Lean thinking principles for cloud software development

Executive Summary

This research summary focuses on the principles and concepts of lean thinking. In addition, the link between the cloud phenomenon and lean thinking was drawn. As was evident from this research summary, concepts, cloud and lean, are difficult to describe explicitly. Cloud is one of the main global trends washing through the ICT industry, bringing in changes to the current technological and business environment. Even though cloud is referred in many contexts, it does not yet have a clear and complete definition in literature. In this paper, we consider cloud more widely than just cloud computing. We believe that lean thinking is one of the most powerful means to help companies to survive and create a competitive advantage in the cloud business environment.

We believe that the ICT industry is currently at the point of its greatest opportunity to utilise the opportunities brought about by the cloud phenomenon. With the help of lean thinking, companies can answer the challenges and capitalise on the new opportunities.

No doubt, cloud increases the complexity in a technical and business environment and it influences in many different ways software intensive companies. The lean principles of software development – eliminate waste, build quality in, create knowledge, defer commitment, deliver fast, respect people, and optimise the whole – create a sound basis for contemporary enterprises. Although, these principles are old and well-known truths, their importance is fore-grounded in the cloud business environment. cloud phenomenon.

Table of contents

Introduction

Cloud is one of the main global trends washing through the ICT (Information and communication technologies) industry, wave upon wave, bringing in changes in external environments such as new technologies, business models, new legislation, open interfaces, security/privacy challenges, user communities, social networks and new ecosystems.

VTT Technical Research Centre of Finland observes that the capacity of an organisation to benefit from the cloud depends on its ability to adapt the entire organisation to the change. In other words, business gains only arise by migrating to a better, more effective production system, which utilises the full potential of the cloud. This includes not only integrating new technologies, such as cloud computing, to the software production but a great deal more. Lean thinking is seen as a key approach proactively responding to the issues and opportunities of the cloud.

Revolutions in software development practices don't just happen. In lean thinking, the ICT industry can learn from other business fields. Lean thinking is a proven paradigm that applies in product development and production and there are a large number of success stories experienced and reported from the manufacturing industries.

The purpose of this research summary is to draw a common thread between the cloud phenomenon and lean thinking. Additionally, the purpose is to briefly summarise the history and principles, as well as some experiences of the lean transformation and the main methods used in some cases reported in literature. The overall aim is to provide readers with background information for lean transformation process within their own organisations.

The work is supported by Tekes (the Finnish Funding Agency for Technology and Innovation, www.tekes.fi) as a part of Cloud Software Program (www.cloud-softwareprogram.org) of Tivit (Strategic Centre for Science, Technology and Innovation in the Field of ICT, www.tivit.fi).

Cloud and lean – contemporary paradigms

"Cloud" and "lean" are contemporary paradigms. Some also say that they are the hype words of today. This chapter provides a short lead-in to cloud and lean paradigms and explains why they are intertwined together.

Cloud phenomenon

At present, there are a large number of discussions related to clouds and cloud computing in particular. Vaquero et al. (2009, p.59) define clouds as follows: *"Clouds are a large pool of easily usable and accessible virtualized resources (such as hardware, development platforms and/or services). These resources can be dynamically re-configured to adjust to a variable load (scale), allowing also for an* optimum resource utilization. This pool of resources is typically exploited by a pay-per-use model in which guarantees are offered by the Infrastructure Provider by means of customized SLAs".

At VTT, we believe that the cloud phenomenon is larger than the technically oriented definition above. We think that the cloud phenomenon also has an impact on other areas of businesses and brings changes to the external business environment. Examples of these are new business models, legislation, open interfaces, security and privacy challenges, user communities, social networks and new ecosystems. These changes will present both opportunities and threats, with some possessing the potential to radically alter how an ICT company conducts its business. In general, "cloud forces" are beyond the direct control of an organisation. Instead, the companies must recognise the cloud as something to be prepared for and responded to. In the Cloud Software Program's strategic research agenda (SRA) (Tivit, 2009), the main Finnish ICT companies identified the following key cloud trends:

- **User experience (UX):** In the future, people will be surrounded by a growing amount of interactive, easily available products and services. User expe-

rience and value for the customers will be the main competitive factor for ICT companies.

- **Web'd software:** Web'd software is different in several ways from conventional software. In service developments that reuse codes and contents from other web sites, the Web essentially becomes a distribution media for software components consisting of objects and codes located anywhere in the world. The transition towards web-based software will dramatically change the way people develop, deploy and use software, making software distribution as easy as the distribution of documents and other services on the Web already is.
- **Openness:** Openness is an ongoing trend that drastically changes the landscape of software engineering and the development of software-intensive systems and services. The change radically affects existing business ecosystems. Free components are available for anyone to use. This also, however, threatens the businesses based on proprietary, closed solutions by a potential devaluation of the software assets by providing the customers with free alternatives. Open source provides communities of developers as a workforce to the development of software that they find interesting and meaningful to build.
- **Security:** Security and related issues have become a core issue of high practical relevance in software development. The lack of appropriate security solutions in the cloud might have serious consequences to the business.
- **Operational efficiency:** The critical starting point in successful business in the cloud is operational efficiency, which in turn means e.g. productivity, quality, effective processes, the automation of routine tasks, customer focus and enterprise-wide lean processes and methods.
- **Sustainable development:** The focus on sustainable development is an emerging strong trend. Consumers and businesses are becoming more aware of environmental issues, such as global warming and pollution.
- **New software business models:** The software's impact on different products, systems and services has grown rapidly. This has led to an enormous increase in the software complexity, as well as increasing the demand for changes in software business models. Software intensive companies face a large number of challenges in growing their business in today's global, fast-moving and cloudy industry environment.

Lean thinking

The capability of an organisation to benefit from the cloud is dependant on its ability to adapt the entire organisation to the change. In other words, business gains only come by migrating to a superior, more effective production system, which utilises the full potential of the cloud. This not only includes integrating new technologies such as cloud computing to the software production, but a great deal more.

According to Treacy and Wiersema (1993), an organisation must focus on the three **value disciplines**; operational excellence, customer intimacy, and product leadership, in order to success. **Operational excellence** means providing customers with reliable products or services at competitive prices and delivered with minimal difficulty or inconvenience. **Customer intimacy** means segmenting and targeting markets precisely and then tailoring offerings to exactly match the demands of those niches. **Product leadership** means offering customers cutting-edge products and services that consistently enhance the customer's use or application of the product, thereby making their rivals' goods obsolete. Womack and Jones's (1996) definition of lean thinking is aligned with the value disciplines. According to them, lean thinking **"provides a way to do more and more with less and less – less human effort, less equipment, less time, and less space – whilst coming closer and closer to providing customers with exactly what they want".**

Lean thinking is seen as a key approach which proactively responds to the issues and opportunities of the cloud. The link between the cloud opportunities and threats and the internal environment of a company is illustrated in figure 1.

What can an ICT company do to benefit from the cloud and make significant progress toward the perfect software and value creating processes? The first steps are as important as ever 1) recognise the impact of the cloud on your company and

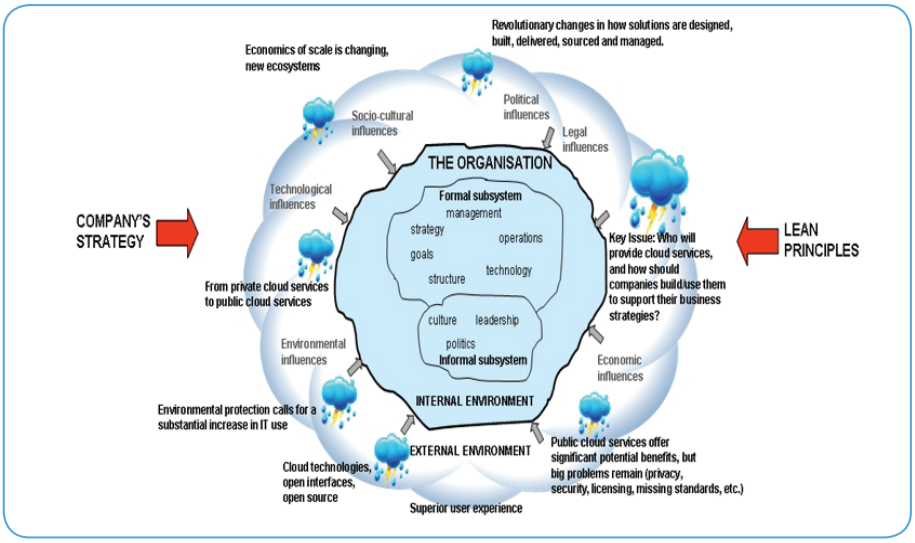

Figure 1: Cloud trends and Organization (modified from Senior & Fleming (2006).

2) identify the key improvement areas in the organisation's formal and informal subsystems.

However, the basis for the lean transformation is to understand the background and main principles of the lean thinking. The purpose of the next chapter, The history of lean thinking, is to clarify where the origins of the lean are. Chapter 4, 'lean principles and concepts, summarises the different key principles and concepts of the lean thinking. Chapter 5, Lean transformation and methods, briefly summarises proposed frameworks for transforming an enterprise towards lean. Chapter 5 also raises some interesting topics related to the lean paradigm, such as the benefits and costs of the lean, a comparison of the lean and agile, and some experiences so far on applying lean thinking in industries. Finally, in chapter 6, conclusions and recommendations are given.

The history of lean thinking

Early steps

The roots of lean thinking go back to the 1700s. At the end of the 1790s, Eli Whitney perfects the concept of interchangeable parts for guns for the U.S. Army. Although Whitney had problems with the contract, he is considered to be a developer of a manufacturing system in which semi-skilled workers use machine tools to make standardised parts that are assembled into products. (e.g. Poppendieck & Poppendieck, 2007).

During the 1800s, manufacturers concern themselves with individual technologies. In the late 1800s, Frederick W. Taylor began to look at individual workers and working methods. Taylor was the developer of Taylorism (also called scientific management) that dominated in the 1910s. Taylorism (Taylor, 1911) is a variation on the theme of efficiency; increasing efficiency, decreasing waste, and using empirical methods to decide what matters.

In 1914, Henry Ford creates the first moving assembly line, reducing assembly times remarkably. Ford raised the wages of workers, and he dropped the price of the car, which were possible due to the advantages enabled by the new, efficient assembly line. The era of mass production had started. (Poppendieck & Poppendieck, 2007; Liker, 2004). The Ford system managed the flow well, but problems arose when variety was desired; customers requested different colours and different models. Ford did not respond to the new requirements as other automakers did, and thus lost its dominance over the market. Other automakers responded to the need for additional models, but with production systems whose design and fabrication steps regressed toward process areas with much longer throughput times (LEI, 2010).

The Toyota Production System

The Toyota Production System (TPS), also referred to as a **kanban**, evolved out of need. The post World War II Japanese automobile industry could not compete with U.S. industries in terms of producing automobiles. Whilst U.S. companies focused on the mass production of a few automobile types, the Japanese industries had to find a way to cut costs by producing small amounts of many car types. The approach of producing many car models in small quantities was the principal objective of the TPS. The essence of TPS is the absolute elimination of waste (Ohno, 1988). The TPS can be seen as the origins of lean manufacturing and a great deal of the elements of lean are based on the ideas of the TPS.

The two central elements to support the TPS system are **just-in-time** and autonomation (automation with a human touch). Just-in-time in the TPS means that the right parts are implemented as they are needed and only in the quantities required. The Just-in-time approach is implemented using a **kanban system**. Kanban is a signal indicating what, for example parts, need to be implemented, including the amount of requested items. The kanban system is the means for a smooth operation and flow, in the TPS. (Ohno, 1988)

Autonomation is a means for preventing the production of defective goods. The idea stems from the operating principles of high performance machines, capable of producing significant amounts of goods. These machines often operate whether or not the actual quality of the products they are processing is satisfactory. Autonomation refers to machines which are capable of identifying problems. If problems occur, the machine is automatically stopped, thus preventing the implementation of defective goods. (Ohno, 1988)

Based on a longitudinal study by Spear and Bowen (1999), the essence of the TPS in fact lies in four rules, instead of tools and practices. These rules are briefly as follows:

1. All work is highly specified in its content, sequence, timing, and out come.
2. Each worker knows who provides what to him, and when.
3. Every product and service flows along a simple, specifiedpath.
4. Any improvement to processes, worker/machine connections, or flow paths must be made through the scientific method and at the lowest possible organizational level.

In 1990, Womack, Jones, and Roos published the book "The Machine That Changed the World" (Womack, et al., 1990), which gave a new name to what had been called Just-In-Time or the Toyota Production System (Poppendieck & Poppendieck, 2007). From then on, Toyota's approach to manufacturing became known as lean production.

According to Poppendieck and Poppendieck (2007), lean thinking has nowadays clearly extended from lean manufacturing to lean enterprise. Similarly, the lean ideas have been expanded from traditional manufacturing to other operational areas, such as supply chain, lean product development and **lean software development.**

Lean principles and concepts

The main objective of this research summary is to clarify the basic principles and concepts of lean thinking, in order to provide a basis for the lean transformation process within the organisations.

The references related to the lean principles and main concepts are various books, articles, research summarys and other public material. Several different sets of lean principles are available, each of them presenting principles from different viewpoints and using slightly diverse wording. However, the main vision of the lean thinking seems to remain the same: "more and more with less and less – less human effort, less equipment, less time, and less space – whilst coming closer and closer to providing customers with exactly what they want" (Womack & Jones, 1996).

In the following sections, the best known sets of lean principles and key concepts are summarised.

The Toyota way

The roots of **the Toyota way principles** can be traced back to the very beginnings of the company. The Toyota way consists of the foundational principles of the Toyota culture, which allows the TPS to function effectively. TPS is not a toolkit and not a set of lean tools, but it is a sophisticated system of production in which all of the parts contribute to the whole. The whole at its roots focuses on supporting and encouraging people to continually improve the processes they work on. TPS is about applying the principles of the Toyota way (Liker, 2004), as outlined in the following table 1.

Based on the Toyota way principles above, Liker (2004) defined the following 14 principles of the Toyota way (table 2).

Table 1: Toyota way principles

Continuous Improvement	Respect for People
Challenge: We form a long-term vision, meeting challenges with courage and creativity to realise our dreams. Kaizen: We improve our business operations continuously, always driving for innovation and evolution. Genchi Genbutsu: Go to the source to find the facts to make correct decisions.	Respect: We respect others, make every effort to understand each other, take responsibility and do our best to build mutual trust. Teamwork: We stimulate personal and professional growth, share the opportunities of development and maximise individual and team performance.

Table 2: Liker's (2004) Toyota way principles

1	Base your management decisions on a long-term philosophy, even at the expense of short-term financial goals	8	Only use reliable, thoroughly tested technology that serves your people and processes.
2	Create a continuous process flow to bring problems to the surface	9	Nurture leaders who thoroughly understand the work, live the philosophy, and teach it to others.
3	Use "pull" systems to avoid overproduction	10	Develop exceptional people and teams who follow the philosophy of your company.
4	Level out the workload	11	Respect your extended network of partners and suppliers by challenging them and helping them improve
5	Encourage a culture of stopping to fix problems, to get the quality right the first time.	12	Go and see yourself to thoroughly understand the situation.
6	Standardised tasks are the foundation for continuous improvement and employee empowerment	13	Make decisions slowly by consensus, thoroughly considering all options; implement decisions rapidly.
7	Use visual controls, so no problems are hidden	14	Become a learning organization through relentless reflection and continuous improvement.

Lean principles at the enterprise level

In their book "lean thinking", Womack and Jones (1996) widened the scope of lean thinking from lean manufacturing to lean enterprise. They presented the following stripped-down lean thinking principles, table 3.

The lean advancement initiative (LAI) at the Massachusetts Institute of Technology (MIT) has developed the lean enterprise model (LEM), which is a systematic framework that encompasses lean enterprise principles, metrics, overarching practices and enabling practices to help organizations identify and assess the leanness of their own organization and processes. According to the MIT authors, the approach provides an equal balance of people-oriented and process-oriented practices that help chart a path to becoming a lean enterprise. MIT's lean enterprise model is based on the following 7 principles of lean enterprise thinking (Nightingale, 2009), table 4.

Lean principles in software development

Middleton and Sutton (2005) studied challenges to the software development and how the challenges are being met by the lean management principles. They began with the five principles of Womack and Jones (table 3) and applied them to software developments. They claimed that the software industry is the most suited of all the industries to transform to the lean production.

At roughly the same time as Middleton and Sutton (2005), Poppendieck and Poppendieck (2003, 2007) studied lean principles in software development. During 2002 – 2007, they concluded with the following seven principles of lean software development, table 5.

Main lean concepts: value and waste

The most important starting point for lean thinking is the value and the value can only be defined by the customer (Womack & Jones, 1996).

Table 3: Lean thinking principles by Womack and Jones (1996)

1	Identify the value.
2	Map the value stream.
3	Create the flow.
4	Establish the pull.
5	Seek perfection.

Table 4: MIT's lean enterprise thinking principles (Nightingale, 2009)

1	Adopt a holistic approach to enterprise transformation.
2	Identify relevant stakeholders and determine their value proposition.
3	Focus on enterprise effectiveness before efficiency.
4	Address internal and external enterprise interdependencies.
5	Ensure stability and flow both within and across the enterprise
6	Cultivate leadership to support and drive enterprise behaviours
7	Emphasise organizational learning.

Table 5: Lean principles in software development (Poppendieck & Poppendieck, 2007)

1	Eliminate waste.
2	Build quality in.
3	Create knowledge.
4	Defer commitment.
5	Fast delivery.
6	Respect people.
7	Optimise the whole.

Waste (Muda, Mura, Muri)) is something that does not create value (muda), unevenness or variability in the workflow (mura) and overburdening equipment or employees (muri). Womack and Jones (1996) further elaborate the concept of waste, stating that it is specifically any human activity that consumes resources but does not create value.

Russel and Taylor (1999) stated that waste is anything other than the minimum amount of effort, parts, equipment, materials, parts, space and time essential to add value to a product. The TPS (Ohno, 1988) defines seven types of wastes, which are briefly described in table 6 below (Hines & Rich, 1997). The seven wastes are not independent from each other; rather, they are largely intertwined which suggests that the wastes should be tackled together instead of focusing on one particular waste alone.

Womack and Jones (1996) identified two different kinds of wastes; Type One and Type Two *muda*. ***Type One muda*** involves non-value producing but unavoidable actions dictated by the operation environment, whilst ***Type Two muda*** creates no value and can be immediately avoided. For analysis and necessary actions, Monden (1993) categorised these wastes as follows:

1. non-value adding (NVA);
2. necessary, but non-value adding (NNVA);
3. value-adding (VA).

The original seven wastes were identified in the context of an environment producing physical goods. Later, Poppendieck and Poppendieck (2003, 2007) identified wastes specific in software development.

Table 6: Seven types of waste (Hines & Rich, 1997)

1	Overproduction; the most serious waste, discouraging a smooth flow and inhibits quality and productivity.
2	Waiting; this waste results from an inefficient use of time.
3	Transport; goods being transported between different locations. In addition, excessive movements are likely to cause damage. Furthermore, this waste relates to communication between processes; the longer that the distance is, the more communication and more time is required between feedback reports on poor quality and corrective actions.
4	Inappropriate processing; results from applying overly complex solutions. Since investing into a complex solution can be costly, this waste usually results in overproduction, through which the investment costs of the machine are attempted to be recovered.
5	Unnecessary inventory; increases the lead time, prevents the rapid identification of problems and increases space, and thereby discourages communication. In short, the inventory hides problems.
6	Unnecessary motion; this waste is related to the ergonomics of the work, how people have to move in order to complete their work. Unnecessary motion can be tiring and result in poor productivity and also often in poor quality.
7	Defects; this waste is seen as the bottom-line waste, as they are direct costs.

Table 7: Wastes in software development (Poppendieck & Poppendieck, 2003; 2007)

1	Partially done work: Something that is not done. E.g. untested codes, undocumented or not maintained business decisions
2	Extra features: Something that is not really required
3	Relearning (a waste of knowledge): E.g. forgetting decisions, re-trying already attempted solutions, the inability to utilise the knowledge of other people.
4	Handoffs: Passing the information/work to someone else, getting information/work from someone else. Handoffs may cause the loss of tacit information and information distortion.
5	Task switching: How many other tasks do people need to do? E.g. the amount of projects performed simultaneously.
6	Delays: Waiting for something.
7	Defects: Something that does not meet the targets, or is not what it is supposed to be. E.g. software bugs, incorrectly implemented business requirements.

Lean transformation and methods

The purpose of this chapter is to describe the change process at a general level; what steps have to be taken, what the timeframe of the change is, and what the requirements for a successful change are. Additionally, value stream mapping is described, lean and agile are compared, and the benefits and costs of lean are discussed. Finally, some experiences of deploying lean principles in software development are provided.

The change process towards lean

One of the lean principles highlights the importance of seeing the whole, optimising the whole. Thus, if a company has decided to apply lean thinking in its operations, the transformation towards lean needs a holistic approach, and the result will be a lean enterprise, not just lean product development or lean supply chains. Smeds (1994) writes about the change management towards lean enterprises. She claims that the evolution of an enterprise towards a lean enterprise cannot be managed with a top-down development strategy, but also a bottom-up approach is needed. Smeds (1994) continues that an emergent strategy can be guided by an overall umbrella strategy, where the management sets out the vision and the broad guidelines for a strategy; detailed specifics can be created at lower levels in the organization. Johnson (1992) agrees with the view and emphasizes that transformation efforts must be both top-down and bottom-up. Smed's (1994) change management framework for lean transformation is described in figure 2.

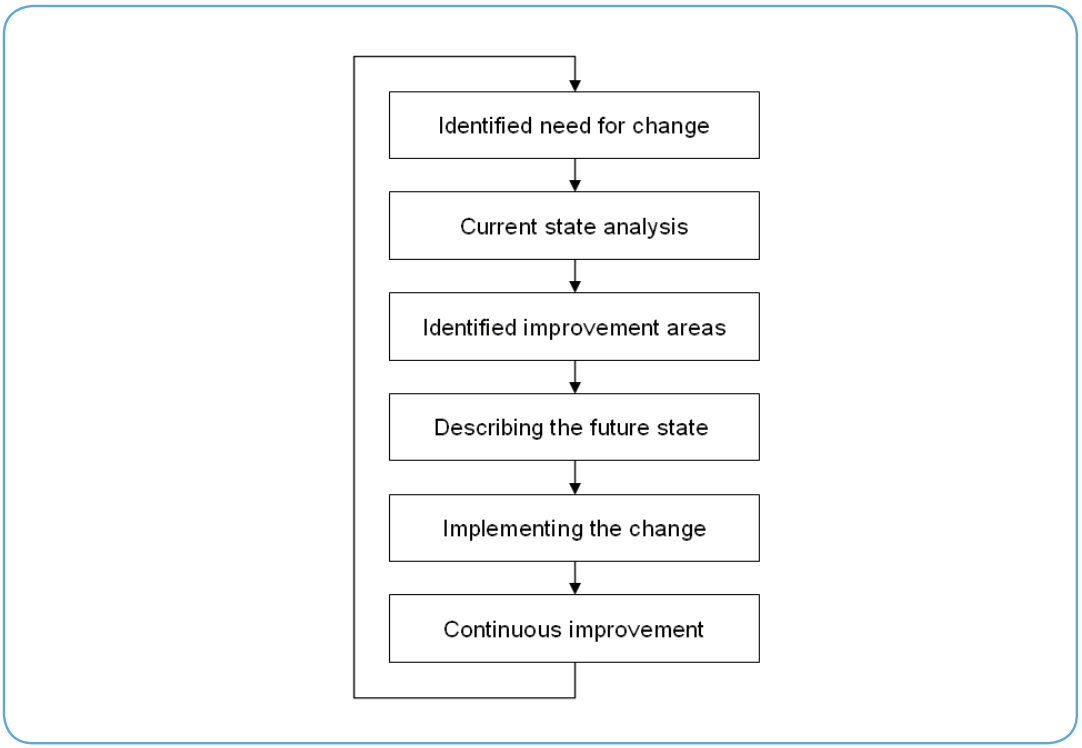

Figure 2: A Change Management Framework for Lean Transformation (modified from Smeds, 1994).

Lean transformation is a long-term activity which requires even years to be completed. Hines, et al. (2004) summarises Womack's and Jones's (1996) time frame for the lean transformation, as shown in the table 8.

Although there are step-to-step instructions for lean transformation, lean is more of a philosophy than a strategy (e.g. Bhasin & Burcher, 2006). Lean thinking requires a holistic perspective and should cover all the processes and subsystems. Bhasin and Burcher (2006) claim that lean needs to be seen as a journey and as a mind-set that governs how one looks at the business and processes. For a successful lean transformation, Bhasin and Burcher (2006) list the following requirements (table 9):

MIT researchers (2000) have created the "Transition-to-lean roadmap", which is aimed at assisting firms in their efforts to transform into lean. The framework includes the steps necessary to initiate, sustain, and continuously refine an enterprise

Table 8: Lean transformation steps with a timeframe (Womack & Jones, 1996; Hines, et al., 2004).

Phase	Specific steps	Time frame
Get started	1. Find a change agent 2. Gain lean knowledge 3. Find a lever 4. Map value streams 5. Begin a *kaikaku* 6. Expand your scope	The first six months
Create a new organization	7. Reorganise the product family 8. Create a lean function 9. Devise a policy for excess people 10. Devise a growth strategy 11. Remove anchor-draggers 12. Instil a "perfection" mindset	Six months through to year two
Install business systems	13. Introduce lean accounting 14. Relate pay to firm performance 15. Implement a transparency 16. Initiate a policy deployment 17. Introduce lean thinking 18. Find the right-sized tools	Years three and four
Complete the transformation	19. Apply these steps to your suppliers/customers 20. Develop a global strategy 21. A transition from top-down to bottom-up improvement	By the end of year five

Table 9: Requirements for a successful lean transformation (Bhasin & Burcher, 2006)

1	Simultaneously apply five or more of the technical tools
2	View lean as a long term journey
3	Install a continuous improvement viewpoint
4	Make numerous cultural changes embracing empowerment
5	Sponsor the lean principles through-out the value chain

Table 10: Main characteristics of lean and agile (Bozdogan, 2006)

Model key dimensions	Lean thinking	Agile approach
Expected outcomes	• Creating value for multiple enterprise stakeholders. • Building a dynamic competitive advantage.	• Enhancing enterprise competitiveness, to thrive in a new, fast-paced, uncertain environment
Defining features	• Mutually reinforcing a set of principles, practices and methods for evolving efficient, value creating and healthy enterprises.	• Attempt to definine the next industrial paradigm beyond lean ideas and flexible production systems.
Core concepts	• Deliver the best lifecycle value to the customers • Eliminate waste with the goal of creating value • Pursue knowledge driven change optimizing peoples capabilities • Evolve a robust and adaptive enterprise	• Anticipate and meet customer needs; create a new market and profit opportunities • Evolve an adaptive, flexible and efficient manufacturing system • Establish cooperative relationships with other enterprises as required
Focus	• Concentrate on the entire enterprise value stream • Focus on identifying enterprise stakeholders and defining their value expectations, constructing robust value propositions, and delivering value continuously by eliminating waste and striving for perfection	• Concentrate on an effective enterprise integration to support manufacturing • Focus on delivering high-quality, low-cost and innovative new products through continuous process improvement • Form virtual organizations to reduce cycle-times and costs
Change process	• Evolutionary systemic change	• Continuous incremental, as well as radical, change
History	• Since the late 1940s	• Since the early 1990s

transformation based upon lean principles and practices. This model was developed from an enterprise perspective.

A so-called Leffingwell model was depicted in the research summary – 'Lean and Scalable Requirements Information Model for the Agile Enterprise' written by Dean Leffingwell and Juha-Markus Aalto (Aalto, 2008). In this research summary, authors describe a model that extends the basic team-based agile requirements practices to the needs of the largest, lean thinking software enterprise. The foundation of the model is claimed to be essentially lean and an agile subset in support of the agile project teams that write and test all the code.

Lean and agile

In recent years, agile software development methods (Abrahamsson, et al., 2002) have been adopted widely by ICT companies (e.g. Dybå & Dingsoyr, 2008). Agile software methods refer to a group of different methodologies and methods such as Scrum (Schwaber & Beedle, 2002) or XP (Beck & Andres, 2005), which are developed for the iterative, light-weight processes. In addition, the agile approach emphasizes the importance of team work, rapid delivery, an alignment between customer needs and company goals (e.g. Abrahamsson, et al., 2002).

In literature, there is discussion on the relationship between lean thinking and the agile approach. Although Dybå and Dingsoyr (2008) consider lean software development to be one agile development method, others may see the relationship in the opposite way or that agile and lean paradigms do no exclude but complement each other. For example, Perera and Fernando (2007) report a study conducted in software development, where they combine agile and lean paradigms. As a conclusion, they reveal that lean principles can be used to improve the agile paradigm, which is criticised of relying on project team members with expert knowledge, of missing a defined, systematic project management approach, and of scaling limitations.

Parnell-Klabo (2006) reports insights gained from a pilot effort applying lean principles in conjunction with the adaptation of agile techniques. The report tells that the pilot project started with creating a value stream map, which focuses on customer values, and then the project employed lean principles, especially the "eliminate waste" principle, which revealed an improvement agenda. The next step was to apply Scrum agile development methodology. The pilot project produced impressive results. The duration of the project decreased by 40% compared to the waterfall baseline estimate. The actual resource costs were 10% under the allocated budget. No degradation of quality in the code was found. Additionally, according to the authors, an increase in morale was achieved.

Bozdogan (2006) provides a comparative review of a number of large-scale enterprise change models including lean thinking, six sigma, total quality management (TQM), reengineering, agile manufacturing, and the theory of constraints. A main conclusion is that lean thinking provides by far the most compelling intellectual architecture for the various systemic change initiatives. The following table, table 10, summarises the main characteristics of lean thinking and agile, as described by the Bozdogan (2006).

Value Stream Mapping

The identification of the entire value stream for each product or service is a key task in the lean transformation process. ***Value Stream Mapping (VSM)*** is the technique of mapping each individual step in the production, from a customer request to the completed product. VSM aims to identify those steps in the production that create customer value, do not create customer value but are necessary in the current production environment and those that are considered to be non-value adding activities that can be eliminated immediately (see section 4.4).

VSM was initially developed in 1995 in order to identify waste and remove it by using a collection of different tools (Hines, et al., 1998). According to Womack and Jones (1996), a value stream is *"a set of all the specific actions required to bring a specific product through the three critical management tasks of the business".* The main actions are:

Problem solving task	Running from the concept through the detailed design and engineering to the production launch.
Information management task	Running from order-taking through the detailed scheduling to the delivery
Physical transformation task	Proceeding from raw materials to a finished product in the hands of the customer

Hines, et al. (1999) state that VSM is an internal benchmarking tool which compares the initial performance of a particular process internally with how good that process itself could be. Thus, it compares the wasteful and value adding activities at the given moment with the possible future process from which a realistic percentage of the waste is removed (Hines, et al., 1999).

Abdulmalek and Rajgopal (2007) summarise three different steps in the mapping of value streams, as follows, Table 11.

Value Stream Maps are often created by using agreed symbols, such as available from (Value Steam Mapping Symbols, 2010) for example. In addition to mapping the actual process from the customer request to the completed product, they visualise the usage of time. Time is divided for example to the value added time and the waiting time. These definitions are used e.g. by Poppendieck and Poppendieck (2003) in the context of lean software development. However, waiting in itself is one of the seven muda defined by the TPS (Ohno, 1988). Thus, a waiting time

Table 11: Value stream mapping steps (Abdulmalek & Rajgopal, 2007)

1	Choose a particular product or product family as the target for improvement
2	Draw a current state map of the process. This can be seen as a snapshot of how things are currently being done and are created by "walking along" the process. This provides the basis for analyzing the system and identifying its weaknesses.
3	Create a future state map. This is a picture that depicts what the system should look like when wastes have been removed. The creation of a future state map is done by answering a set of questions related to efficiency and from the technical implementation side, how to utilise lean tools in improvement activities.

Table 12: Value stream mapping tools (Hines & Rich, 1997)

	Mapping tool	Origin of mapping tool
1	Process activity mapping	Industrial engineering
2	Supply chain response matrix	Time compression/logistics
3	Production variety funnel	Operations management
4	Quality filter mapping	New tool (Hines and Rich 1997)
5	Demand amplification mapping	Systems dynamics
6	Decision point analysis	Efficient consumer response/logistics
7	Physical structure mapping	New tool (Hines and Rich 1997)

can also result from other muda, such as transportation, especially while producing physical products. In addition to the value stream map discussed above, Hines and Rich (1997) have defined seven different tools for mapping the value stream. The tools and their origins are depicted in table 12.

In addition to producing physical products, the VSM technique has also been applied in the software engineering context. For example, Poppendieck and Poppendieck (2003, 2007) discuss VSM in a software development context. The principle of the process is similar to the one applied in the production environment of physical goods; the process from a customer request to its completion is illustrated with the timelines for value adding and non-value adding activities and the wastes are identified. The process for eliminating waste is also outlined. The wastes of software development however vary from those identified in the physical goods production environment.

The benefits of lean

According to the many sources in literature (e.g. Middleton, 2001; Fujitsu, 2010), the lean approach has many benefits compared to the traditional approaches. Middleton (2001) summarises reasons in his article for the importance of lean thinking; lean thinking can reduce defect rates to 1 per million units, it at least doubles the productivity of both manufacturing and service operations, it reduces the time taken to deliver new products, whilst substantially reducing costs, and in general, lean techniques produce significant levels of improvement. Fujitsu reports similar benefits, they claim, for example, using lean has increased productivity by 30% (Fujitsu, 2010).

MIT researchers (2000) say that companies that have succeeded in transitioning to lean are those that take a holistic approach and view the transformation as a fundamental restructuring of the enterprise, including its organizational structure, business and information systems, workforce policies, incentive systems and relationships with customers and suppliers. Lean thinking should be applied to all functions in the enterprise.

MIT researchers (2000) continue that while there are no reported studies that quantify all the benefits of lean beyond production, the following general outcomes can be logically expected:

- The voice of the customer becomes the primary driving force in the enterprise. This has an impact on the product quality, organizational structure, production processes, policies and overall behaviour.
- New product development time is greatly reduced.
- Relationships with suppliers have been revolutionised.
- The responsiveness to changing market conditions is enhanced.
- The organizational structure shifts from a vertical to horizontal focus, aligning value-adding activities with the customer value stream.

- The workforce is empowered.
- Improved operating margins and increased flexibility provide enhanced business opportunities in existing and new markets.

Cusumano (1994) states that in a number of industries – however, mainly in the car industry - many benefits have been achieved as a result of applying lean principles: high productivity, high quality in engineering and manufacturing, and high price-performance in the value of products delivered to the customer. High levels of flexibility, by producing relatively small lots of different models, have also been achieved by Japanese firms. In the 1970s and 1980s, the Japanese car manufactures resulted in fast development times and a very aggressive expansion of product lines.

The costs of lean

MIT researchers (2000) highlight that the primary investment required is the time of the entire management team and workforce. Lean conversion does not typically require extensive capital investments, but lean requires a considerable investment in education and training.

Cusumano (1994) claims that, in spite of many benefits enabled by lean thinking, (especially in the car industry, but the same also applies to other manufacturing industries as well) some disadvantages have also appeared. Just-in-time deliveries have caused traffic congestion, which pollutes the environment and wastes time while people are jammed in traffic and in manufacturing plants waiting for components. Thus, Japanese companies even dispersed their plants to other parts of the world, the consequence of which was that reliable suppliers were not easy to find. And further, new problems arose: the shortage of factory labour was born. In the 1980s and 1990s, the product variety enlarged; Cusumano (1994) says that too much product variety and too many options were offered to customers. The large product variety requires better scheduling and control systems, and firms need to treat the root cause of the problem and reduce variety. A product variety that is too great has also created environmental concerns; consumers are encouraged to replace their vehicles very often.

Hines, et al. (2004) argue that over time, criticism has been presented in regards to lean thinking. As lean thinking evolved, the gaps in lean thinking have changed. Hines, et al. (2004) say that key aspects of this criticism are the lack of contingency, lack of consideration of human aspects, scope and lack of strategic perspective, and coping with variability.

Experiences of implementing lean principles in software development

Mehta, et al. (2008) argue that in addition to successfully applying lean principles in a variety of industries, the same principles can also be utilised for software development. They propose applying lean principles of providing the highest customer value, maximizing flow, and eliminating waste in the context of software development. In their research, Mehta, et al. (2008) disclose the following experiences of lean:

- **Organizing lean project teams.** Lean methods promote teams, where the project manager, or the value stream manager in lean terminology, maintains the flow or work. Functional managers are responsible for managing subprojects, the elimination of waste using the concept of value-added tasks and non-value-added tasks, and the reduction of common cause variation in the form of small errors.
- **Investing in team culture.** Lean also means a cultural movement for continuous improvement, utilizing the team efforts on a daily basis. The lean team culture promotes a failure tolerant environment, where employees are encouraged to try out new ideas and not penalised if their ideas fail.
- **Top management involvement.** In lean thinking, the firm and visual involvement of top managers is crucial.
- **The importance of initial planning.** The best practices of software engineering stress the importance of early and effective requirements gathering and specification, as well as architecture development and design. Software development engineering methods stress the value of this activity on multiple levels. (Drawing a parallel with manufacturing, designs in manufacturing consume around 20% of the total cost of a product, but locks in 70% of the total cost of the product.) One rule of thumb is to devote at least 5-15% of the total project time to upfront analysis and architectural work, before the writing of any code.
- **Working on new features versus fixing defects.** The application of lean is centred on the concept of “first time right” and “the best resource for the task at hand”. In lean thinking, resource allocation decisions are made more frequently regarding how each working activity provides value to the customer. This approach ensures customer satisfaction and enhances the flow of work. Thus, the choice of working on new features versus the fixing of defects shall be based on the ultimate customer value.
- **Improving the flow.** Although software projects are designed to produce one system, the requirements and functionality can be decomposed into components and then into units to which the notion of work flow can be applied. In lean, great attention is paid to the flow of work through the process. Improving the flow of work through the process is one of the greatest benefits obtained from applying lean techniques.
- **Reducing complexity.** The notion of reducing complexity has been promoted as best practices within software engineering industry for decades. This is also a lean concept. Best practices reduce the complexity to achieve higher quality and a smooth flow, hence also include using more frequent and smaller builds. Lean recommends the frequent evaluation of the priority of the work. Such frequent evaluations, reviews, and the reprioritization of small work packets, allows the team to deliver optimal value to the customer.

The paper of Parnell-Klabo (2006) describes how a large company applied lean principles in conjunction with the adaptation of agile techniques. The company executives outlined a speed to market goal calling for a significant decrease in product and project development timeframes. The pilot projects began in 2004. By 2005, almost 20% of the portfolio held projects leveraging lean and agile practices that enabled software delivery which were 30% to 50% quicker than projects of a similar size and complexity. Lean principles had been successfully applied to internal production operations, resulting in speed and efficiency gains at the company. Lean was recognised to be a proven method that not only showed improvements in speed, but also generated stronger quality outputs whilst lowering costs at the same time. Parnell-Klabo (2006) claims that the most challenging obstacles were obtaining facility space for collocations, gaining executive support, and influencing the change curve. In spite of these, impressive results were achieved. The duration of the project decreased by 40% compared to the waterfall baseline estimate. The actual resource costs were 10% under the allocated budget. No degradation of quality in the code was found. Additionally, an increase in morale was indicated.

Middleton (2001) claims that the concepts of lean manufacturing can be successfully transferred from the manufacture of cars and electrical goods to software development. He says that lean software development can produce rapid quality and productivity gains. Middleton (2001) argues that the vital concept for software development is that instead of producing large requirement documentation, it is to create it in smaller pieces and to pass these immediately to designers and developers. As soon as they find an error, it is returned back for correction. The paradigm shift required for software is to see that specifications and unfinished programs are stocks or work-in-processes. The real cost is in the errors that remain undiscovered in the documentation until a later date. Middleton (2001) highlights that adopting lean concepts in software development may require the entire organization to change, as the lean approach is holistic.

References

Aalto, J-M., 2008. Large scale agile development of S60 product software. Nokia. Available at: http://www.cs.tut.fi/tapahtumat/olio2008/esitykset/aalto.pdf [Accessed 14 May 2010].

Abdulmalek, F.A. & Rajgopal, J., 2007. Analyzing the benefits of lean manufacturing and value stream mapping via simulation: A process sector case study. International Journal of Production Economics, 107(1), pp. 223-236.

Abrahamsson, P., Salo, O., Ronkainen, J. & Warsta, J., 2002. Agile Software Development Methods: Review and Analysis. Espoo, VTT Electronics 408: 107.

Beck, K. & Andres, C., 2005. Extreme Programming Explained. Addison-Wesley Professional.

Bhasin, S. & Burcher, P., 2006. Lean viewed as a philosophy. Journal of Manufacturing Technology Management, 17(1), pp. 56-72.

Bozdogan, K., 2006. A Comparative Review of lean thinking, Six Sigma and Related Enterprise Process Improvement Initiatives. Working paper. Available at: http://lean.mit.edu/index.php?option=com_docman&Itemid=776. [Accessed 14 May 2010].

Cusumano, M.A., 1994. The Limits of lean. Sloan Management Review, Summer 1994, pp. 27-32.

Dybå, T. & Dinsoyr, T., 2008. Empirical studies of agile software development: A systematic review. Information and software technology, 50 , pp. 833-859.

Fujitsu, 2010. Lean creates a solid platform for growth. Available at: http://www.fujitsu.com/downloads/SVC/fs/casestudies/fujitsu-lean.pdf [Accessed 22 April 2010].

Hines, P. & Rich, N., 1997. The Seven Value Stream Mapping Tools. International Journal of Operations and Production Management (IJOPM), 17 (1), pp. 46-64.

Hines, P. et al., 1998. Value stream management. International Journal of Logistics Management, 9(1), pp. 25-42.

Hines, P., Rich, N., & Esain, A., 1999. Value Stream Mapping: a Distribution Industry Application. Journal of an International Benchmarking, 6(1), pp. 60-77.

Hines, P., Holweg, M. & Rich, N., 2004. Learning to evolve. A review of contemporary lean thinking. International Journal of Operations & Production Management, 24(10), pp. 994-1011.

Johnson, H.T., 1992. Relevance regained: From top-down control to bottom-up empowerment. Free Press.

Larman, G. & Vodde, B, 2009. Lean Primer, v.1.5. Available at: http://www.leanprimer.com [Accessed 14 May 2010].

LEI, 2010, Available at http://www.lean.org/whatslean/History.cfm. [Accessed 14 May 2010].

Liker, J.K., 2004. The Toyota way. 14 Management Principles from the World's Greatest Manufacturer. McGraw-Hill, New York, USA.

Mehta, M., Anderson, D. & Raffo, D., 2008. Providing Value to Customers in Software Development Through lean Principles. Software Process Improvement and Practice, 13, pp. 101-109.

Middleton, P., 2001. Lean Software Development: Two Case Studies. Software Quality Journal, 9, pp. 241-252.

Middleton, P. & Sutton, J., 2005. Lean Software Strategies. Proven Techniques for Managers and Developers. Productivity Press, New York, USA.

MIT, 2000, "Transitioning to a Lean Enterprise: A Guide for Leaders", Volume I, Volume II, Volume III. Available at: http://lean.mit.edu/index.php?option=com_docman&Itemid=776. [Accessed 14 May 2010].

Monden, Y., 1993. Toyota Production System: An Integrated Approach to Just-in-Time. 2nd ed., Industrial Engineering and Management Press, Norcross, GA, USA.

Nightingale, D., 2009. Principles of Enterprise Systems. In: Second International Symposium on Engineering Systems. MIT, Cambridge, Massachusetts.

Ohno, T., 1988. Toyota Production System: Beyond Large-Scale Production. Productivity Press.

Parnell-Klabo, E., 2006. Introducing Lean Principles with Agile Practices at a Fortune 500 Company. In: Agile 2006 Conference, Minneapolis 23-28 June 2006.

Perera, G.I.U.S. & Fernando, M.S.D., 2007. Enhanced Agile Software Development – Hybrid Paradigm with Lean Practice. In: Second International Conference on Industrial and Information Systems, ICIIS 2007, Sri Lanka 8-11 August 2007.

Poppendieck, M. & Poppendieck, T., 2003. Lean Software Development: An Agile Toolkit. Addison-Wesley, Upper Saddle River, NJ, USA.

Poppendieck, M & Poppendieck, T., 2007. Implementing Lean Software Development: From Concept to Cash. Addison-Wesley.

Russell, R.S. & Taylor, B.W., 1999. Operations Management. 2nd edn, Prentice Hall: Upper Saddle River, NJ.

Schwaber, K. & Beedle, M., 2002. Agile Software Development with Scrum. Upper Saddle River, NJ, Prentice Hall.

Senior, B. & Fleming, J., 2006. Organizational Change. 3rd edn, FT Prentice Hall, Pearson Education Ltd, Harlow.

Smeds, R., 1994. Managing Change towards Lean Enterprises. International Journal of Operations & Production Management, 14(3), pp. 66-82.

Spear, S. & Bowen, H.K. Decoding the DNA of the Toyota Production System. Harvard Business Review, September-October 1999.

Taylor, F.W., 1911. The Principles of Scientific Management. [e-book] Harper & Brothers, Eldritch Press. Available at: http://www.eldritchpress.org/fwt/ti.html. [Accessed 14 May 2010].

Tivit, 2009. Software - Strategic Reserach Agenda. Available at: http://www.tivit.fi/fi/dokumentit/22/Software-SRA+v10.pdf. [Accessed 22 April 2010].

Treacy, M. & Wiersema, F., 1993. Customer Intimacy and Other Value Disciplines. Harvard Business Review, January-February 1993, pp. 83-93.

Vaquero, L.M., Rodero-Merino, L., Caceres, J. & Lindner, M., 2009. A Break in the Clouds: Toward a Cloud Definition, ACM SIGCOMM Computer Communication Review, 39(1).

Value Steam Mapping Symbols, 2010. Value Stream Mapping Symbols. Available at: http://www.strategosinc.com/value-stream-mapping-3.htm. [Accessed 26th April 2010].

Womack, J.P., Jones, D.T., & Roos, D., 1990. The Machine That Changed the World, New York, Macmillan.

Womack, J. P. & Jones, D. T., 1996. Lean thinking: Banish Waste and Create Wealth in Your Corporation. Simon & Schuster.

About the authors

The main author of this white paper, Dr. **Raija Kuusela**, has almost 30 years experience of the ICT sector. She has been working as a senior research scientist at VTT since 2009. Prior to that, she worked for Nokia for 14 years in several roles including software development roles, Engineering Manager, Quality Manager, Scrum Master, and Product Owner. Before her career at Nokia, Kuusela worked with other software intensive companies for 13 years in software development roles. She received her PhD in Industrial Engineering and Management from the University of Oulu, Finland in 2007.

Dr. **Tua Huomo** has almost 20 years experience of the ICT sector. She is currently working as a research coordinator at VTT in the area of software technologies and business research. Additionally, she is working as a program coordinator in the cloud software program (www.cloudsoftwareprogram.org), which is a large collaborative research initiative in the field of cloud technologies, processes and business. Huomo has worked for VTT from 1997-1999 and from 2004 onwards. Prior to VTT, she worked as a Director of R&D at Cybelius Software and had various development and management assignments at the CCC Group. She received her PhD in information processing science from the University of Oulu, Finland in 2000 and an MBA from Oxford Brookes University, UK in 2009.

Mikko Korkala has ten years of experience of the ICT sector. He is currently working on his Ph.D thesis on customer communication in a software agile development. He has been working as a research scientist at VTT Technical Research Centre of Finland since 2007 and has previously worked at the Department of Information Processing Science, University of Oulu, Finland, from which he also received his M.Sc. In addition to research tasks, he has provided several agile trainings and workshops and has held invited agile talks both in Finland and abroad. He has also worked as an onsite agile consultant for the management in a large software company and has helped to outline agile processes for software companies.

Glossary

Autonomation	Automation with a human touch
ICT	Information and communication technology
JIT	Just-in-time (JIT) is an inventory strategy which strives to improve the return on investment of a business by reducing in-process inventory and associated carrying costs
Kaikaku	A lean production term, which in Japanese means a radical overhaul of an activity to eliminate all waste and create a greater value. Also called breakthrough kaizen.
Kaizen	Kaizen (Japanese for "improvement" or "change for the better") refers to a philosophy or practices that focus upon a continuous improvement of processes in manufacturing, engineering, supporting business processes, and management. It has been applied in healthcare, government, banking, and many other industries
Kanban	Kanban is a concept related to lean and just-in-time (JIT) production. Kanban is a signalling system to trigger action.
LAI	Lean advancement initiative
LEM	Lean enterprise model
MIT	The Massachusetts Institute of Technology
Muda	Muda is a traditional Japanese term for an activity which is wasteful and doesn't add value or is unproductive
Mura	Mura is a traditional general Japanese term for unevenness, inconsistency in physical matter or human spiritual condition
Muri	Muri is a Japanese term for overburden, unreasonableness or absurdity
NNVA	Necessary but non-value adding
NVA	Non-value adding
Scrum	Scrum is an iterative, incremental framework for project management and agile software development
SLA	Service level agreement. The SLA is a part of a service contract where the level of service is formally defined
SRA	Strategic research agenda
TPS	Toyota production system
TQM	Total quality management
UX	User eXperience
VA	Value-adding
VSM	Value stream mapping
XP	eXtreme programming

Lean transformation with VTT

Whether a company is starting a lean transformation program or simply needs some extra guidance, VTT's lean transformation service provides a systematic and structured current state analysis and makes recommendations on the next steps of the transformation.

The analysis encompasses:

- The impact of cloud/lean: opportunities and threats
- Strategy and structure
- Tools and methods
- Lean software development practices
- Continuous improvements

Results

- The current state analysis of lean transformation
- Key improvement areas, metrics and next steps

Value proposal

- **A wider perspective** - the opportunities and threats of cloud, lean business including strategy, systems, management, software development processes, people
- **Proven benefits** - the key areas for improvement are identified in order to receive early benefits
- **Value driven** – continually measure and evaluate the value being delivered

This VTT's research summary was completed in Tivit's (the Strategic Centre for Science, Technology and Innovation in the Field of ICT, (www.tivit.fi) Cloud Software Program (www.cloudsoftwareprogram.org). The work is partly funded by Tekes (the Finnish Funding Agency for Technology and Innovation, www.tekes.fi).

Making the Transformation to Cloud?

For more information please contact Dr. Tua Huomo, Program Coordinator, Cloud Software Program at +358 20 722 2451 or cloudsoftware@vtt.fi.

About VTT

VTT Technical Research Centre of Finland is a globally networked multi- technological contract research organization. VTT provides high-end technology solutions and innovation services in several research areas; one of those is ICT (Information and communication technologies). The business worldwide in the ICT sector is experiencing a cloud phenomenon.

VTT considers the capability of an organisation to benefit from cloud is dependent on its ability to adapt the entire organisation to the change. In the cloud, the production of amazing experiences is an imperative for business success.

VTT TECHNICAL RESEARCH CENTRE OF FINLAND
Vuorimiehentie 5, Espoo
P.O.Box 1000, FI-02044 VTT, Finland
Tel. +358 20 722 111
www.vtt. fi