected since one of the main purpose with the meetings was to increase horizontal communication.

The stand-up meetings are still continuing today and it's begun to work better. To address the problem of not reaching the goals, the number of goals have been reduced. This supports what is later discussed that fewer changes at the time increase the rate changes can be made. The interviews also revealed that when the team leaders were reminded what it used to include, that they saw the value in it and would likely implement it again. It seems like the supporting manager gave up the project to easily. About the rejected suggestion of having meetings one by one with the head of advisors, they did start to see value in having them together and share informations and goals. This is probably correlated with them being able to reach their goals.

7.4 Why Developing Lean at a Service Organisation Might Fail

To understand how to best implement lean in a service organisation it might be good to understand how and why company fails to achieve lean. Many companies have tried to achieve lean but came out short and failed. Some projects might have been successful with waste reduction achieved by the use of some tools. However, what wasn't achieved was for lean to be implemented down to it's roots, to interlock lean principles in the company's culture and consistently be lean in all areas. Jeffrey K. Liker argues that it might be because the companies focused too much on tools like 5S and just-in-time. It might also be that management is not involved in the day-to-day operations and continuous improvement in projects fails to be achieved.

7.4.1 How Lean Production Principles are Correlated

When analyzing all the principles one-by-one and mapping how they are correlated, one finds that trying to implement lean with half-hearted attempts, only using some of the principles involved, simply will not work. Figure 14 is a model created by the author to show how the lean principles correlate based on the Toyota's way to be lean, the model is named the correlation model. Each principle has a number and the correlation between them is shown with a line. A short explanation of how they are correlated is explained below. To understand the explanations the reader might need some insight in the lean production principles explained in section 4. The correlations are not based on hard facts. Lean principles can be a part of a company's culture and values so some of the correlations could be questioned as well as some more links could be needed. The model is more a visualization that shows how lean is hard to implement, step-by-step, taking one principle at a time.

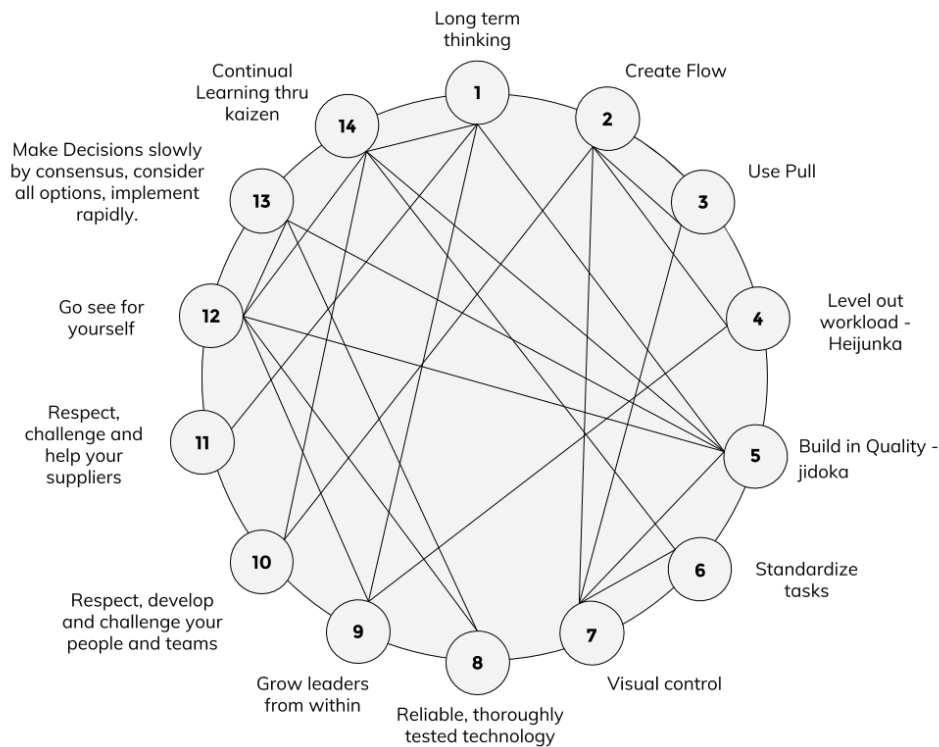


Figure 14 Correlation of lean production principles

1-9: Long term thinking and clear values can make employees stay longer since they are serving a higher purpose. In order to develop people and grow leaders within, it is necessary not to have a high staff turnover.

1-11: In the long term it is better to educate the suppliers so that they can produce at a lower cost. This is more successful than exposing them to a price war and in that way creates short term cost savings.

1-14: Long term thinking and clear values can make employees stay longer since they are serving a higher purpose. Low employee turnover increases the possibilities to see and understand improvements.

2-3: When using pull, one piece flow with short lead times is necessary. Also if products vary and there are low inventories. With one piece flow the first product can be delivered sooner.

2-4: Using flow highlights problems with bottlenecks. The bottlenecks can be areas where the environment is stressful and the staff are prone to being behind - Mura. Bottlenecks could also be areas where the staff and equipment are overburden - Muri. Flow can decrease waste such as defects since faulty products are found earlier - Muda

2-7: Visual controls like 5S can help achieve one piece flow, one concept in flow is the personnel completing more than one task in different working stations. If tools were misplaced in the working station by a co-worker it can take valuable time and reduce the benefits of one-piece flow.

2-10: Communication between team members is important. This is so that all members can learn the job in one cell and not be dependent on any one

person. The tight coordination between each step in the flow process and constant coordination helps build efficient work teams.

3-7: Using kanban is a means of visual control that can help to accomplish a good pull system.

4-9: Growing leaders/managers from within can increase heijunka. When a new leader is appointed from within, the leader should know the ways of the company and understand the culture. This means the leader will be much more likely to walk on the same path as the previous leader, without making large pivots that would require major changes in the organisation. That could cause large variations of working processes and profoundly affect the workload.

5-7: Building quality within should mean that the system be allowed to halt so that a problem can be found. When the system stops, it's a visual indication that something has to be done.

5-12: When the production line stops, it's possible for managers to see what the problems are directly, instead of receiving a report with daily incidents where information might get lost.

5-13: Considering more than one option in the beginning might seem time consuming, for example with set-based concurrent engineering. In the end though, it will likely be a better decision that saves time.

5-14: Bringing problems to the surface with concepts like andon or stopping the system before a problem occurs, are ways to make sure that the problems are noticed and fixed and there is constant improvement with Kaizen.

6-7: Standardizing tasks is a visual control to easily spot abnormalities

6-14: Individual and team innovation should be spreads through the organisation into organization-wide learning. The new way working and the insights can be transferred with standardization and practiced across the organisation until a new and a better way is discovered.

8-12: All new technology must be tested with a cross function of people that it might concern so that they can see directly if it works as promised. It shouldn't be bought or put into practice on promises or feelings that it would work as expected.

8-13: Testing should be thorough and not rushed. Once it is tested, it should be implemented rapidly.

9-12: Leaders who are grown from within have seen a lot of the company's processes first hand.

10-14: In Toyota, the team leaders on the shop floor are working with problem solving and implementing continuous improvements on behalf of the group.

12-13: Take time to make the right decisions. While it might feel quicker to just act on what feels right, facts and good decisions need to be set by seeing for yourself.

12-14: When going to see it by yourself increase insights and learnings to enable continual improvements.

7.4.2 Culture

One problem that occurs when implementing lean is that some of the principles are difficult to translate and be understood in the western world. Words have different meaning in different cultures. For example, *nemawashi* means the use of an informal meetings to lay a foundation prior a meeting where decisions are made. In Japan, if the leaders are not engaged prior to meetings and are faced with new information at the meeting, they will likely be offended. In the western world, *nemawashi* could also be used as a means to reach a consensus. However, if western managers did not receive prior information, they would be less likely to take offence, therefore this rule is harder to implement. In some cases the westerner might even be offended if someone takes their time prior to a meeting where the matter it is to be discussed anyway.

The culture of the company could also inhibit lean principles. For example, the Lean principle - kaizen need the workers to be empowered. It relies on a supportive culture so that the workers comes up with suggestions that are then implemented by managers or workers themselves. If the workers don't come up with ideas, this concept is impossible to implement. The culture should allow mistakes to be made so that they can be solved and prevented in the future. If the culture make the workers hide the problems there a no change that someone could find solutions for the problems. To give an example of this, a case involving the cultural environment on an oil rig in America is presented.

The case is about an oil rig in the Gulf of Mexico - Ursa, which when it was built, was one of the biggest in the world. Before building, personnel had to take personal development sessions. The idea with these sessions was to break down the male dominated work team's macho culture and see if this had an affect on the high accident rate.

In the 90's and prior, it wasn't unusual to see colleagues die. One of the guys working on Ursa had earlier seen seven people die on the job. Usually, after an accident leading to death, the colleagues had fifteen minutes of mourning before they were expected back to work. As mentioned, it was a macho culture on the oil rig, the workers didn't want to appear weak. They would hide their mistakes, avoid any evidence of vulnerability and refrain from asking too many questions. A drastic organizational change was needed to set the oil rig into proper functionality. Partly because of the complexity and the sheer population working there, but especially to provide a safer working environment. They started working on a new organizational structure in the early nineties to reduce injuries called Safety 2000. (Ely & Meyerson, 2010)

A woman working for a leadership consulting firm was responsible for a leadership program to change the culture. To achieve higher degree of safety they had to admit mistakes, be open for learning and ask for help. She had long frequent sessions with all employees where they learnt to express their feelings, talk about them and exercises to start and trust each others. The methods were included discussions about the worker's private life, talks about their values

and massaging each others. This succeeded to break the macho culture. The result was 84% decline in accidents. But the result was not just less accidents, the production efficiency and quality exceeded the industry's benchmark as well. This was due to that people were more open with each other not just on expressing feelings, information, and technical information started to flow in the company. The earlier macho culture of being tough, emotionally detached and technical infallible prohibit an learning environment. (Rosin, 2016)

When the culture was set on this oil rig newcomers quickly could get into the new culture. The quote below is an example how a manager would act on newcomers on the oilrig described from the manager.

I said, "Explain to me why your [procedure form] says you're supposed to wear a hard hat and face shield, and you don't even have safety glasses on."

And the newcomer answered, "Well, they're dark. I don't have clear ones."

I said, "I'll get you clear ones. And how come you weren't tied off? I watched you for several minutes, and you were not tied off. I don't want you to get hurt. You got friends and family at home. I want you to go home just like you came out here. Not go home with something in your eye, with a knot on your head. Not go home where you fell off of here and broke your back or your neck or worse."

He said, "You're right. This is my first day out here. And at [company X], they don't do this."

And I said, "You're not at [company X]. Forget everything you know about where you came from. You're here now." (Ely & Meyerson, 2010, p. 16)

This is an example of when the culture sticks. Even though the workers were experienced, the new workplace set new standards. Some terminology involved with the lean theory can't be directly translated, because they are so connected to Japanese culture. However, the oil rig is an example of how culture can affect the core principles of a work place. Lean principles enables companies to detect problems and find solution but the culture is an essential part in making it work.

8 How to Achieve Lean by Creating Communication Flow and Keep the Capabilities from Lean Startup

The initial goal of the master thesis was to show a timeline/growthline and try to find the best order of implementing one principle at the time to achieve lean. The correlation model does prove this hard, if not impossible. However, when diving deeper into the literature, there are some major aspects that are necessary to build a lean company. Combining the different aspects of lean as lean development, lean startup and lean production and with the cases and experiences learned from Telavox a model can support important areas of organisational development towards a lean company.

8.1 Meta principles - How to Develop Lean Principles in a Service Organisation

The chapter that follows will explain some areas and insights of how to achieve lean. It begins with some new principles to support implementing lean principles. To easier understand and differentiate them from the other principles, a new word needs to be presented - meta-principles. Meta-principles don't necessary have to be lean themselves, but are examined areas that can be a way to create lean. Some are in fact lean principles themselves, which is natural since lean principles help to create change. The meta-principles are the foundation the enables other models or theories to work.

8.1.1 Consistency

Months after abandoning the trials which attempted to implement lean tools (5s, create a kanban board, 5-why), the personnel actually saw the values in the methods and would have liked to continue working with them. The experiments with lean tools at Telavox seemed to have little effect. Bit by bit they were slowly forgotten until the personnel was reminded of them in much later review. The lack of consistency caused the project to fail. This was partly due to a lack of effort and belief from the supporting manager.

However, there need to be a balance being consistent and drop project for organisational change. If changes clearly makes the working processes less efficient it should abandoned. That was the reason why the failed experiments on Telavox were abandoned in the first place. The manager thought they were leading to nothing and just took people's time. It is also one of the lean startup principles to be able to try and abandon and change if it didn't work out. With that in mind when being consistent, you should be so by not try half-hearted but truly get all people in the project dedicated to see how it goes on a relative short timeframe, and when it's done evaluate if it worked or not and if it did continue. To reduce risk of creating wastes in the organisation, but still enforce changes the mantra should be - drive change with consistency and fail consciously.

8.1.2 Pull Instead of Push - Empowering

If the manager dictates change without explaining root cause/why it's needed, it might be met with some resistance. The relevant staff need to see it's value, and feel excited by the opportunities that comes with it. Lean principles therefore shouldn't be implemented from senior management and pushed down the organisation. It should pulled from the employees, who, with support and aid from the management, should understand the principles. Since most of the incremental improvement can be achieved by the workers with good insight into daily operations, a model to create change should support their empowerment.

8.1.3 Spare Time for Change and Make Changes at the Right Time

Timing can affect the development of lean principles. At Telavox for example, the timing involved in using lean tools was unsuccessful. These errors occurred when the company was undergoing major organisational changes. It was believed the right time to create a new process, routine and culture was when there was none, instead of having to redirect conditions that already existed. However, both team leaders and personnel were extremely stressed and had to undergo “*fire fighting*” and work reactively. They did not have time to think about proactive measures that would ensure it didn’t happen again. Interviews with the team leaders proved that it was too much to take on at the time. The pressure on them was high since a lot had been invested in the re-organisation.

While timing is important, time pressure could be another factor when introducing lean in service companies. If all is well and results are good, there is less initiative and inspiration to create change. If the results are bad and the company is having problems, there is much more sense of urgency and initiative. Change requires effort and time but reduced waste won’t happen until time is first invested. Personnel in companies with problems are probably already under pressure. The conclusion is that introducing lean should be done proactively, when the company is in a period of balance and high function. Time should be spared before hand to allow the change to be achieved in a non stressed environment.

8.1.4 One Change at the Time

With inspiration from one piece flow and the results from interviews with the team leaders it became apparent that it is better to finish developing one process at the time. Without change there is no organisational development, and change can be demanding for workers and organisations. The employees at Telavox realised that it was the extent to which they changed processes, routines, working tools etc that was the main problem, not lack of faith for the new way, or lack of involvement. All the simultaneous changes created many unnecessary meetings, documents and routines which were communicated but never implemented. If the change had been better structured and occurred systematically, one by one, waste would have been prevented.

8.2 The Model

To develop a lean organisation, the focus should be on how to create good communication channels supported by lean theories. At the same time these can be expanded with the company’s growth or partially implemented one department at the time. At first glance the model might seem complicated, so it it will be broken down and explained phase by phase. The model is created to be detached into smaller blocks and worked for a large span of companies depending on size. This also makes it possible to keep adding on the model with the company’s growth.

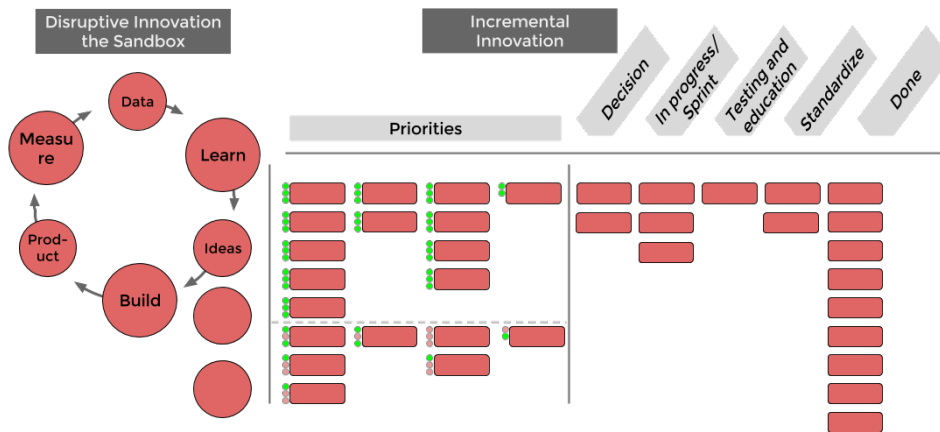


Figure 15 The Model

8.2.1 Internal Disruptive Innovation Versus Incremental Innovation

Disruptive innovation by definition is an innovation that creates a new market and disrupts an existing one. The Internal disruptive innovation could be an disruptive innovation, but not necessary. In this model it is defined as an innovation that are far away from the company's original core competences with a non existing product/service within the company. For example, take a car company that start to produce bikes. Bikes themselves are not a disruptive innovation, but to be competitive, the company needs to add new and even radical features and therefore adding new capabilities. Internal disruptive innovation is what the company is striving for as a startup, to find something new for the company that fits the market.

Incremental innovation is developing something that already exists. This can slowly develop into a new product or service, but won't disrupt the market. This is necessary to be competitive in the market, always strive for cost reduction and improve the product to fend off competitors.

When a startup finds a market-product fit, it continues to innovate mostly incrementally. In the case of telavox, the product became better and better, however, slowly the market and other large companies also developed similar products. This lowered Telavox's average revenue per user (ARPU) by 33% over three years due to price wars. Company's ARPU usually lowers over time (Ovum, 2014) if there are no intellectual property rights working as a barrier for competitors to take market shares. Therefore, it is still important for companies to find new products and areas of higher revenue. If the startup was successful or an existing company had a successful internal disruptive innovation, they would probably have competences as entrepreneurs and have good knowledge of how to find a market-product fit with internal disruptive innovation. Since the product likely needs a lot of optimization, they continue to develop it incrementally which needs other ways and methods for innovation. This is a shift in focus where there are a risk that the innovation processes changes from disruptive to incremental that makes the company

forfeit/forget their capabilities. Later, when the market has matured and competitors are up to speed, it might be too late to re-learn how to do internal disruptive innovations and find new ways of revenue. The companies should therefore continue to innovate with internal disruptive innovation. They should also keep on increasing their innovation capabilities and develop their disruptive innovation processes, even if they feel at the time that they have a product that generates revenue.

Figure 16 shows how the internal disruptive innovation works as a high risk and high reward function for the company, while incremental innovation works as low risk and low reward function to ensure that the company has continuous growth on the existing market. This is true if high risk is equal high reward.

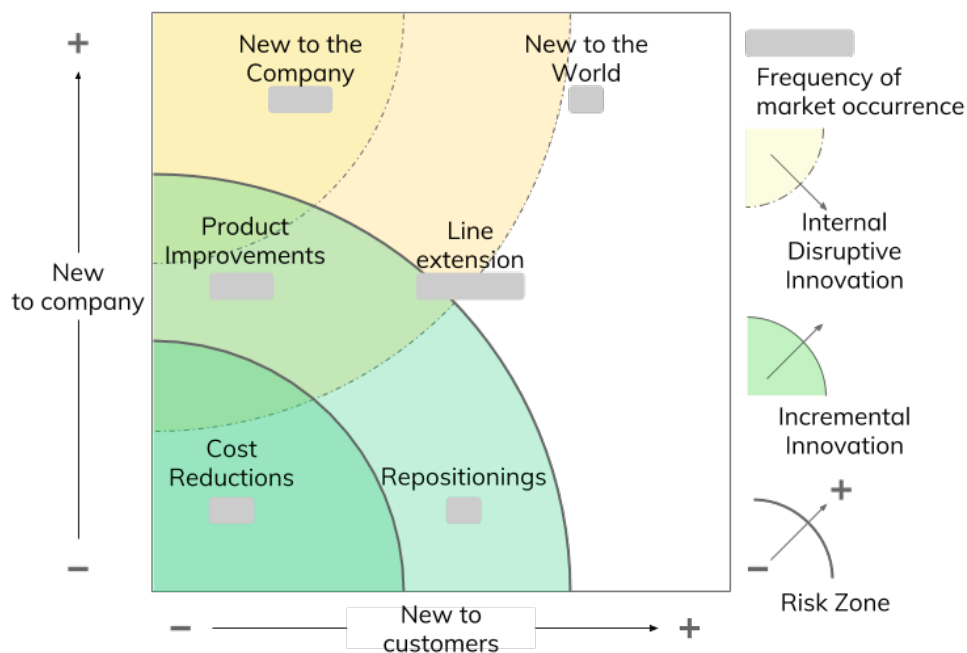


Figure 16 Internal Disruptive and Incremental Innovation as Rewards, Adapted from (Booz, Allen and Hamilton, 1982).

Another way to see it is with the technology S-curve presented below. It indicates how a product, or in this case strategy (since it could also involve services or processes) decreases in performance over time. While the company experienced a period of increased profit due to development improvements that reduced costs or market penetration, they should proactively begin finding the next growth strategy before it's too late. When innovating proactively and in time e.i early, the S-curve will move to the left (as the dotted line shows) and secure future revenue.

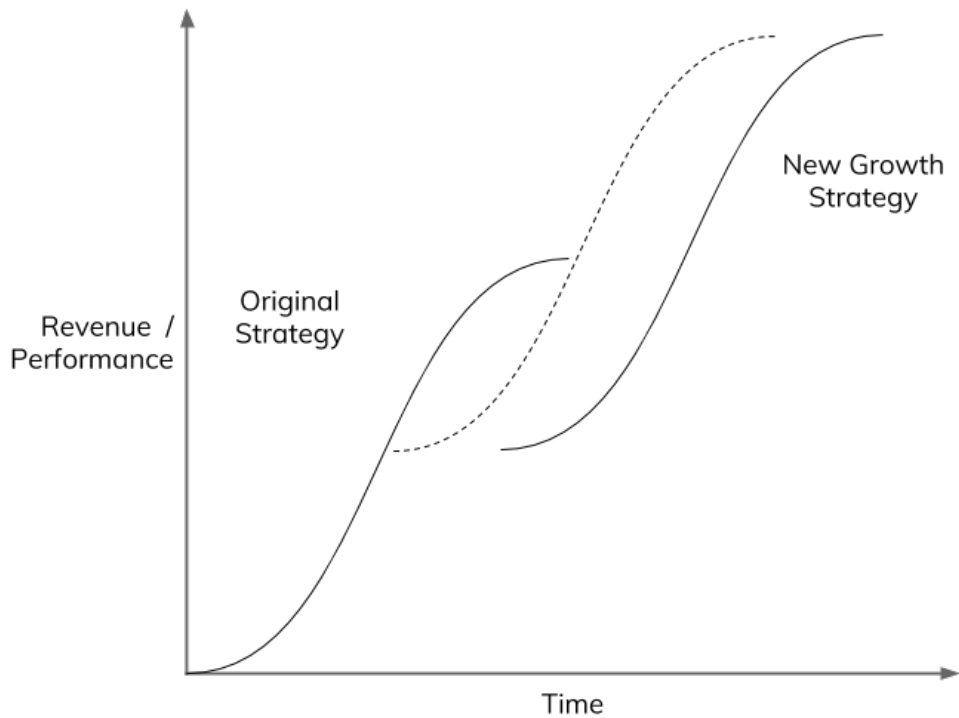


Figure 16 Technology S-curve

The technology S-curve is the result of the cumulative value from diffusion of innovations presented by Everett Rogers in the book *Diffusion of Innovations*. He indicated that different personality types were more likely to adapt to new technology than others. This indicates that it can take time in the beginning for new technology to settle. (Rogers, 1983)

Geoffrey A. Moore released a book in 1991, *Crossing the Chasm*. It explains a phenomenon between early adopters (called visionaries) and early majority (called pragmatism) where it is more difficult to diffundate between the two types of customers. The chasm only affect disruptive innovations and can affect the timespan until a disruptive innovation is accepted by the larger segment. (Moore, 2002) As mentioned earlier, disruptive innovation should focus on the early adopters with methods described from lean startup. To reach larger segments, inspiration can be taken from lean production combined with lean software development to optimize the product or the later presented communication flow anban. The chasm is often displayed by a gap between the two types of customers, and would affect the s curve with the same gap.

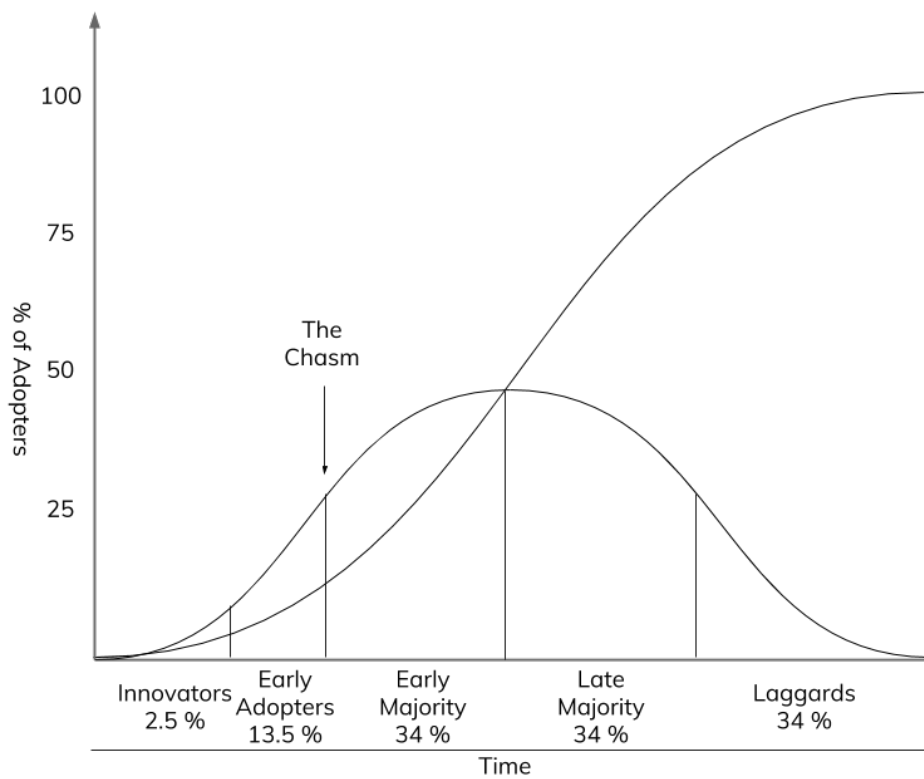


Figure 17 The Chasm and Diffusion of Innovations (Adapted from Rogers, 1983).

8.2.2 Telavox Startup Culture to stimulate disruptive Innovation

Telavox have a saying “*It is better to seek forgiveness than to ask permission*” closely related to the Stewart’s Law of Retroaction “*It is easier to get forgiveness than permission*” (Block, 2003) It stimulates process speed without the need to wait for decisions or answers, and promotes the notion of managers trusting their employees to do their best (compared to the original quote). It is also a way to stimulate intrapreneurs and evolve the company with radical innovations that are beneficial in a fast changing, highly competitive market. “We are a startup company” often echoes in the corridors at Telavox headquarters and in presentations for potential customers, but for how long can they say that? By the definition “A startup is a human institution designed to create a new product or service under conditions of extreme uncertainty” it’s hard to claim that Televox is still a startup since they have a product that proved to been accepted by the market. They are selling well and generate good profit, so in terms of uncertainty, there is close to none in the near future. However, there are some indications that Televox have a strong startup culture.

Telavox are taking some risks that could affect the Telavox economy and to some extent their customer as well. Two examples from the first two quarters of 2016 are:

- The release of a new distribution system with a freemium product
- Reorganization of customer support with personal contact persons, namely Advisors, to all customers

These are both pivots and making pivots is one of the capabilities Telavox still has from their early days. The pivots might not be necessary for short term survival and revenues since they have a good revenue stream. But in the long term, they might be necessary to be ahead of competitors and for long term survival.

8.2.3 Startup Culture in a Larger Enterprise Company

The technology and market is changing quicker than ever before. The companies should adopt the agile methodology and continuously searching for new ways to find income. They must safeguard themselves for other disruptive innovations. With social media and a world that is globally connected with the internet, new technology is not only created quicker, but is also spreads quicker. It can be too late to see and adapt to new innovation, since there is less time to see it coming. On the other hand, this can be an advantage. This is if the company has the capability of being adaptive and can change quickly, without losing momentum in their working processes.

When a company grows, communication can be a bottleneck for change. New ideas might get lost in the noise of other ideas. Opinions and other essential communication flow between rising number of departments, number of people and maybe even between countries creates can start create wastes. Good ideas get lost, and the motivation to share new ideas can die since they are rarely acknowledged and rarely implemented. The people who liked the startup culture to begin with, the same people who have been in the company longest, get the feeling that they can't influence as much as they'd hoped, and therefore run the chance of quitting. If they haven't coached and spread their knowledge for a longer period of time, that knowledge might get lost, and those who created the culture from the beginning are gone.

As the organisations grow both vertically and horizontally, the decision-making process might slow down. This means that the earlier benefits of being a quick changing company might disappear when decision-making grows longer with the chain of managers.

The next chapter will address the difficulty of retaining startup capability (for internal disruptive innovation) within a growing company.

8.3 Disruptive Innovation - The Sandbox

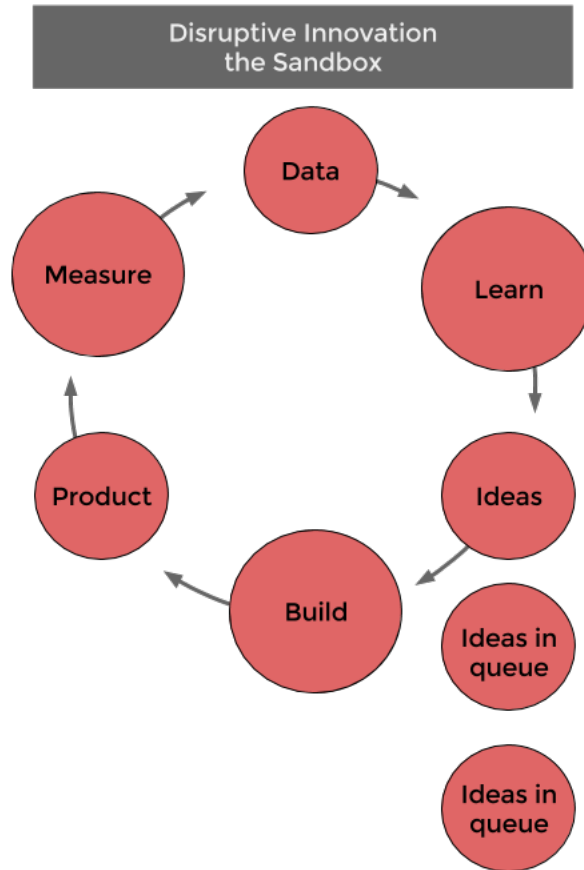


Figure 18 The Sandbox

As previously discussed, companies should continue to use disruptive innovations. But how should they apply it? There are no standard model or right or wrong answers as to how to innovate or be an entrepreneur. According to lean startup, the startup years of a company is about learning. With learning, the company should have capabilities in house on how to innovate. These capabilities should not be forgotten when the innovation that leads to the company's success reaches the next phase, with different management methods of achieving quality in the product.

In theory the build-measure-learn feedback loop can be one good option to stimulate internal disruptive innovation. This was not however tested on Telavox and their might be other models that works better for Telavox or other companies.

Telavox is a successful startup with stable quick growth, good revenues and low customer churn. To be able to recognize similarities in Telavox and the lean startup principles, one of the founders and developer Filip Olsson was interviewed.

8.4 Lean startup at Telavox

8.4.1 Build-measure-feedback loop at Telavox

Lean startup principles indicate that it's important to work close with the customer, monitor what they like and dislike, and make changes early. The early years of Telavox showed some similarities with these principles. According to F. Olsson one of the partners was a management consultant who made strategic visions and plans, these were however not followed. Since they had scarce resources and "*followed the money*", they offered their services to the highest bidder in combination with helping those who were "*loudest*" and those who asked for critical features. Working this way ensured they created value for the customer, since they sold the product before it was developed. As with the build-measure-learn feedback loop, they started to build the product, released it early and worked closely with the customer to get feedback. It was the customers who decided where the development resources were allocated, and the founders were not blinded by the original management plan. (Olsson. F, personal communication, September 9, 2016)

Telavox were mostly working with wholesale customers, which they had few of, which was one reason they never collected quantitative data to be able to measure progress. At the time Telavox was considered a true startup, the tools for testing (like split-testing) weren't as developed as they are today. If they were, Telavox might have utilised them. As it was though, it took many years before they knew what KPI was, so this was not an area of focus. However in direct contact with the customers they got qualitative feedback which was important to keep develop the product in the right direction. (Olsson. F, personal communication, September 9, 2016)

Some of the development culture from the early days remained until the present. In the early days, much like today, it was okay to fail. Telavox advocated experiment and to test things on the customers and see if it works much like the principles behind lean startup. Some capabilities didn't stick. The execution of some ideas takes more time. Olsson stated "*If you don't have the knife at the throat, you lose the edge*". People tend to take the easy way out and produce things that are relatively easy to solve, instead of challenging features that would produce more value in the end. (Olsson. F, personal communication, September 9, 2016)

F. Olsson mentioned that another issue with a growing organisation is the lack of communication between departments. Telavox's products involve two platforms which basically have two larger teams working around them. Even though developers from the two platforms regularly swap teams, not enough of the experience of the processes is communicated between them. Therefore, one of the teams could start working on a new project without realizing how similar

it is to something that was previously developed on the other platform. (Olsson. F, personal communication, September 9, 2016)

8.4.2 Pivots at Telavox

The early days consisted of frequent pivots. No decision was strategically aware ie. approved at a meeting, “*they just happened*” as F. Olsson put it. But still, there must have been forces encouraging the changes. One of them sticks out from the rest since it was due to internal circumstances - the *runway* was out. They didn’t have enough resources to continue working and provide for themselves on the revenue from their cloud platform that they were working on. They found an opportunity to develop a billing system which would build more capital. When it was done, they continue working on the cloud platform since it had higher potential of success. This momentary pivot didn’t just made the runway longer, they kept their abilities and learning from the billing system project and could develop one of their own. They built a modified version of the billing system that is still used today. (Olsson. F, personal communication, September 9, 2016)

When reading about pivots, you get the feeling they are used for strategic change, since the old way was doomed to fail. For Telavox, the function of the pivot was usually to extend the potential of the business, while retaining the original value they found for the customer. For example, from the start they have been using the zoom-out pivot and adding features for the customers. This can be due to the complexity of the telephony services, and what the companies within the industry need to satisfy each customer. Another pivot is the business architecture pivot. Since their first segments of customer was B2B with relatively high needs, high margins but low quantity, streamlining their services made it possible to sell to private customers (B2C) with lower margin and higher quantity. They did this from a wholesale partner which lead to another pivot namely channel pivot. They added sales via wholesale to direct sales and selling through resellers. This was a strategically aware decision to lower the risk of a wholesale partner finding another distributor, and increase growth of B2B sales services through resellers and direct sales.

8.4.3 Engine of Growth at Telavox

Pivots are still happening, but few of the pivots have replaced the old strategies. Telavox has found a way of changing the focus to increase growth while retaining the old capabilities and keeping a stable stream of income. The growth principles in the beginning were mostly considered sticky engine of growth to ensure low churn. In the telecommunication industry, there are some legal aspects that support this with bound times so that the customer can’t end their subscriptions until the subscription bound time runs out. This made it natural to focus on gaining market shares, and later to focus on gaining capabilities to persuade the customer to stay.

Bound time on subscriptions could inhibit innovation. Customers have different ways of communicating their dissatisfaction. However the customer

communicates, the company should address the problem and try to solve it. Since the customers are locked into a contract, the biggest signal of discontent is not possible, namely customers voting with their feet. Customers voting with their feet and showing their approval creates a high incentive to fix issues. These problems can't be neglected due to the fact that the company is losing money - today and not tomorrow. Telavox has recently implemented an annual survey on the customer. It is called Net Promoter Score (NPS), and with one question can indicate the preference the customer has for the company. The question is: *In what grade would you recommend Telavox to others on a scale 0-10?* It is aligned with the Telavox business strategy that quickly takes market shares and an NPS survey can indicate what the growth rate could be. To only measure how often a customer uses a product could give faulty conclusions. This is since the customer could be contractually bound to use the core product. On the other hand, decreasing usage isn't necessary a sign for decreased customer loyalty or satisfaction. For example, as a person grows older, they might not buy a car as often since they don't drive as much, but the loyalty stays the same. (Reichheld, 2003)

Measuring usage is a viable factor, just look at Facebook and it's venture capital. The first summer they had 150.000 registered users. Even though they made a very little revenue they succeeded to raise \$500.000 and less than a year later \$12.700.000 more. Their investor was impressed by two things. The amount of time every user was on facebook (half of the users logged in frequently - once every day), and the rate of growth over expanding from universities at a quick rate. (Ries, 2011) The investors could see the value of usage in combination with the growth rate, and were therefore ready to invest a significant amount of money.

8.5 Capabilities Needed to Create a Sandbox Organisation

8.5.1 Scarce Resources

The team in the sandbox are suggested to be working with relatively scarce resources. This is to stimulate the workers to find a product that the customer really wants. With enough marketing resources it might be possible to push the product in the market and find a few customers to acknowledge it. However, the acquisition cost of reaching the few customers would be higher than the sales revenue from the product. The company might try to scale the production to lower the unit price with hopes of finding revenue in the future, but since the market is relatively small, they might fail to sell all of the products. If they were forced to wait and use marketing methods (due to scarce resources), and had developed the product further to find a better customer-product fit, less marketing resources could be used in the future. The product would be ready to reach a larger portion of customers.

Marketing could give vanity metrics in the measurement phase if cohort analysis is not done correctly. The positive numbers might not be due to the

product itself, but all the investments in marketing. When the marketing campaigns stop, the product might not be able to stand on its own and might fail.

Another reason for working with scarce resources in the sandbox unit is the fact that the project might need to be killed. Being entrepreneurial is high risk and it is inevitable that projects might fail. Regarding a doomed project as sunk cost might make it harder for the decisionmakers to drop the project and find new areas to work on, especially if a lot of time and money was invested. For this reason the sandbox should have budget that defines a runway. When the resources are depleted, it's natural to kill the project so it won't continue to drain the company's resources. Another benefit to having a fixed amount of resources that define the runway are the incentives it creates, these make quicker build-measure-feedback loops.

8.5.2 The Culture in the Sandbox

The unit working in the sandbox should foster a different culture from the company itself. The unit should create conditions that are as agile and fast as possible, and they should be working with scarce resources. As mentioned before, waste is when all processes are not creating value for the customer. Since the sandbox team do not have customers for their specific product/feature, they have to implement the loops to find a good market-product fit. To do this the group needs to complete the loops with speed, so they can learn and redo as many times as possible before the runway is out. The other parts of the company might have long decision-making processes and tests that ensure quality. Everything that slows down the process of discovering what product/feature fits the customer best, should not affect the sandbox team. This means the unit should work independently, and be empowered to make its own decisions without having to wait for approval from higher command. This is the reason it's named sandbox organisation, it is a playground for testing which should not be affected by the culture from the rest of the organisation.

8.5.3 The People in the Sandbox

One problem with innovating new products is those who invented the product tends to follow it phase by phase through its life cycle. Strong creative managers are not necessarily good at optimizing their invention, this needs a different kind of management (I.e. Managers schooled in lean production). (Ries, 2011) Being an entrepreneur within a company should be a job title - intrapreneur. This is a person who act as an entrepreneur but does it for the company and with the company's resources. The people in the unit should have a intrapreneurial profile and be working in cross functional teams to be able to build, market and deploy the product. The skillset should be varied so they don't need involvement from other functions in the company. This to ensure speed in development process.

The sandbox organisation also wider span of career paths or working opportunities for the employees in the company. This create a motivating environment for personnel that would like to have the possibilities to follow up on their own ideas. Empowering personnel is important within lean, employees that can follow up on their own ideas are truly empowered. If the sandbox attracts employees that would otherwise seek opportunities elsewhere, it can achieve Heijunka. The sandbox can provide a great opportunity to learn to work with people from other departments and could increase leaderships skills. When the employee stays and grow as a leader within the company, it can level out variations that come when leaders are not employed in-house.

Being an intrapreneurs compared to entrepreneurs are less risky since the intrapreneur should still be given a salary. However it should not be totally without risk, the intrapreneurs should put personal stake in their innovation and apply a lot of thought before starting. One suggestion to ensure higher risk and higher reward for the intrapreneur compared to the rest of the organisationen, is to have a different incentive model. For one, the salary should be drastically lower than average. This ensures that the company won't throw too many resources into the project, and will create determination for the intrapreneurs to succeed and follow through. If the project in the sandbox is a success, the reward should be greater. One way to achieve this could be to give the unit a percentage of the income for the feature/product/service they created for a certain amount of time, no matter how well it sells.

8.5.4 The Sandbox's Organisation

There are a couple of examples of how to boost innovation in companies. One famous example exists in companies like Google and 3M. They give their employees a certain percentage of time for their regular tasks to work on any project of their choosing that they think would benefit the company. Another way is to dedicate a short timeframe eg. a day, to build something innovative like an MVP and end the session with a short presentation about the findings. (Baldwin, 2012)

This however might not be the best way for lean startups. Giving employees 20% of their work time for innovative projects might create long build-measure-learn feedback loops. Even though employees have more opportunities, the future projects and findings might be postponed as the market changes, making them not viable any more. The intrapreneurs in the sandbox should be specialists at innovation. Some might stay there because of their success, others might go back to operational work and have new intrapreneurs take their place. After some time the company should have good innovative processes and know-how that spills over to newcomers in the unit. This ensures that the same mistakes/discoveries don't have to be repeated.

It is clear that not everyone would like to innovate. Only one out of ten of the engineers at google utilize the 20% time. (D'onfro, 2015) However, many could have good ideas they would like to share. In the build-measure-learn feedback loop there should be a priority of ideas that can be clearly seen for all

colleagues to discuss. This way, employees who are not in the sandbox organisation, can get a feeling of belonging and ownership of products. These ideas are visualized as *ideas in queue* in *figure 19* in the beginning of the chapter. The priority of ideas could be managed with a voting system, or the intrapreneurs could pick. Valve studio, a successful game developing company, have total flat organisation without managers. No one is receiving instruction on what to do from anyone else. However, if they would like to start a project, they have to find participants for the team. If the employees don't believe in the idea, they vote with their feet and no one will work on the team. This will make room for better ideas. (Valve, 2012) The same principle could be applied when picking ideas in the sandbox. If no one would like to work with the idea, chances are it's not a good one, or won't get the attention it needs to be a success.

8.6 Incremental Innovation

Telavox have experienced continuous growth in turnover, number of employees and the number of teams. Telavox has however seen some negative effect on efficiency and effectiveness due to the fast growth. As done previously, this section will also examine the customer contact department and it's communication with the development department. Telavox is a very flat organisation, therefore the vertical communication won't be examined deeper, since there is more efficiency and effectiveness losses to be found on the horizontal communication.

8.6.1 Growth

To visualize the effects of growth, a diagram with round tables is designed, which symbolizes a team. The red chair symbolizes someone asking a question that the individual in the green chair knows, and the team's table is small enough so they can communicate freely.

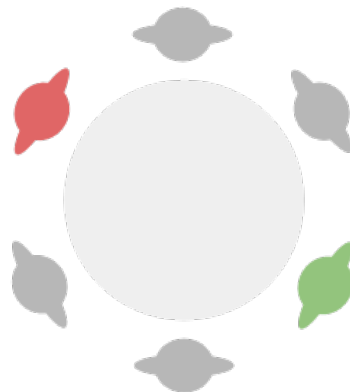


Figure 20 Growth - a Team

A company beginning as a startup probably has a good idea of what everybody is doing. When the red person asks the green person the question, everyone around the table passively receives the information. This is the highest level of horizontal communication flow, where there is no communication loss at all. Everyone shares the same information and decisions are quick since everybody knows everyone and knows their challenges and opportunities. Communication flows easy since everyone knows who to ask and what to ask.

When the company is growing and hiring new people, wastes increases due to problems with communication. The horizontal and vertical communication flow gets more complicated and the process slows since it takes longer for the sender to find the right receiver. Information could be misinterpreted, the information could get into the wrong hands and start spreading without final decisions being made.

When departments are created, people start working in a more secluded setting, with their own processes. If the company has a product or service that changes without other departments insights or knowledge it can start to create wastes. Waste occurs since they have to change their working processes reactively, and the product/service and internal processes to sell, deliver or handle the product/service might not be optimal. Those who are developing the product/service might also lose contact with the customer and therefore might not find the optimal value to develop for the customer. The diagram below shows the challenges one employee might face in finding the right person in a company approximate to the size of Telavox today.

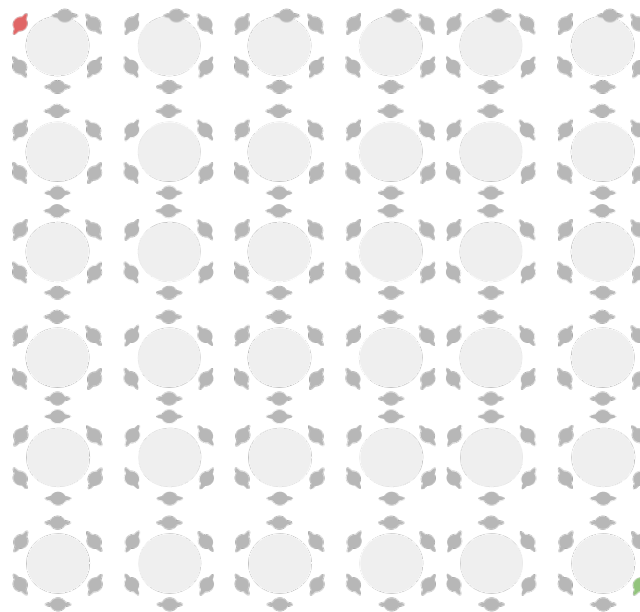


Figure 21 Growth - a Company

8.7 Communication waste experienced by Advisors at Telavox

There is always going to be waste. That is a part of undergoing continuous improvement and kaizen - never settling. One big area of waste for a service company, and the root of many problems, is likely to be bad communication. Therefore, this will be explored further. Most efficiency or effectiveness loss can be described as waste from lean production or lean software development. The examples below are difficult to attribute to a particular kind of waste, therefore new areas of waste is created concerning communication.

8.7.1 Decisions Without Reaching Consensus

Telavox released an updated platform whereby customers were able to buy new services on their own. One of these new services was that the customer could buy telephone numbers without being in contact with any support function. Telavox buys the number from Swedish Post and the Telecom Authority (PTS) for a small cost, and to make sure that customer could not abuse the system, a small cost for each number was to be implemented. A large quantity of developers were involved in the discussion on how this could best be implemented. People from management including the CEO were involved in deciding the new pricing model. It was a high priority task and a dedicated developer did the necessary coding for two months to get it done. When it was finally released, personnel responsible for managing resellers flagged it as being unsatisfactory for the resellers and demanded that the code be withdrawn, which it was. This is not so unusual and according to lean software development it's necessary to learn about the customer and pivot or change until real value is created. But, this is different since it's not a product or service adding value for the customer. This waste was created due to a lack of communication in making sure everybody was aligned with the decision. It should have attempted to reach consensus before implementation, that way the stakeholders could have flagged the problem earlier.

8.7.2 Single Minded Decisions

An advisors working process is closely linked with their customer relationship management (CRM) -tools, when something changes in the CRM-system, they need to change their way of working. Telavox builds their own CRM-system, which contains all the information on the customers, and is used to communicate with the customers. Emails are sent from the system and if the customer contacts the advisor by phone, they are obliged to issue a ticket of information regarding what the advisor helped the customer with. Telavox is a flat organisation, which means that if someone comes up with an idea, there are little to no systems in place for how to actualize it. There are cases where individuals have come up with ideas for new features (in the CRM system for example) that do change the process for the better, for the person involved. However, it ends up causing more overall ineffectiveness for the rest of the team. Therefore, communication must take place between all individuals, before any features are actualized.

8.7.3 Waste Relocation, Suboptimization

Waste relocation can occur when an entire department changes their work process. Telavox experienced this when, as a result, sellers did not end up gathering sufficient information to make deliveries. Therefore, the advisor's department had to collect the information themselves, through emails and phone calls. This ended up making the process longer and more time consuming.

Another example, linked to the example with the CRM system above, is when the ideas for further development are not thoroughly thought through. Trying to save time with a quick idea sketch, without doing deeper analyses and descriptions, causes the developers to receive inconclusive input for the feature. This means they have to spend time figuring out what needs to be done. In some cases when it is done, it might not be as good as originally thought and the developers have to withdraw the code and redo it. This takes significantly more time away from the developers than the time that was saved by the person who perceived the idea.

8.7.4 Lost in the Noise

With more people, more information is spread. A huge amount of information flows through emails, chats and through face to face conversation. This can mean important information being lost in the noise, and lead to:

- Increased waiting time as questions remain unanswered due to unread emails/chats.
- Double work/extra processing as the same information has to be shared multiple times in order to remind people. Reminders increase noise.
- Decreased learning. Relevant information should focus on improvements that can lead to higher customer value. If this is overlooked due to too much shared irrelevant information, valuable understanding could be lost.

8.7.5 Unnecessary Information

One primary means of communication at Telavox is an internal chat service (which they also provide for customers). This is a useful means of communication with quicker response time than email and increased communication between departments. However, the chat is most often used to throw out broad questions to all, and can therefore decrease the possibility of the question making it to the relevant individual for a satisfactory answer. Telavox has situations with group chats of over 80 people. This is where waste can occur, due to unnecessary reading and task switching. This chat service can be a great tool for organizational learning through *Nemawashi* (informal meetings to ensure everyone is aligned on upcoming meeting decisions), but when decisions are made and there are facts on the table, other communication channels should be used to spread this information.

8.7.6 Information in the Wrong Hands and Misinterpreted Information

Some information is shared as suggestions for the purpose of needing decisions to be made for further improvements. There have been examples of when this information was regarded as fact, not suggestion. An extreme example of this is when a seller begins selling something that is not yet implemented. This can force the decision maker to approve a suggestion, even though it might not be the most value creating idea. To reduce the misunderstanding there should therefore be different forums and information channels regarding suggestions/ideas and decisions.

8.8 Waste at Telavox with Lean Software Development Principles

8.8.1 Extra Features

Since developers at Telavox have lost the majority of customer contact, some decisions about what new features and innovations should be administered, are done so on gut feeling. These features might not create value or be used at all, so other priorities on what to developed should be made. This not only pertains to the product, but the internal systems as well. One developer at Telavox mentioned that he didn't know if a feature he built for the internal system was used at all. It wasn't, the department that asked for the feature didn't know it was released. This is clearly an indication that the feedback back and forth from the developers and the other departments were insufficient. Lack of feedback means that there is a risk the features won't be used as intended. The developers are therefore wasting their time to develop and maintenance on unused features.

8.8.2 Task Switching

Developers can have issues brought to them that need immediate attention, they are usually things that are not functioning as they should. This might require them to stop their own work, deal to the issue and later refocus on their own previous task. This takes time and can be frustrating for the developers who are losing flow. To be able to work on a new feature, most developers need time to read old code, and time to see how this part of the code is dependent on the other systems. With small features or bugs, this can end up being a relative large amount of the time for the coding processes. Then when the developer have solved the issue, he/she need to regain focus on their initial problem, which takes more time.

8.8.3 Extra Processes

Since Telavox workers are agile when developing their own product, the documentation is relative low. However, there can be double-ups on work since the documentation on features might have duplicates.

All bugs detected or ideas for improvement are posted as tickets in the Jira system. If someone detects that a posted ticket has already been ticketed before, there is a possibility to mark it as a duplicate. From 2015-01-01 to 2016-08-01 about 600 tickets at Telavox were marked as duplicates, but the unrecorded numbers could have been higher since they have to be detected and manually marked. This is waste since two people have done the same thing, one of them didn't create value. As the earlier tickets have not been resolved (even though there is a need for it), there is waste in terms of waiting. Duplicates occur due to lack of communication, it would be impossible for everyone to talk to everyone before releasing an idea. However, if the communication could be visualised so one could pinpoint what to focus on immediately, many of the communication problems could be avoided and waste greatly reduced. Extra Processes are not just about documentation. Unnecessary meetings due to lack of routine are wastes as well. This often occurs when wrong people are invited to the meetings, or when topics are brought up that does not concern everyone.

8.8.4 Partially Done Work

If priorities aren't executed correctly and large projects start from the wrong end, some code can be useless until the other parts are done. Also, some features are unused due to too many bugs, so the customer contacts can't use it correctly. This doesn't not happen frequently at Telavox since the most features are released in small batches and make it easier to prioritize tasks. However there are examples when managers have interrupt initial plans by relocating developers to other projects that put the initial one at hold. If not the whole system of development are taken into account this can increase wastes.

8.8.5 Unnecessary Movements

The developers are centralized at the offices in Telavox and with their communication tools, like the chat program, there are few unnecessary movements.

8.8.6 Defects

Even though all the code is tested and reviewed by several developers, there might still be hidden defects. If a feature isn't used that often, or the defect so small, finding it the first time might take time. It can take weeks or month before it's found. This makes it harder and more time consuming for the developer to figure out what's in the code that has created the defect. Solving small defects early when the developer has the work fresh in mind, might make the process quicker.

8.8.7 Waiting

The features to be developed are discussed in jira. At occasions the developers lack information to be able to start coding. When the developers ask for more information they often have to wait for feedback. Many times the ticket in jira is forgotten and have to be *bumped*. Bump is an informal saying used to send out notification to the stakeholders so they can take action an answer the questions. This can have effects on priorities were not the most value adding feature are developed when it should have been.

8.8.8 Unused Employee Creativity

Jira is used to share ideas and potential innovations. Some regard this as a possible *black hole*. As the number of ideas grow day by day, the potential for those ideas to be addressed declines. By the time this master thesis is written there will be about 3,500 pending and 80 in progress. If an idea goes unaddressed, it inhibits the continuation of idea sharing and potentially good ideas are lost.

8.9 The Communication Flow Kanban

One part of the model supports incremental innovation by a visualization tool for communication, with the goal to reduce the waste scenarios explained above. The tool takes inspiration from one piece flow as a principle and the tool kanban, since many other lean theories can be supported in the tool. It should be able to be implemented small scale and diffuse it through the company to achieve lean without interrupt a company's daily operations drastically. This should also ensure that it can work for small companies. The process of working with the tool will be described step by step and analogies will be taken from Telavox to describe how it can work in practice. Since most of the incremental developments for advisors are features in Telavox's services or CRM systems for the advisors, the examples will be from the point of view of developing features. But it could work for suggestion for improvements on processes as well. Finally, every section will delve into how it supports lean principles and how it can reduce waste.

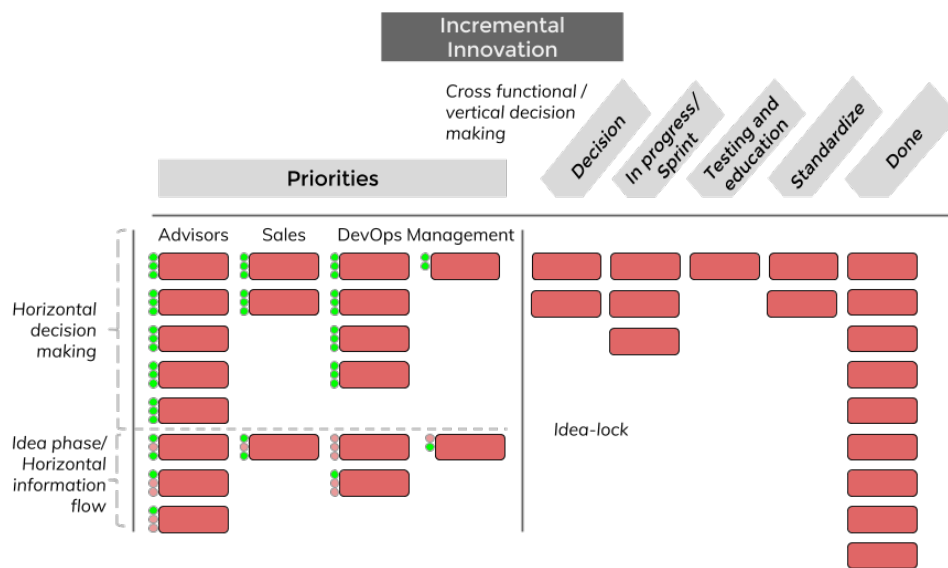


Figure 22 Communication Flow Kanban

8.9.1 Describe a Broad Problem in an Epic

When an opportunity for improvement is detected, for example if something goes wrong or an idea for a new feature is recognized, it should be communicated directly. The analysis of the problem or feature should only be specifically about the problem, or what problem the feature would solve. This kind of focus can uncover the root cause and the full extent of the issue. Those who describe the problem should be the ones who experienced it, they will have the best understanding and information. There should not yet be discussions on how to implement the changes. When those involved (often workers) discover a larger problem, they should either start creating something called an *epic*, or if it's a solution to an already existing problem an *solution under an epic*. The solutions under the epic will be described in the next phase. The epics are the red rectangles in *figure 22*.

An epic is a combination of an A3-report and a kanban board (for software developers). The epic includes a batch of several smaller solutions to wider and larger problems. The number of solutions in the epic can be compared to batch size. This is dependent on either the number of solutions, or the time it would take to create the solution. The epic is designed to give an overview of the problems and the opportunities, and be a foundation for priorities. The design of the epic should be standardized so that team members, managers, developers or other stakeholders can quickly and easily get a good overview. This also make it possible to be able to compare the epics between departments and make accurate decisions. The epic takes inspiration from Toyota's A3-report using the principle that to make an accurate decision, all subject information should fit on a A3 sheet of paper. The only difference between the epic and the A3-report is that an epic should only provide enough information on how to

prioritize problems that need solving first (according to the department), not also suggest solutions like in an A3-report.

The Design of the Epic – Background

Describing the reasons behind the issue could create understanding in other departments, who might not have their issues solved directly. This will reduce the risk of dissatisfaction and competition within the company. The background section in the epic takes inspiration from the book *The Pyramide Principle*, which uses the letter combination SCQA. It discusses that used in a short, structured way, SCQA can foster understanding of the issues for managers and other departments, and make for easier prioritizing. (Minto, 2009)

The S stands for *situation*, which should be described as simply as possible without any opinions. This should create a foundation of shared knowledge among all the readers.

Example:

“When my alarm rings in the morning, I turn it off and fall back to sleep. This make me late for work”

The C stands for *complication* or complexity. These are the issues regarding what complications arise from the situation as it is today. This should help the reader understand the problem and the extent of the problem. To figure out the extent of the complication, a 5-why analysis can be done. It can determine the root cause and help to figure out the right questions that need to be asked to solve the issue. This maximizes the value that would be created if the epic is solved.

Example:

“Since I’m always tired in the morning, I feel like I have to sleep more. I don’t use the snooze button on my alarm as I’m not sure how it works. I’m aware that my colleagues are frustrated with me since I’m always late”

The Q stands for *question*. People tend to start with the question, but this method encourages the reader to understand the reasons behind it, therefore they are more aligned with the questioner. One might think this would be a good question: “How can I wake up on time in the morning?”. But this does not encompass all of the complications. So, the questions should be as follows:

Example:

“How can I ensure that my employer will be satisfied with the time that I arrive at work?”

The A stands for Answer. The method was originally designed to convince the audience on a particular subject. (Minto, 2009) The answer is not supposed to be in an Epic but are supposed to be concluded in the issue in epic. The

main reason the Epic was created is when an employee asks “*why are we doing this?*” Or “*how should we do this?*”, since it is probably an issue with a broader subject that should be evaluated in an Epic. But to see the effect of an SCQA analysis answers for the question will be presented. Note that the third answer would not be discovered with the question “How can I wake up on time in the morning?”

Example for answers could be:

1. “*I should create a new evening routine so that I can get to bed on time.*”
2. “*I’ll learn how to use the snooze button by searching it on the internet.*”
3. “*I’ll talk with my colleagues and figure out a way where I can start work later in the morning.*”

The Design of the Epic - Analytic Framework

An analytic framework is mainly used to quantify the issues. The unit for measurement should be something that the employees within that department can easily identify and make calculations (or calculated guesses) on, without complex formulas. For example, advisors use the unit of *time*, since the cost of handling customers rises with time spent per customer. Calculating time spent early on might be too large to grasp for a single employee, therefore it is described in days. However, other departments like sales, don’t make time-saving an overall goal. Their value lies in sales volume, and what monetary worth the solution to the problem, or creating the feature would add. The quantification can be both for internal processes and/or for customers. How to compare different department’s epics to help with prioritization is described further in the *decisions phase*. Not all problems are correlated with time savings for advisor, for example reducing churn is another important priority. This can be added in the calculations so there are not a shift in focus to a single goal.

In the analytic framework, customer benefits should also be considered. As there are no suggestions for improvements, but rather descriptions of the problem, the benefits would be those if the problem was removed altogether. There are benefits that can’t be quantified but can still create value. Describing the benefits for the customer creates awareness, and when suggesting ideas for improvement, they should always add value for the customer.

Figure 23 is an example of how the epic could be designed.



Figure 23 Epic Design

8.9.2 Supporting Lean Principles - Epic

Lean Production Principle 2. One Piece flow

Even though epics tend to indicate a larger problem, each epic should not take too much time to solve. It can be compared with large batch sizes in the production industry. One advantage the service industry has over the production industry is that the whole batch does not need to be created. A portion can be completed, and the rest can utilize a new epic for further analyses and prioritized evaluations. Not only should the epic conform to a reasonable size, but the issues within the epic should be broken down as well.

They should be small enough so they can be solved in 1-2 weeks (depending on the resources and the company), and the method for solving the problems are suggested to be agile. An IT research advisory firm - Standish group conducted a research experiment on software development. The results was that 62% of the successful projects in software development were small projects, while 2% and 6% respectively were huge or large projects. Among the failed projects, only 11% were small while 17% and 24% were huge or large. This argues the case for small projects. The same report also proved that the agile method was successful in 39% of the cases, while the waterfall method was successful in 11%. (Hastle & Wojewoda, 2015)

With epics that include smaller solutions (like smaller batch sizes) it is possible to release features quicker and earlier, which creates less risk of developing non value adding features and can save time for the developers as well as advisors.

Another benefit of creating one piece flow in the development processes, is that it is easier to spot bottlenecks with different problems. Today, development at Telavox is chaotic and has limited structure. When the projects are done in smaller batches, it becomes easier to measure and find waste in the development processes.

Lean Production Principle 4 - Leveling out Workload (Heijunka)

When developing in small batch sizes with frequent communication from advisors, to developing on a regular basis, staff schedules can be leveled (heijunka) as they don't have to complete large projects. These large projects usually involve a lot of research and detailed specifications, and take a significant amount of time. When the new features from small batch sizes are live, the staff's working process doesn't change radically.

Small batch sizes can also level out the developer's workload (Mura) and therefore reduce the risk of overburden (Muri) A stable flow of tasks with short deadlines makes it easier to plan and set targets. With very large projects, it is hard to give a good estimate on how much time is remaining. If the large project is a blocker for other parts of development or process, it creates high pressure to finish.

Lean Production Principle 13 - Making Decisions Slowly by Consensus

To make decisions slowly Toyota advocates 5 steps:

1. **Understand the situation truly by seeing for yourself.** Since those who wrote the epic should be the ones who experienced the problem, this is built into the process of documenting the issues.
2. **Understand the core problem.** Many problems found could at first glimpse look small or be isolated to just a few customers, therefore not serious enough to create an Epic. If a 5-why analysis is made, the extent of the problem can be widened and solve other problems related to the first problem. For example, when a customer is churning, it's

possible that there is a problem within the company that should be addressed. This is an example of how it could look for Telavox:

Why is the customer leaving?

The customer is not happy with the services

Why is the services not satisfactory?

The mobile reception in the factory is lousy

Why is the reception in the factory lousy?

The concrete walls make it hard for signal to get through

Why don't the customer know this?

We haven't informed them that there could be problems with thick concrete walls.

Why didn't we inform about this?

We don't have clear routines when selling or delivering to the customer.

In this example, there is a small chance that the customer's problem will be solved if they go to another contractor. The answer to this problem is to install a repeater or signal enhancer in the building, rather than initiate a new procurement. It is almost always possible to backtrack the problem within your company. Telavox could be more proactive and could provide better research about the customer's situation. Solutions in the epic could have the following headlines to answer the problem.

- Sellers should complete a more thorough environment check before signing
- Sending out test equipment to the customers before delivery
- Adding services with repeaters and signal enhancers
- Visualize and explain eventual problem with mobile reception in buildings.

When asking why five times, one can see that the root cause can derive from human error, and could be pinpointed to a single person in the company. It is however important not to blame each other. Making and finding errors should be encouraged, when the problems are highlighted and the process of solving them shared, there is increased learning.

3. **See phase 8.9.5: Suggested Solution in the Epic**
4. **See phase 8.9.8: Reaching a Consensus**
5. **Information sharing and communication should be efficient and easy to understand.**

An A3 report is one tool to make information easy to understand. The design of the epic suggested in this thesis is based on theories from the pyramid principle and is intended to create enough information to make accurate decisions. The reader should bear in mind that the design it is not thoroughly tested on management and the workers who have used the design framework on their own epics, have missed some of its principles. It might take time to learn what it is all about. Still, the epic might need to be re-

designed and adapted to the individual companies capabilities. If so, the epic should be designed with the principles of an A3 report. That is:

- Communicate with as few words as possible to make the core purpose as understandable as possible.
- Make it available for all stakeholders
- Create means for easy decision-making
- Reduce risk of misunderstanding

Lean Development Principle 7. See The Whole

Without an epic, developers might begin working on a feature that in turn sub optimizes another feature or working process. The epic partly solves this problem by a wider complication description, by providing an overall goal with a question and producing alternative solutions within the same context. This can make it easier for the developer to create the feature that works best for a future system.

8.9.3 Reducing Waste - Epic

Unnecessary Information

An epic is where all the issues are gathered around a specific problem. If a worker finds a structured way to locate the documentation around the problem, it could reduce unnecessary questions.

Getting Lost in the Noise

The ensure that all the documentation is in the right place, not forgotten in a chat group or mail inbox. This can reduce wait time, double work and extra processing because the same information doesn't need to be shared twice. It can also increase learning/knowledge within the company since the information is easy to find and no necessary information are overlooked.

Task Switching

Working on a larger project with similar content can reduce the waste. The epic gathers problems that can give synergy when working with different solutions, as those involved get a better understanding of how the code integrates with the system.

8.9.4 Suggested Solutions in the Epic

The Epic contains broad sections of solutions that are relevant to the concluded question, namely *solution in epic*. These solutions tends to appear over time and are gathered as a list of ideas, iterated by the workers in the departments. This stimulates Kaizen by structuring and categorizing the ideas for further improvements.

Ideas for further development should be deconstructed into small and easy tasks suitable for things like sprints, which can be used by developers. Every

solution in an epic can be assigned to an individual or group of developers to solve and should be set into a short timeframe to be able to minimize waste.

There could be several solutions for the same problem in the epic, this is to be expected. It creates an opportunity to use set-based concurrent engineering and evaluate the solutions in a later phase in the developing process.

The solutions in the Epic is not visualised in the picture. There are likely to be too much information in all the issues in epic to be able to visualise them.

8.9.5 Supporting Lean Principles – Suggested Solution in the Epic

Lean Production Principle 10. Respect, Develop and Challenge your People and Teams

By enabling workers to be problem solvers, they will be greatly empowered. The workers have the most knowledge of how their jobs work, and are the ones working closest to the customer. Therefore, they should be more capable of finding solutions than the managers. At Telavox today, when a worker encounters a problem and has the time, they post the problem in jira. There could be many problems that they haven't encountered yet, and they could have ideas for solutions. By categorizing areas for further improvements in epics, and explaining the problems as they are, it enables workers to be creative and find solutions. It is important that all of their suggestions be addressed so that they feel heard, valued and are encouraged to keep finding new ideas.

Empowered, problem solving workers can create an innovative culture. To stimulate this, the managers should be supportive, challenge the workers thoughts and coach them on their solutions.

Lean Production Principle 13. Make Decisions Slowly by Consensus (continued)

3. One should broadly consider different solutions.

At Telavox today there are suggestions for improvements scattered everywhere. If there are several suggestions in different issues to be addressed to solve the same problem, one of those might steal the focus (due to debates about details) and therefore might be taken without considering the others. Instead, if these suggestions are gathered under the epic, it's easier to evaluate them closer to the decision phase, or even start to develop them all concurrently. This will lead to finding the best solution in the end.

Lean Development Principle 2. Amplify Learning

For developers, there are several benefits when creating smaller issues under a larger group of common projects. It can improve the quality of the feature due to quicker feedback, since it enables development based on fact, not forecast. When a feature is complete, they will get quick feedback on how it works. This makes it possible to do small incremental adjustments. The feedback also brings the development process closer to the customer. When

dealing with smaller tasks under the epic, it creates a motivational environment with workers that can see weekly progress and the end goal of solving it.

Lean Development Principle 5 - Empowering the Team

When workers are involved in the innovation processes, it can create a strong sense of belonging. In finding their own solutions, this positive, intrinsic motivation caters to a higher sense of pride in the work. When an issue is dealt with, they know that they were a vital part in making the company a better service for its customers.

8.9.6 Reducing waste – Suggested Solution in the Epic

Unused Employee Creativity

When all solutions get attention, no suggestions for improvement are overlooked. If an idea is rejected and therefore not implemented, it is important that the employee receive feedback as to why. This ensures that the employee still will keep on giving suggestions for improvement, even though if nine out of ten is rejected, the one that was approved and implemented creates value for the company.

8.9.7 Reaching a Consensus - Horizontal Decision-Making

Once workers have discussed and built a framework in the epic, other teams under the same department should join the discussion and contribute. When all the teams in the department have agreed on the solutions and reached a consensus, the epic containing all the solutions, goes to the next phase. If the team have reached consensus or not is visualised beside the epic with green or red dots in *figure 22* and *figure 24*. The green dots indicate that all issues in the epic are approved by the team. The red dots indicate that the issues in the epic are not approved by the team.

The process of reaching a consensus are as follows:

- Workers give suggestions for solutions regarding an issue in the epic. This is in the idea phase.
- The team have weekly meetings where the suggestions are addressed. If they have opinions about them, someone addresses the problem in the epic and informs the other teams. They should try to find new solutions to the problems. In this phase the team still has a red dot on the epic to show that they don't agree.
- When there are finally no more opinions/issues from the team, they signal this by switching to the green light.
- When all teams have signed off with a green light, the epic moves into the *horizontal decision-making* phase. Here the teams discuss the priority of the epics to address the most value adding features first.
- Finally, the team leaders have a weekly meeting and decide on the priorities using the information given from the teams.

In practice it could take several meetings until a consensus is reached, and can seem like a slow process early on. In the case of Telavox, good ideas are often widely supported, however misunderstandings can affect this. The process ensures that everyone understands what the ideas are truly about, and can be adjusted so that misunderstandings don't occur later on, leading to even greater waste.

Reaching a consensus spreads knowledge and understanding across the team. This avoids centralized knowledge among only a few key people. If only someone in the team is a '*know it all*' this creates a high risk situation. If they decide to terminate their job contract, they leave with their knowledge.

When the epic has brought about a consensus, it goes into the idea-lock (visualized with a dotted line in *figure 23* and in *figure 24*) state where no further features can be added. This reduces the risk of an epic being constantly altered and never concluded. However, if it does still really need a new feature, it should be withdrawn back into the idea phase and continuously discussed until a new consensus is reached.

8.9.8 Supporting Lean principles - Reaching a Consensus

Lean Production Principle 5. Build in Quality (jidoka)

Even though this is not an automation, it can be viewed in that way. Since no single minded decisions go through to the developers, the organisation stops the issue before errors occur (instead of the machine in production). The red and green dots are inspired by andon lights and communicate the message "*there is something wrong here that we need to give attention to*".

Lean Production Principle 9. Grow Leaders Who Thoroughly Understand

When team leaders are working closely with the team to solve a problem, innovate the product or create better working processes, they acquire a better understanding of the workers challenges and hardships. With this understanding, it is easier for them to mentor the worker and further empower them. They can help by guiding them to the right decisions and thus, solve the problems. The team leader should have good insight into the long term goals of the company, and should make sure that the solutions are aligned with that overall goal.

Lean Production Principle 13 - Make Decisions Slowly by Consensus (continued)

4. Building Consensus, with all Stakeholders.

Toyota advocates informal meetings called *nemawashi* where they gather information, and prepare and discuss with the attendees before the meeting. By doing this they increase the probability of achieving a consensus at the meeting. If propositions are achieved with input from all attendees, the likelihood of making accurate decisions increases. To prepare everyone in the whole department for a decision with *nemawashi* would be impossible. The team

meetings are supposed to work as an equivalent to the informal meetings. They serve the same purpose, to get a good idea of what the decisions should be, but not to actually make the decision. The team meetings gather information that can improve the decisions, make the decision processes quicker and increases the chance of reaching a consensus.

8.9.9 Reducing Waste - Reaching a Consensus

Single Minded Decisions

When the team must sign off on all the solutions, there is no risk that a solution based on a single minded decision is developed.

Extra Features

Only features that have been chosen and decided on should be developed. The idea-lock indicates when something is good enough and the development process can start. If new features seem suitable for an epic, but that epic has been locked, they should not be added. Instead, a new epic should be initiated. It can be similar to the first one, but must follow the whole communication flow kanban process, be evaluated and achieve full consensus before it's developed. This ensures that no extra features are developed.

8.9.10 Making Priorities

Members from all teams should discuss the priorities and give input to the team leaders, who then decide with the rest of the team leaders. Those actually doing the work have the best feeling as to the extent of a problem and what is most critical for the customers, therefore, are most suitable to fulfill the priorities. This is also an important part of empowering personnel, as they are involved in decision-making from early on.

The priority section can have several departments that are not dependent on each other. Each department should cover their own priorities. They are however still shown on the Kanban, since it's important to see what challenges the different departments face. This helps the departments to see opportunities on their own, where development could have a spillover effect. Another positive outcome in providing a visual representation of department priorities, is the benefit of gaining a deeper understanding for how the decisions are made. It could be difficult for a worker to have good ideas that others believe in, but that are never realized and developed. If one can see and understand that other departments have urgent plans and better ideas, the worker should feel less discouraged.

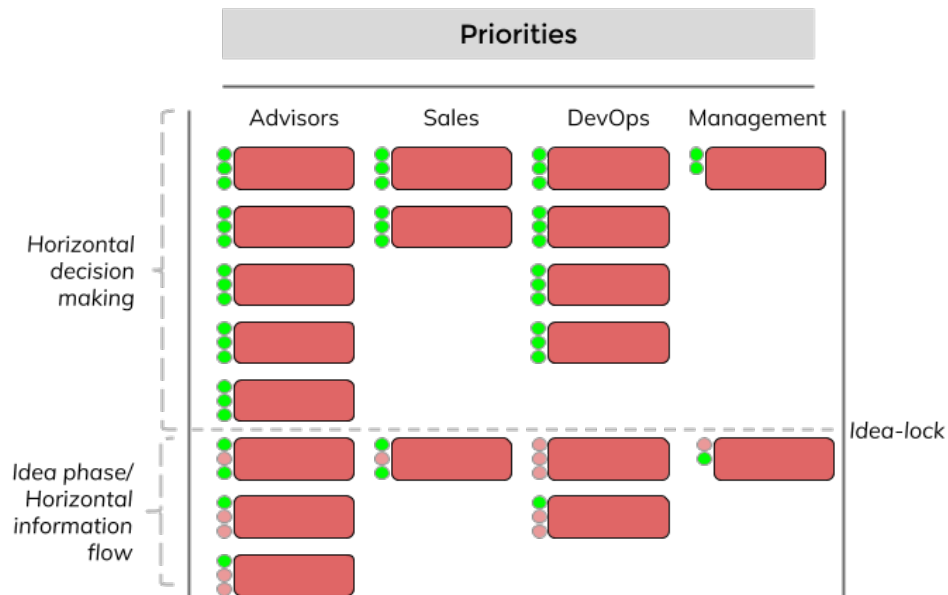


Figure 24 Priorities

8.9.11 Supporting Lean principles - Making Priorities

Lean Production Principle 10. Respect, Develop and Challenge your People and Teams

When the teams working closely with the customers can be a part of prioritizing the development the decision-making comes closer to the customers. Here the solutions are not just regarded by one individual, but need the group to work as a team and figure out what the most important priorities are. The teamwork required to achieve the priorities can also be a way for the members to learn from each other, and give the team a better understanding of their daily challenges.

8.9.12 Decision-making

Decision-making can concern deciding on new working processes or routines that should be implemented, investments in new equipment or as with Telavox, development of the product or CRM tools. The decisions might require a better overall understanding of the company and their long term challenges. The team deciding should therefore be cross functional with representants from the departments, so all stakeholders are aligned and the decisions are understood by all departments. One department might get several of their ideas processed first, at least for a period of time. This can be strategic, for example if a company is trying to boost sales, the ideas for improvements from the sales personnel should get more attention at that time.

Managers deciding what should be done have capabilities of calculating cost either in employee resources, investment or time to develop. They should pay attention to the priorities, but can freely choose what to take action on next, from either department and in what order they prefer. Their attention should only be given to the top five Epics in the idea-lock stage. The limit is set to five Epics since these should have been sufficiently prioritized and therefore be truly value creating. Limiting the number of epics developed, and only concentrating on those that have reached consensus also accelerates the decision making process.

Even for the projects that have been approved, time is initially required for acquiring the right people or resources, this is a another topic that have to be decided - who and how many should work on the epic.

As was shown in the previously mentioned situation regarding the advisors example of quantifying the value for solving the epic. The value adding measure was *time saved in hours per day*. Other departments are probably going to have other measurements, sales could have *increased sales per month in profit and seller*. When making decisions on time saved in hours per day for the advisors it should be recalculated as cost savings per year and for the sellers recalculate as profit per month and sellers to profit for all sellers per year. When dividing the combined cost savings and profit per year with the time needed, it is possible to compare which creates the most value for the company.

8.9.13 Supporting Lean Principles - Decision-making

Lean production Principle 1. Long-Term Thinking

All decision should take notice to the long term benefits and not the short term costs. Long term decisions should be in the mindset of managers taken decisions. When the managers are comparing epics between the departments it might not always be best in long term to give priority to the most value adding epic at the moment. An example of this could be when comparing sales' epics and the advisors, since the reorganisation of the advisor department had some effects on the motivation and working environment for the advisors, their epics should maybe be given higher priorities to make their work easier and hopefully prevent employees to terminate their contract. Even though a sales epic could generate more income than the cost savings were for advisors.

Lean Production Principle 3. Pull Instead of Push

Even though Telavox has a flat organisation where workers are free to contribute with ideas for improvement, the larger projects are usually driven by managers who push new features down through the organisation, all the way to the customer.

With the communication flow kanban, decision-making with priorities and suggestions for solutions are relocated within the organisation closer towards the customer. This is closely linked to innovation pull, since the customers affect the customer's contacts (advisors or sellers) who pulls innovation from

the customers instead of managers pushing it on the customers. The communication flow kanban also stimulates pull in the working processes since it helps to indicate when developers are reaching the end of their project. The epics flows in the communication flow kanban like products in the production line, when the developers are ready for a new task they pull new assignments down through the organisation.

Figure 25 below shows how the ideas and work flows from customers, thru advisors and managers to the developers. Figure 26 visualize how information travels from developers back to the customers

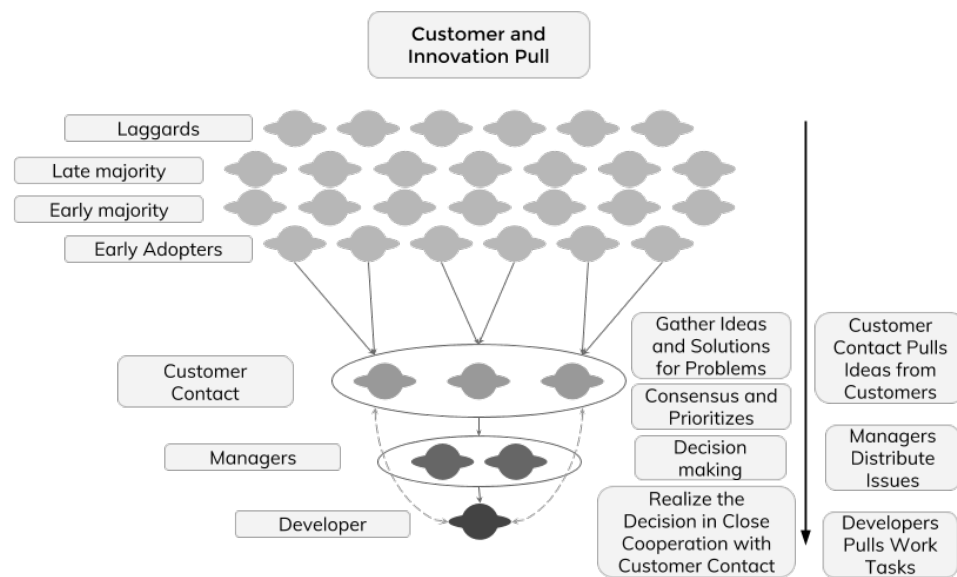


Figure 25 Customer and Innovation Pull

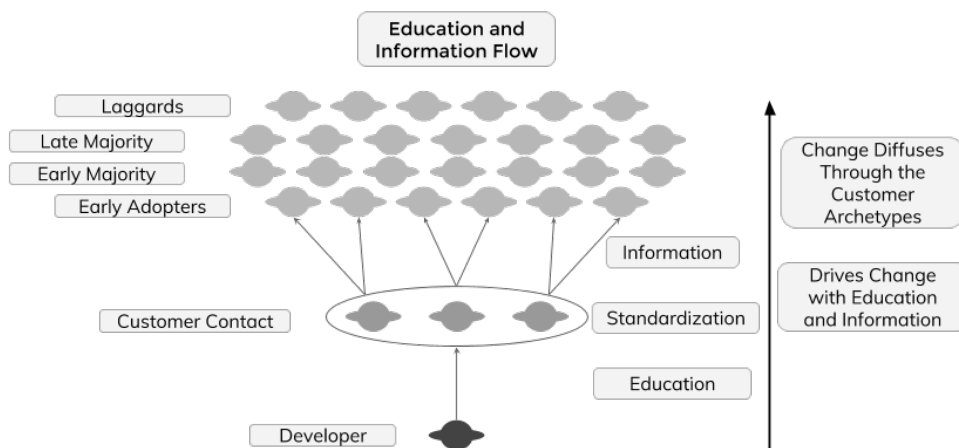


Figure 26 Education and Information Flow

Lean Development Principle 3. Making Decisions as Late as Possible

In the communication one piece flow kanban, decisions are made late in the process, just prior to the developers beginning their work. In a fast changing environment, the circumstances regarding the business might change, causing work the developers have done to become obsolete. Making decisions late also increases the information gathering time for projects in the queue. When all tasks are gathered and there's been time to reach a consensus, there is less risk of the developers starting at the wrong point and having to re-do work. The developers can also try to figure out different ways to solve the problem simultaneously with the set-based concurrent engineering approach, and later decide which way works the best. This requires less time to plan and analyze and instead tests which creates the most value. Many problems might be found while coding, but adjusting the solutions and applying workarounds might in the end produce a feature that is as value-adding as initially intended.

8.9.14 Reducing Waste - Decision-making

Information in the Wrong Hands and Misinterpreted

It should be clear that nothing is decided prior to the actual decision phase. If this happens, certain features can mistakenly be communicated to customers, and the customer's high expectations might force the decision makers to settle with a less value-adding solution or priority in development. But since it clearly visualised what is decided and not the risk to take suggestions for decisions can be reduced.

Relocating Wastes, Suboptimization

The deciding managers should be a cross functional team with representatives from different departments, then they can make sure that the changes do not negatively affect another department. If the feature is necessary, but has a negative impact on a particular department, the certainty should be that the value created for one department is higher than the reduced value in the other. If not, new solutions might need to be found.

8.9.15 In Progress - Start Developing

When the team is assembled and work on the project begins, the project is *in progress*. Before proceeding with development, each team member should personally *go and see for themselves* to get a good understanding of the situation they intend to solve. The developers should work closely with the client so the finished feature accomplishes what was intended. Preferably the developers should see how they work and get a good understanding where the problem arises.

It's also important to visualize that the epic have reached the *in progress stage*. This should signal to the clients that they quickly need to give feedback to the developers questions and therefore decrease waiting time for the developers.

8.9.16 Supporting Lean principles - In Progress

Lean Production Principle 12. Go and See for Yourself

Even though all information regarding the problem and solution should be documented for the developer, they should still take time to see the problem for themselves before developing the solution. This reduces the risk that the documentation is misinterpreted, and increases speed since asking and answering question on text basis can take time. When the developer understands the core of the problem, it can make sure that the solution doesn't miss any vital elements.

Lean Development Principle 4. Deliver as Fast as Possible

The earlier increase in value for the company or customers is one reason for supporting quicker delivery. But releasing the feature sooner also makes it possible for earlier detection of bugs and helps the developers' work to be more efficient. The client won't always know what they want until they see it. This was obvious when developing a dashboard for the CRM tool at Telavox which the advisors were going to use. Before the project started, the input was relatively low, it had 5 sprints or smaller issues. When the dashboard were developed, 36 new bugs or propositions for features were reported. The advisors didn't really know what to ask for before they had a foundation to work from. Once they did have that foundation, the developer received a lot of input. If the developer waited to deliver the foundation for the advisors to test, the development time would have lengthen greatly.

Lean Development Principle 6. Build Integrity In.

Even though one developer might be assigned to solve a single issue in the epic, they should have a good understanding of the other issues as well. This ensures internal integrity so the new features work well with existing features. At Telavox, it sometimes happens for new features to cause disfunction in other parts of the system. It is hard to maintain internal integrity when there is a balance of needing to ship out new features quickly (to enable live tests) and retain quality. If the developer understands the surrounding system and what is about to be developed, future system dependencies could be optimized.

Since the epic includes information regarding the value given to the customer, it could increase external integrity as well. At Telavox, developers have very little customer interaction. This makes it easy to focus on the technical issues, solve them and forget about the customer's perceptions. Even though the customer's value is described in the epic, it is important to deliver quickly, get feedback and adjust the feature.

8.9.17 Reducing Waste - In Progress

Waiting

When a feature to be developed is shown on the kanban, the client should be aware that it is important to answer any question as soon as possible. This can reduce waste in terms of waiting time for the developer.

Partially completed work

Even if all the issues in the epic cannot be solved in the time frame required, the developer can finish what they started working on, if the solutions in the epic is small enough. If there were larger solutions/projects that take months to finish, and it get interrupted with other features that are prioritized higher at the moment. This would mean that all the time working on the first feature would be partially completed work.

8.9.18 Testing and Education

The complete feature should be tested before running it live for the customers or internal stakeholders. In development, there are different methods and processes for testing the code. This testing won't be discussed further, however testing the feature for the clients is still important. As mentioned before, the clients may not know what they want until they see it, this is their opportunity to see how it works and if it solves the intended problem. If not, it might need to go back into production until the feature is fully functional. One way to increase communication is by letting the developers educate the personnel on how the new feature/process works and why. This can also increase feedback when questions arise. Some things might work, but with small adjustments it could be further improved and easier to understand. In the case of Telavox, the advisors are working daily with settings in the customer's IT environment to help the customers with settings the customer doesn't understand. The goal is to make the customers self sufficient and to be able to do all the settings on their own. If advisors have questions of how new features work, the customers probably wouldn't have figured it out by their own.

How the testing and education is carried out depends on how large the group of stakeholders are, and to what extent the feature or process affects the overall experience. Bigger changes might need several seminars to explain the concept and how it works. However with smaller changes, a short sit down with a few representatives could be enough. The testing should be considered complete when representatives from all teams approve. Then, it should be released live and implemented to all relevant colleagues.

8.9.19 Supporting Lean principles - Testing and Education

Lean principle 5. Build in Quality (jidoka)

This is another area where faulty features can be spotted and stopped before running live. When the developer can see how the clients are using it live, they might see that it's not being used as intended. The client on the other hand might be unaware of the problem and fail to report it.

8.9.20 Reducing Waste - Testing and Education

Defects

When the developers are educating the client on their product, they have final input, and can ensure the minimum number of defects in the feature.

8.9.21 Standardize

Even though the feature is live or a new process is partially implemented, it is important that it is used throughout the organisation. The epic makes it possible for everybody to see what changes have been made and what this means for their department. Not all matters concern everyone, but when they do, team leaders, coaches or managers should make sure that the personnel use the new features or processes as they should. This can be likened to Taylorism, monitoring how the people are working and making sure they follow their routines. This is important, because if everybody does things their own way, further improvements would be difficult to implement.

Creating changes using one piece flow as an inspiration to make one change at the time can make it easier to develop a working standard. Adding a new feature might just take hours, but the working habits and communication needed to inform everyone can take days or weeks. This length of time is to ensure that the changes are diffused throughout all of the employees. Empowering the worker to come up with ideas, get along with their colleagues through consensus and lastly prioritize the ideas, should make them willing to adjust to the changes without any resistance.

8.9.22 Supporting Lean Principles - Standardize

Lean Production Principle 6 - Standardize

Standardization is especially important when optimizing internal processes or developing new features or support systems. Standardization is the phase where technology meets the people. If clients are not using the features as they should, the success of the feature becomes irrelevant, the process changes from day to day, and it is difficult to see improvements as they are just regarded as deviations. Without standardization, it is hard to measure the impact of the change, even if the feature improved the product.

Lean Production Principle 7 - Visual Control

Kanban is a method of visual control. Standardization can also be a way to visualize further improvements. Since every department belongs to the company as a whole and has dependencies between each other, suboptimization is likely to occur. If everyone in a department is working in the same way, and outer circumstances from customer or other departments make a deviation in the working processes, these deviations are easy to spot and attend to.

Lean Production Principle 14. Continual Organizational Learning through Kaizen

If everyone has their own working process which interacts with the system in their own way, ideas for improvement might involve a few, but not many. In the worst case scenario it could cause a decrease in efficiency for others. When

the process is standardized, it works as a foundation for improvements. When new ideas are developed, standardization makes it possible to evaluate the improvements with the communication flow kanban. It ensures that there are actual improvements, not suboptimization, by comparing the current feature/process to how it was before.

8.9.23 Done

This step clearly shows the employees what work is considered to be finished. It gives them the opportunity to backtrack and make sure they haven't missed any information due to events like vacation or sick leave. It is important for an employee to be able to visualize what has been done, because it creates a feeling of success and a sense that the company is moving forward.

8.10 Other Benefits From the Communication Flow Kanban

The definition of quality from production and development is different. In production, quality means no deviation from the standard. In development, quality is adapting and meeting the changing expectations of the many different customers. To support quality in both production and development in a fast changing environment like Telavox, it's good to change one thing at the time. Phases 1 through 7 described above pertain to the definition of quality from the development standpoint. They describe how to innovate by finding and enforcing new ideas for new features, developing them through testing, trying different solutions and allowing people to make mistakes.

When the best solution is found and developed, it is put into production. The production phase starts with number 8, to standardize. *One-piece flow* from the production industry could be the equivalent of *one-change flow* for organisations in change. Interviewing the team leaders made it apparent that when many changes occurred simultaneously, the cumulative time needed is greater than if the change occurred one by one. It is not just that change can be implemented quickly, the first change would also create value sooner. This is the same as with smaller batch sizes in the production industry, the first product can be delivered to the customer sooner. Smaller batch sizes could also grant higher quality, since when a change is standardized, it can be analyzed to see if the change is for better or worse. If the result is lower than before, it is easy to withdraw. If many changes occur at the same time, it is harder to ensure high quality. This is due to causality and the intercorrelated relationships between the changes.

8.11 Evaluation of the Communication Flow Kanban

Note that not all lean principles from lean software development and lean production were taken into account for the model. *Lean production principle 11 - Respect, channel and help your suppliers* and *lean production principle 8 - Use reliable, tested technology* were left out. When decisions were made and analysis performed on different suppliers or technology, it might have been possible to use the model. Using it at Telavox couldn't give sufficient

information to perform analysis on the principles, but it could have been implemented by the suppliers themselves, if it was to increase their processes.

All the teams in the advisor's department had a seminar about the communication flow kanban before they started using it. The feedback from the seminar was positive. Most of the employees liked that there would be more structure in sharing their ideas and that features were implemented one at a time. They saw the value of slowing down the decision-making phase to ensure consensus before implementation. The developers indicated that they liked to receive feature requests with epics containing the issues. This was mostly since the requests (with issues in the epic) were better organized and the communication between developers and advisors was vastly improved. Since the communication flow kanban was implemented, there were no single minded decisions.

Because communication flow kanban was only tested on one department of advisors, and no cross functional team of decision making managers existed, the decision phase of prioritizing between departments could not be evaluated. However, the Chief Product Officer (CPO) who decided on the priorities for the developers, gave positive feedback on the communication flow kanban. It made the managers' jobs easier as they received better insight into what needed to be done and a better overview of future tasks.

There were still some challenges that needed more time to overcome. The teams were supposed to go through the priorities and epics to reach a consensus on a weekly basis. This step was often overlooked due to time pressure (which risked the whole process failing), but did indicate a bottleneck from the advisors which had not been apparent without the communication flow kanban. Since there were no more tasks for the developers to work on (no more epics that reached *idea lock*), they had to start working on epics from the *idea phase*. This could create a negative spiral if the advisors couldn't catch up and come to a consensus quicker.

This could be an area for further improvements. Instead of having weekly meetings to go through and discuss the epics, it might be more efficient to have regular workshops where the most important epics are discussed. This would ensure that the developers always had work to do. It would also be a forum where the different teams in the department had an opportunity to discuss with each other directly, instead of through their team leader.

Another issue to consider was the standardization phase, and how it hadn't been evaluated thoroughly enough. The advisors were using the features that were developed for them, but the process to make sure that everyone was using them in the same way, and for the same purpose, had not been reviewed.

The conclusion was that the communication flow kanban worked well while implementing new features at Telavox. The goal of communication flow kanban was to implement lean in a service organisation. It looked like the model itself

was based on many lean principles. This was intriguing as it meant using lean principles could increase the ability to implement those lean principles. It was not tested, but could work in theory. For example, the stand up meetings could be decided and prioritised by the kanban. Discussion on how to establish them could take place in the idea phase, then a consensus reached before idea lock. The managers could give their approval and when everything was decided, it could be tested and evaluated for final standardization.

9 Discussion and Conclusion

This master's thesis attempted to find out how to implement lean. To do this it came up with a communication tool that could achieve a higher degree of success while still driving change. Using this model, the company would be successfully more lean. The model would hopefully work even better and drive change with an even higher degree of success. This creates a positive loop demonstrated by the diagram below.



Figure 27 The Lean Loop

Before implementing the communication flow kanban, all involved personnel received more information through seminars. This was also an opportunity for them to question the process and get answers. The implementation therefore differed from the other experiments made. Calculating waste, 5 why and stand up meetings were all decided and pushed down to the employees from the supporting manager. It was questioned whether the communication flow kanban worked better than the lean tools, or if it was the way it was implemented that caused it to succeed.

The communication flow kanban had mostly been tested and implemented in the development stage. However, it could have worked successfully (with some changes) in working processes as well. Most change in software development for internal tools also meant a change in process, and working processes could be decided by the employees, not just the managers (as the

communication flow kanban process suggests). Success in implementing stand up meetings and 5-why as a coaching tool could be achieved if they were implemented step by step. All subjects intended to be covered in stand up meetings could be implemented one at a time, and when they were standardized, the process would continue by implementing the next subject. Unfortunately, the implementation of the standup meetings as they first was planned was put on ice, even though the value of it was recognized by all of the team leaders later. With the communication flow kanban the possibility of success would have been higher.

The communication flow kanban stated that waste could be reduced, but it lacked significant proof of concept. There were indications from employees that some waste was reduced, but it couldn't be supported by quantified data or qualitative research. This was hard to achieve as only a few areas could be measured in a relatively short time frame, and some waste couldn't be measured as it was at the time. One area of future research could therefore be titled: *How to measure the impact of the communication flow kanban.*

It should be noted that the sandbox organisation, with the build-measure-feedback loop, was a copy of the model from lean startup, and was not implemented on Telavox. This thesis rather discusses the importance of a company not losing their capabilities from the startup years, an area which is hard to examine. Since these startup capabilities are reliant on only a few employees/owners, many of them might not diffuse through to the additional staff, especially when there is quick growth. Interviewing only one of the founders can result in a biased opinion, so deeper research should be undertaken with more newly employed staff.

While this thesis was written, the author had many discussions with friends and family about the concept of lean. There seemed to be much confusion as to what lean was all about. Some individuals from the healthcare industry look upon lean unfavorably. It was difficult to fully explain what it was all about since the most vital parts of lean are the culture and values that come with it. This was an area that was also hard to find in literature, or in the company, even though working at Telavox during the writing of this thesis helped understanding, and gave a greater time frame for analysis. Too late was it realized that lean principles were not just to be implemented according to a recipe book, but that the implementation needed to be adapted and evaluated based on every company's unique culture.

When implementing something new, it is important to set the expectations correctly. When explaining lean, it is important too use the right definitions and words so expectations are met. The left triangle in the diagram below has been presented earlier. It combines all the processes, tools and principles needed to be a successful and true lean company. However, for someone new to lean, who has never been working in a lean company, it might be better to explain these with the triangle to the right.

- The culture of the company can be seen in how the employees are trusted and empowered, and whether they are eager to always do their best and to continuously improve their work and the company. Healthy culture is reliant on a fun and happy environment for the employees and the respect of employees, suppliers and even competitors.
- The employees have knowledge and capabilities that always grow as long as they stay in the company. In larger companies with complex products/services, this can be a factor that causes them to stand out from their competitors. High knowledge and capability can result in better ideas for improvements, better coaching of personnel/new leaders/suppliers, better understanding of when to do what etc.
- Processes involve how the company accomplishes tasks and uses their personnel's abilities to maximum. It ensures a structured environment and makes sure that nothing is overlooked.
- Tools are supportive means to ensure high results. It can be for example what software is used to support the processes.

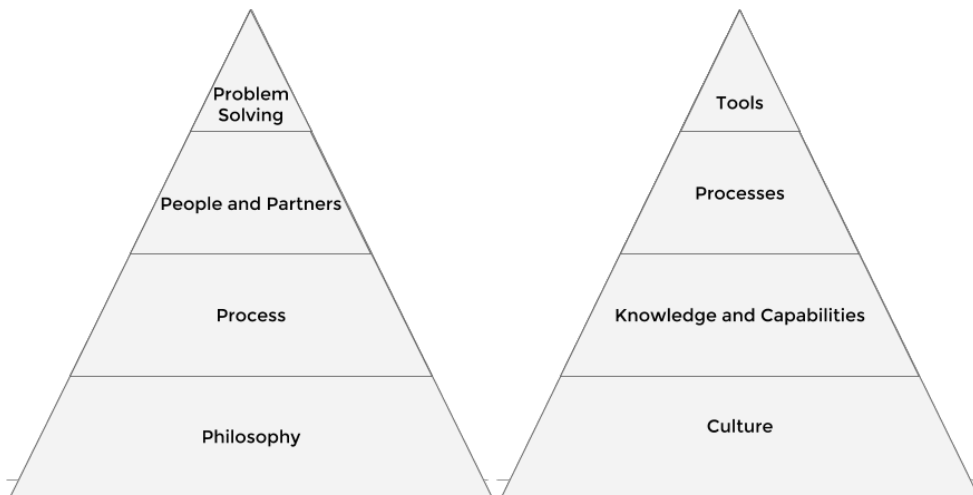


Figure 28 Rethink Lean

The point of bringing this up in the discussion is to emphasize how complex lean can be. This thesis presents two processes for incremental and radical innovation that can aid in better products/services or in achieving organisational change. However, how the culture and the employees knowledge/capabilities affect the implementation of lean might need to be looked into further. Though communication flow kanban worked well at Telavox, it is still uncertain whether it would work well in other companies. There is correlation between the culture, the willingness and drive to change at Telavox and the model itself which makes it hard to give conclusive evidence as to how the model could work in other companies.

10 Bibliography

Baldwin, H. (2012, July 24). Retrieved August 25, 2016, from Computer World: <http://www.computerworld.com/article/2506129/it-management/time-off-to-innovate--good-idea-or-a-waste-of-tech-talent-.html?page>

Blanchard International. (2007, March). Retrieved July 13, 2016, from <http://kenblanchard.com.au/Ignite/IgniteArticle.php?Doc=37>

Block, A. (2003). *Murphy's Law: The 26th Anniversary Edition*. New York: Perigee Trade.

Clark, K. B., & Takahiro, F. (1990, November-December). The Power of Product Integrity. *Harvard Business Review* .

D'onfro, J. (2015, April 17). Retrieved September 12, 2016, from Business Insider: <http://www.businessinsider.com/google-20-percent-time-policy-2015-4?r=US&IR=T&IR=T>

Drucker, P. F. (2006, February 1). *What Executives Should Remember*. Retrieved 11 11, 2016, from <https://hbr.org/2006/02/what-executives-should-remember>

Ely, R. J., & Meyerson, D. E. (2010). *An organizational approach to undoing gender: The unlikely case of offshore oil platforms*. Harvard: Elsevier Ltd.

Hastle, S., & Wojewoda, S. (2015, October 4). Retrieved September 27, 2016, from infoq: <https://www.infoq.com/articles/standish-chaos-2015>

Holland, K. (2013, September 13). *Eight Steps To Practical Problem Solving*. Retrieved July 19, 2016, from Kaizen-news: <http://www.kaizen-news.com/eight-steps-practical-problem-solving/>

Inman, R. A. (2006, February 23). *Management*. Retrieved June 9, 2016, from Reference for Business: <http://www.referenceforbusiness.com/management/Or-Pr/Poka-Yoke.html>

Larsson,j (2015), Sony Mobile [Lecture]

Liker, J. K. (2004). *The Toyota Way: 14 Management Principles from the World's Greatest Manufacturer*. New York: McGraw-Hill.

Locke, E. A. (1982). The Ideas of Frederick W. Taylor: An Evaluation. *Academy of Management Review* , 7 (1), 14-24.

Lotz, M. (2013, July 5). *Waterfall vs agile methodology*. Retrieved July 20, 2016, from Seguetech: <http://www.seguetech.com/waterfall-vs-agile-methodology/>

Minto, B. (2009). *The Pyramid Principle: Logic in Writing and Thinking*. London: Financial Times Prentice Hall.

Moore, G. A. (2002). *Crossing the Chasm: Marketing and Selling High-Tech Products to Mainstream Customers*. New York: HarperBusiness Essentials.

Optimizely. (n.d.). *Optimizely*. Retrieved July 25, 2016, from Split-Testing: <https://www.optimizely.com/split-testing/>

Ovum. (2014). Retrieved August 5, 2016, from <http://info.ovum.com/uploads/files/Ovum'Telecoms'Media'and'Entertainment'Outlook'2015.pdf>

Pay, R. (2008, March 1). *Industry Week*. Retrieved 07 04, 2016, from <http://www.industryweek.com/articles/everybodys'jumping'on'the'lean'bandwagon'but'many'are'being'taken'for'a'ride'15881.aspx>

Poppendieck, M., & Poppendick, T. (2003). *Lean Software Development An Agile Toolkit*. Boston: Addison-Wesley.

Reichheld, F. F. (2003, December). The One Number You Need to Grow. *Harvard Business Review* .

Ries, E. (2011). *The lean startup: How today's entrepreneurs use continuous innovation to create radically successful businesses*. New York: Crown Business.

Robehmed, N. (2013, December 16). *Forbes*. Retrieved July 4, 2016, from <http://www.forbes.com/sites/natalierobehmed/2013/12/16/what-is-a-startup/#24964ff44c63>

Rogers, E. M. (1983). *Diffusion Of Innovations* (Vol. 3). New York: A Division of Macmillan Publishing Co., Inc. .

Rosin, H. (2016, June 17). The New Norm. [Audio podcast], Retrieved from <http://www.npr.org/sections/healthshots/2016/06/17/482203447/invisibilia-how-learning-to-be-vulnerable-can-make-life-safer>

Scaled Agile, Inc. (2016, March 31). *Set Based Design*. Retrieved July 25, 2016, from Scaled Agile Framework: Original graphic modified by the author with permission from © 2011-2016 Scaled Agile, Inc. All rights reserved. Original graphic, Figure 1 of the SAFe® Set-Based Abstract, found at <http://www.scaledagileframework.com/set-based-design/>

Sobek II, D. K., Ward, A. C., & Liker, J. K. (1999, January 15). Toyota's Principles of Set-Based Concurrent Engineering. *MIT Sloan Management Review* .

Survival of the fittest. (2015). *Economist* , 415 (8934), 15-16.

Toyota. (2013, May 31). Retrieved July 15, 2016, from <http://blog.toyota.co.uk/nemawashi-Toyota-production-system>

Valve. (2012). *Valve Handbook For New Employees*. Valve.

Zaremba, M. (2013, March 03). *Kultur & Nöje*. Retrieved July 12, 2016, from Dagens Nyheter: <http://www.dn.se/kultur-noje/pa-vilken-prislista-stardin-njursvikt/>

Zaremba, M. (2013, February 25). *Kultur & Nöje*. Retrieved July 12, 2016, from Dagens Nyheter: <http://www.dn.se/kultur-noje/kulturdebatt/hur-mycket-bonus-ger-ett-benbrott/>

Appendix

10.1 Questions for interview with Filip Olsson (Asked in Swedish)

(All questions are relevant for the startup years in Telavox history)

How did Telavox work with the innovation process?

Did you do any MVP that you tested on customers?

Was many decisions done based on data in the first years of Telavox?

Did you Telavox do any strategical changes (pivots)?

What capabilities have Telavox kept from the startup years?

Have Telavox lost any capabilities to innovate?

Have you been involved in any project that you didn't let go off even though you should have? To make it more clear, did you have any problem with "kill your darlings"?

How was the life as an entrepreneur? How did it work with your private life?

Do you thing the life as an entrepreneur would have work for all kinds of people?(if not) for who wouldn't it work?

10.2 Questions for interview with Team Leaders (Asked in Swedish)

10.2.1 Intervjue 5-why question

Have you heard about 5-why before?

Do you understand what 5-why is about?(if not refresh the memory)

Have you personally used 5-why?

How do you find it to be "questioned" with 5-why?

How do you find it to be boomeranged questions?

Have you done any findings you wouldn't without thinking/been coached with 5-why?

Do you find 5-why to have any negative effects?

10.2.2 Visualization of organisational development processes

Do you remember what the board contained?

- Top of mind (check if remembered)

- Hidden failures (check if remembered)

- Weekly tasks (check if remembered)

- Happynings? (check if remembered)

In the different areas, what worked well and contributed to your work:

- Top of mind?

- Hidden failures?

- Weekly tasks?

- Happynings?

In the different areas what didn't work well and didn't contribute to your work.

- Top of mind?
- Hidden failures?
- Weekly tasks?
- Happynings?

Overall. What did you think about the board?

Did the weekly stand-up-meetings contribute in your work?

Was the stand-up-meeting motivational?

10.2.3 Creating Communication Flow

What do you remember from the seminar?

What did you think about the seminar?

What did you think about the principle with flow?

10.2.4 Organisational change

What “bumps” have you experienced when changing from operations to the advisor organisation?

What could have been better?

10.3 Calculating Waste – Data

10.3.1 Overall Results and Data for Delivery

Overall - results							
Number of working days	Working hours	Phonetime/working time	Waste time on outgoing phone calls	Waste time on phone calls	Total waste time of working time		
986	7888	9,43%	11,17%	6,95%	0,65%		

Date start	Date end	nbr of working days	nbr of working hours	Number of employees	Phone time/working time	waste time on phone	% of nbr waste calls
2015-01-01	2015-12-31	229	1832	2	33%	7,52%	56%

Delivery							
Type	Count of Length	Sum of Length	Average call time (s)	Average time until answered (s)	Value added count	Value added time	Time on Phone (h)
0,00	1016,00	0,00	0,00	20,00			5,64
Incoming	74,00	0,00	0,00	20,00			0,41
Outgoing	942,00	0,00	0,00	20,00	17%	1,47%	5,23
30,00	3372,00	29032,00	8,61	16,64			23,65
Incoming	1918,00	14252,00	7,43	16,64			12,83
Outgoing	1454,00	14780,00	10,17	16,64	27%	3,05%	10,83
60,00	964,00	41632,00	43,19	16,64			16,02
Incoming	317,00	14045,00	44,31	16,64			5,37
Outgoing	647,00	27587,00	42,64	16,64	12%	3,00%	10,65
90,00	603,00	44605,00	73,97	16,64			15,18
Incoming	258,00	19160,00	74,26	16,64			6,52
Outgoing	345,00	25445,00	73,75	16,64	6%	2,44%	8,66
120,00	501,00	52233,00	104,26	16,64			16,83
Incoming	249,00	25982,00	104,35	16,64			8,37
Outgoing	252,00	26251,00	104,17	16,64	5%	2,38%	8,46
180,00	776,00	115365,00	148,67	16,64			35,63
Incoming	415,00	61992,00	149,38	16,64			19,14
Outgoing	361,00	53373,00	147,85	16,64	7%	4,64%	16,49
240,00	577,00	119869,00	207,75	16,64			35,96
Incoming	306,00	63682,00	208,11	16,64			19,10
Outgoing	271,00	56187,00	207,33	16,64	5%	4,74%	16,86
360,00	711,00	209585,00	294,77	16,64			61,51
Incoming	402,00	118525,00	294,84	16,64			34,78
Outgoing	309,00	91060,00	294,69	16,64	6%	7,52%	26,72
480,00	428,00	177064,00	413,70	16,64			51,16
Incoming	249,00	103551,00	415,87	16,64			29,92
Outgoing	179,00	73513,00	410,69	16,64	3%	5,98%	21,25
600,00	226,00	120708,00	534,11	16,64			34,57
Incoming	124,00	66279,00	534,51	16,64			18,98
Outgoing	102,00	54429,00	533,62	16,64	2%	4,39%	15,59
600+	847,00	1103571,00	1302,92	16,64			310,46
Incoming	313,00	339722,00	1085,37	16,64			95,81
Outgoing	534,00	763849,00	1430,43	16,64	10%	60,40%	214,65
Grand Total	10021,00	2013664,00	200,94	16,64	100%	100,00%	605,68

10.3.2 Data for Team Customer Care

Date start	Date end	nbr of working days	nbr of working hours	Number of employees	Phonetime/working time	waste time on phone	% of nbr waste calls
2015-11-02	2015-12-19	35	280	2	39%	18,42%	69%
Team Customer Care							
Type	Count of Length	Sum of Length	Average call time (s)	Average time until answered (s)	Value added count	Value added time	Time on phone (h)
0	712	0	0,0	20,00			3,96
Incoming	7	0	0,0	20,00			0,04
Outgoing	705	0	0,0	20,00	22,93%	4,12%	3,92
30	1185	11349	9,6	16,64			8,63
Incoming	145	1562	10,8	16,64			1,10
Outgoing	1040	9787	9,4	16,64	33,82%	7,92%	7,53
60	393	16911	43,0	16,64			6,51
Incoming	27	1179	43,7	16,64			0,45
Outgoing	366	15732	43,0	16,64	11,90%	6,38%	6,06
90	185	13676	73,9	16,64			4,65
Incoming	16	1220	76,3	16,64			0,41
Outgoing	169	12456	73,7	16,64	5,50%	4,46%	4,24
120	131	13429	102,5	16,64			4,34
Incoming	10	1012	101,2	16,64			0,33
Outgoing	121	12417	102,6	16,64	3,93%	4,22%	4,01
180	177	26211	148,1	16,64			8,10
Incoming	12	1800	150,0	16,64			0,56
Outgoing	165	24411	147,9	16,64	5,37%	7,94%	7,54
240	135	28464	210,8	16,64			8,53
Incoming	13	2573	197,9	16,64			0,77
Outgoing	122	25891	212,2	16,64	3,97%	8,16%	7,76
360	193	56773	294,2	16,64			16,66
Incoming	20	6237	311,9	16,64			1,82
Outgoing	173	50536	292,1	16,64	5,63%	15,61%	14,84
480	99	41223	416,4	16,64			11,91
Incoming	17	7148	420,5	16,64			2,06
Outgoing	82	34075	415,5	16,64	2,67%	10,36%	9,84
600	51	26896	527,4	16,64			7,71
Incoming	8	4416	552,0	16,64			1,26
Outgoing	43	22480	522,8	16,64	1,40%	6,78%	6,44
600+	109	98762	906,1	16,64			27,94
Incoming	20	17929	896,5	16,64			5,07
Outgoing	89	80833	908,2	16,64	2,89%	24,06%	22,87
Grand Total	3370	333694	99,0	16,64	100%	100,00%	108,27

10.3.3 Data for Economy

Date start	Date end	nbr of working days	nbr of working hours	Number of employees	Phone time/working time	waste time on phone	% of nbr waste calls
2015-01-01	2015-12-31	229	1832	2	2%	62,17%	61%
Economy							
Type	Count of Length	Sum of Length	Average call time (s)	Average time until answered (s)	Value added count	Value added time	Time on phone (h)
0	583	0	0	20			3,24
Incoming	153	0	0	20			0,85
Outgoing	430	0	0,00	20,00	17,19%	19,97%	2,39
30	1801	13511	7,50	16,64			8,33
Incoming	935	6318	6,76	16,64			4,32
Outgoing	866	7193	8,31	16,64	34,63%	33,47%	4,00
60	561	24752	44,12	16,64			2,59
Incoming	335	14936	44,59	16,64			1,55
Outgoing	226	9816	43,43	16,64	9,04%	8,73%	1,04
90	529	39216	74,13	16,64			2,45
Incoming	382	28478	74,55	16,64			1,77
Outgoing	147	10738	73,05	16,64	5,88%	5,68%	0,68
120	471	49375	104,83	16,64			2,18
Incoming	342	35958	105,14	16,64			1,58
Outgoing	129	13417	104,01	16,64	5,16%	4,99%	0,60
180	707	104005	147,11	16,64			3,27
Incoming	555	81825	147,43	16,64			2,57
Outgoing	152	22180	145,92	16,64	6,08%	5,87%	0,70
240	470	97330	207,09	16,64			2,17
Incoming	341	70700	207,33	16,64			1,58
Outgoing	129	26630	206,43	16,64	5,16%	4,99%	0,60
360	529	155100	293,19	16,64			2,45
Incoming	402	117355	291,93	16,64			1,86
Outgoing	127	37745	297,20	16,64	5,08%	4,91%	0,59
480	260	108030	415,50	16,64			1,20
Incoming	185	76982	416,12	16,64			0,86
Outgoing	75	31048	413,97	16,64	3,00%	2,90%	0,35
600	141	75059	532,33	16,64			0,65
Incoming	90	47970	533,00	16,64			0,42
Outgoing	51	27089	531,16	16,64	2,04%	1,97%	0,24
600+	328	426213	1299,43	16,64			1,52
Incoming	159	187407	1178,66	16,64			0,74
Outgoing	169	238806	1413,05	16,64	6,76%	6,53%	0,78
Grand Total	6380	1092591	171,25	16,64	100%		29,50