# VALUE STREAM MAPPING OF A MOBILE NETWORK ELEMENT SOFTWARE - FEATURE DEVELOPMENT CYCLE USING LEAN DEVELOPMENT CONCEPT

Obe Olabode Sunday

Oulu University of Applied Sciences
2011

# VALUE STREAM MAPPING OF A MOBILE NETWORK ELEMENT SOFTWARE - FEATURE DEVELOPMENT CYCLE USING LEAN DEVELOPMENT CONCEPT

Olabode Sunday Obe

Bachelor's Thesis

Spring 2011

Degree Programme in Information Technology

Oulu University of Applied Sciences

# PREFACE

This work was done for Nokia Siemens Networks (NSN) during spring 2010. The topic of the thesis is the value stream mapping of a mobile network element software feature development and a consequence analysis of the development cycle, in order to improve in the future using lean development concept.

Ville Suomi was the company´s supervisor and Dr. Lauri Pirttiaho was the school´s supervisor of my thesis work. Jussi-Pekka Pudas, Sanna Sivonen, Carl Höglund, Matti Uusitalo, Hanna Koivula, Hannu Mikkola, Esa Trög, and Mika Narinen were my colleagues, who gave me valuable suggestions on how to solve the problems encountered in the course of this work.

I would like to thank Jarmo Nissi, Ville Suomi, Päivi Kuuva and Lauri Pirttiaho for their supervision and help during the work. I would also like to thank my family most especially my parents for supporting me financially, my brothers Dr. Olajide Obe, Adebayo Obe, Olorunfemi Obe and my one and only sister Olubunmi Obe for always been there for me when I needed them most.

My special thanks also go to Kaija Posio for language checking of this work. I would also like to appreciate Lea Hannila, Risto Korva, Janne Räihä, Sami Nahkiaisoja, and Juha Räty for their support during my studies in Raahe.

Oulu 31 October 2010

Olabode Sunday Obe

# TIIVISTELMÄ

Oulun Seudun Ammattikorkeakoulu

Tietotekniikan Koulutusohjelma

Tekijä: Obe Olabode Sunday

Opinnäytetyön nimi: Matkapuhelinverkon elementin uuden toiminnallisuuden toteutuksessa käytetyn ohjelmisto kehitysmenetelmän arvovirtakuvaus Lean-kehitysmenetelmän pohjalta.

Työn ohjaaja: Ph.D. Lauri Pirttiaho.

Työn valmistumislukukausi ja vuosi: Kevät 2011 Sivumäärä: 33

## TIIVISTELMÄ

Tämä opinnäytetyö on tehty Nokia Siemens Networksille. Tämän työn tarkoituksena on analysoida arvovirtakuvauksella (value stream mapping) ohjelmistokehityksen aikaisia toimia, joita käytettiin matkapuhelinverkkoelementin uuden ominaisuuden kehittämisen aikana. Kuvaus tehtiin mahdollisten hyödyttömään toimintaan (waste) johtaneiden syiden tunnistamiseksi ja ehdotusten tekemiseen ongelmien poistamiseksi tai ohjaamaan tukiaseman ohjelmistokehitystä Lean–periaatteiden mukaiseksi tulevaisuudessa.

Tämä työ lähtee kyseisen verkkoelementin tämän hetkisestä toimintamallista, missä useita ominaisuuksia kehitetään samoilla resursseilla vaiheittain samanaikaisesti. Kehittäjät ja johto ovat väittäneet, että yhden ominaisuuden kehittäminen kerrallaan alusta loppuun tuottaa asiakkaalle enemmän ja nopeammin arvoa. Näin ollen koko ohjelmistokehitysvaiheen kriittinen arviointi Lean–kehitysperiaatteiden mukaisesti on tarpeen.

Verkkoelementin ohjelmistokehitysprosessin arviointi toteutettiin tekemällä henkilökohtaisia haastatteluja sidosryhmille sekä hyödyntämällä kyselylomakkeita. Jokaiselle osaprojektille laskettiin tehokkuusluku, tehtiin arvovirtakaavion (value stream map) luonnos, osoitettiin waste–alueet ja tuotiin esille ehdotuksia mahdollisista tulevaisuuden kehityskohteista.

Asiasanat:

Arvovirtakuvaus, kevyt ohjelmistonkehitysprosessi, hukkatyö, ohjelmistokehitys, asiakas ja arvo.

# ABSTRACT

Oulu University of Applied Sciences

Degree Programme in Information Technology

Author: Obe Olabode Sunday

Title of Bachelor´s thesis: Value Stream Mapping of a mobile network element software - feature development cycle using Lean development concept.

Supervisor: Ph.D. Lauri Pirttiaho

Term and year of completion: Spring 2011 Number of pages: 33

## ABSTRACT

This thesis was done for Nokia Siemens Networks. The main objective of this work is the value stream mapping analysis of the software development activities that took place within the development cycle of a mobile network element feature. The mapping was done to identify the possible sources of wastes and suggest ways to eliminate the waste or streamline the work in accordance with lean principles, to improve the future mobile network element feature software development.

This work originated in the current mode of operation in the network element organization, where several features are developed concurrently with the same resources in a sequential manner. Developers and management have argued that developing one feature at a time from start to finish will deliver more value to the customers faster. Hence, critical evaluation of the overall mobile network element feature software development cycle is needed in accordance with Lean development principles.

The evaluation of the development process was carried out by conducting personal interviews with the relevant stakeholders, and also by using questionnaires. The efficiency value for each subproject was calculated, and a sketch of the value stream map was made, areas of waste were spotted and the possible future improvements proposals were stated.

Keywords:

Value stream mapping, Lean development, waste, software development, customer and value.

**TABLE OF CONTENTS**

# 1 INTRODUCTION

This bachelor's thesis deals with the value stream mapping of the software development cycle of a mobile network element feature and a consequence analysis of the development cycle for future improvements, according to Lean software development principles.

This work originated in the current mode of operation in network element organization within NSN, where several features are developed concurrently with the same resources in a sequential manner, and this has constantly being a source of challenge, in terms of how fast things are been done and how efficient the current process of development is. This has affected both the quality of the eventual software and also increases lead time.

## 1.1 Value Stream Mapping

A value stream is a series of end to end activities carried out within the software development cycle from when there is a customer´s demand to delivery. For example, documentations, specifications, meetings, coding, studying etc. It was first propounded by the Toyota production system (TPS) in the year 1948-1975 in a bid to make their manufacturing process leaner. The value stream map on the other hand, is a map that uses symbols and metrics to show the flow of activities in the value stream. This map is a graphical representation of all the activities that are carried out in the course of development. It makes it easier to spot value adding and non-value adding activities in the value stream. The term "value" is seen from the customer´s point of view as what they consider productive and are willing to pay for (Suomi 2010, discussion).

### 1.1.1 Software development within the NSN-Mobile Network Element

The software development process used at the time in the mobile network element organization was the traditional waterfall development methodology. The waterfall is a sequential model of development, whereby, the development process moves from phase to phase in a pre-defined falling order until the final delivery to the customer. For example, from

Requirement analyses -> specification->design-> implementation ->testing -> deployment to the customer (Software development process 2010, date of retrieval 1.6.2010).

The unique thing about this type of process is that each phase is completed before the start of the next phase and at the end, complete working software is deployed to the customer. The draw backs associated with this kind of process are enormous, for example, the inability to meet customer´s immediate changing requirements.

This and many more of the disadvantages of waterfall model are the major reasons behind this thesis work. Apart from the fact that the development method here is the waterfall model, the development of any mobile network element SW feature in the organization in question is carried out simultaneously at different locations across the world with seemingly unclear interface between the multisites. Other reasons behind this work are that; several other features are being developed at the same time, which have also raised questions from the team members and upper level management as to whether it is better to develop one feature at a time from start to finish with frequent releases to the customers or whether to continue with present state of development whereby two or more features are developed at the same time.

NSN is looking for a situation where it will be able to deliver on customer based requirements in just in time (JIT) manner in the future, i.e. deliver only what is needed by the customer in due time without wasting too much time on unnecessary or extra features. NSN is also envisaging a development model that will be used to accomplish the JIT delivery mode and also to overcome the disadvantages associated with the present approach.

The major goals of this project are to evaluate the activities carried out within the mobile network element feature development cycle, draw a value stream map for the overall development cycle, calculate the process cycle efficiency and state general improvement proposals for future development.

### 1.1.2 Mobile Network Element Software Feature

The target was to use a specific feature as a case study which covers as many subprojects and subcomponents as possible. The aim is that the proposals would also be applicable to as large number of future mobile network element features as possible (Suomi 9.2.2010, discussion).

# 2 WORK ENVIRONMENT

The development of this mobile network element software is structured around different subprojects and subsystems. Note that the following subsystems are specific to the feature in question, other features might include subsystems etc. that are not mentioned as part of this work. The subprojects for the feature being investigated are; product management (PM), entity feature specification (EFS), digital signal processing feature specification (DSP FS), system component testing (SCT), entity integration and entity verification.

The DSP FS is further subdivided into layer one, layer two and layer three subsystems based on their different levels of functionalities.

Layer one (L1) is further divided to two lower level subsystems which are L1 subsystem 1 and L1 subsystem 2. The L1 subsystem 1 comprises uplink and downlink subcomponents, while the L1 subsystem 2 comprises specification, receiver and HW API subcomponents.

Layer two (L2) subsystem involved is an uplink subsystem. L2 subsystem is comprised of eight different subcomponents, which are; L2 uplink labelled L2 subcomponents 1-8.

Layer three (L3) subsystem involved is a control subsystem of a telecom subsystem for this particular mobile network element feature. The L3 control subsystem is comprised of three subcomponents labelled L3 control subcomponents 1-3 respectively.

In addition, activities of the different subprojects vary based on their layers and levels of functionality. The product management (PM) is responsible for delivering market and mobile network element software product information from the customers to research and development (R&D), as well as advocates for the customer's needs and desires. They make development proposals to the upper level management and also analyse the monetary and human effort needed to develop BTS software. The Entity feature specification (EFS) is responsible for getting the customer requirements of the mobile

network element SW features from the product management based on customer demand. They analyse and specify the features for development based on 3GPP standards. Digital signal processing (DSP) feature specification team is responsible for specifying the feature for layer one (L1), layer two (L2) and layer three (L3) functionality, while L1 subsystem 1 is responsible for actions as specified for DSP. Rake team is responsible for implementing features that use services provided by layer one. While L2 uplink subsystem with its subcomponents is responsible for implementing layer two functionalities. Others are SCT (System Component Testing); whose primary responsibility is to test the integrated layer one and layer two components, while L3 control subsystem SCT is responsible for testing the integrated L3 control subcomponents.

In conclusion, the major responsibilities of the entity integration (EI) and entity verification (EV) teams are; integration planning, integration testing, verification planning, verification testing, and support.

(Nokia Siemens Networks mobile network element SW development team 2010, interview).

# 3 DESCRIPTION OF THE RESEARCH

The theoretical foundation of this value stream mapping development task is based on the principles of Lean development in software engineering, quality and process as propounded by Mary Poppendieck and Tom Poppendieck (2002). Lean software development is a concept that is primarily aimed at eliminating waste, and delivering quality to the customer in software development process. It focuses on value to the customer, less work and greater overall process cycle efficiency.

Activities in lean software development are categorised into three different types:

- Activities or actions that do not add value from the customer´s perspective.
- Activities or actions that do not add real value but are necessary to complete the process.
- Activities or actions which create value from the customer´s perspective.

According to the principles of Lean, activities which do not create value from the customer´s perspective should be eliminated or reduced in order to streamline the value stream. It always starts with a clear definition of what the customer see as a value and what they see as waste. When both terms are defined, then it is easier to spot the waste from the value stream. The whole idea of this concept is to optimize the value stream by making the development process leaner and more efficient.

## 3.1 Method

The research was carried out by conducting personal interviews with the relevant stakeholders and also by using questionnaires. The research questionnaire consists of a series of questions pertaining to the stakeholder's involvement in the development, from when it started in April 2007 till December 2009 when there was the first customer release. The maintenance phase that was carried out in the year 2010 by L2 uplink subsystem and DSP SCT was also considered. Below is the template of the questionnaire used in the interview process:

- Brief introduction and your role in the development of mobile network element feature.
- When did you start working?
- When did you finish working?
- How many value added hours did you put into the development of this feature?
- What are the major challenges you encountered in the course of the development?
- What are the improvement proposals you would like to suggest?
- Who else is involved in the next phase of the development after you are done?
- Anything else or any other comments?

A total of thirty-four people were interviewed, One person was interviewed from the product management (PM), one person from the entity feature specification (EFS), two people from the digital signal processing (DSP), four people from the L1 subsystem 1 subproject, three people from the L1 subsystem 2 subproject, six people from the L3 control subsystem subproject, eight people from the L2 uplink subsystem, four people from the DSP SCT, one person from the L3 control subsystem SCT, one person from the entity integration (EI), and one person from the entity verification (EV).

In addition to the number of people interviewed, the head of Lean and agile development in NSN, the subproject manager of quality and process within the mobile network element organization, and four other software developers in the tools and automation department were also personally interviewed. The interviews took the following forms; face to face in the meeting room, through teleconferencing, over a phone, by email and as group interviews as well as casual discussions with colleagues. All the findings from the interviews were published on the value stream mapping Wiki page of NSN.

# 4 RESULTS

The development of this feature started in April and ended in December when there was the first customer release but the maintenance is still ongoing.

Based on the total number of people that were interviewed, the Table 1 below shows the breakdown of all the activities that took place in the course of the development in the mobile network element subprojects and subcomponents, the number of people involved in each phase, the value added time spent in each phase, the total cycle time and the efficiency of each phase as well as the overall efficiency and the time duration of each phase of the development.

From the Table, it can be seen that no number was recorded for the value added time (VAT), total cycle time (TT) and efficiency for the product management subproject. This is due to the fact that the product management personnel cannot specifically state how much time was used. Similarly, the digital signal processing (DSP) personnel stated that the time they used for the specification is very negligible and therefore inconsequential. All other entries represent the actual overview of the development cycle with a very little margin for error.

*TABLE 1. The summary table of the overall mobile network element feature SW development cycle*

The table below shows a comprehensive summary of the activities in all the subprojects; the number of designers involved, value added time, total cycle time, efficiency and the duration of the development from start to finish of each subproject.

| NAME OF SUBPROJECT | ACTIVITIES | NUMBER OF DESIGNER | VALUE ADDED TIME (HRS) | TOTAL TIME(hrs) | EFFICIENCY (%) | DURATION |
|---|---|---|---|---|---|---|
| **PRODUCT MANAGEMENT** | Business case analysis, effort estimation, and pipeline analysis | 2 | N/A | N/A | N/A | 25.1.2005 – N/A |
| **ENTITY FEATURE SPECIFICATION** | Specification & Maintenance | 1 | 503,00 | 4800,00 | 10,47 | 2.5.2007 – 31.12.09 |
| **DIGITAL SIGNAL PROCESSING** | N/A | 2 | N/A | N/A | N/A | N/A |
| **L1 SUBSYSTEM 1** | | | | | 44,88 | 13.4.07- 31.12.2009 |
| L1 subsystem 1 downlink subcomponent | Specification, Implementation, Implementation, module testing and bug fixing related effort | 2 | 616,00 | 945,00 | 65,19 | |

| | | | | | | |
|---|---|---|---|---|---|---|
| L1 Subsystem 1<br>Uplink subcomponent | Implementation, module testing & Implementation, module testing | 2 | 1496,00 | 3760,50 | 31,61 | |
| **L1 SUBSYSTEM 2** | | | | | 13,78 | 13.6.2007-31.12.2009 |
| L1 subsystem 2 Specification | L1 subsystem 2 Specification, Implementation, Testing & Maintenance | 1 | 465,00 | 1800,00 | 12,40 | |
| L1 subsystem 2 Receiver | Implementation, Integration & Maintenance | 1 | 485,00 | 3450,00 | 14,06 | |
| L1 subsystem 2 HW API | Implementation, Integration & Maintenance | 1 | 125,00 | 600,00 | 20,83 | |
| **L3 CONTROL SUBSYSTEM** | | | | | 8,60 | 2.5.2007-31.5.2009 |
| L3 control subsystem subcomponent 1 | Implementation & Maintenance | 2 | 200,00 | 3817,50 | 5,24 | |
| L3 control subsystem subcomponent | Implementation, Integration & Maintenance | 2 | 80,00 | 210,00 | 38,10 | |
| L3 subsystem subcomponent 3 | Implementation & Maintenance | 2 | 80,00 | 150,00 | 53,33 | |
| **L2 UPLINK SUBSYSTEM** | | | | | 34.11 | 1.12.2007-18.3.2010 |
| Subcomponent 1 | Specification, Meetings, Supports | 1 | 740,00 | 2107,50 | 35,11 | |

| | | | | | | |
|---|---|---|---|---|---|---|
| Subcomponent 2 | Application Development | Managing, Supports and Planning | 1 | 305,00 | 3000,00 | 10,16 |
| Subcomponent 3 | Design Manager Control | Design, Implementation, Module Testing & Maintenance | 1 | 550,00 | 3150,00 | 17,46 |
| Subcomponent 4 | Design Resource Management | Design, Implementation, Module Testing & Maintenance | 1 | 1330,00 | 3150,00 | 42,22 |
| Subcomponent 5 | Design Scheduler | Design, Implementation, Module Testing & Maintenance | 1 | 1090,00 | 3150,00 | 34,60 |
| Subcomponent 6<br>Subcomponent 7 | Design<br>Mac_e | Design, Implementation, Module Testing & Maintenance | 1 | 360,00 | 3150,00 | 11,42 |
| Subcomponent 8 | Design<br>Sumo | Design, Implementation, Module Testing & Maintenance | 1 | 470,00 | 3150,00 | 14,92 |
| Target Testing | Target Testing | Testing | 1 | 3600,00 | 3900,00 | 92,30 |
| **DSP SYSTEM COMPONENT TESTING(SCT)** | **USER PLANE SYSTEM COMPONENT TESTING(SCT)** | | | | | 63,70 |
| DSP New Feature Testing | User Plane & New Feature Testing | Test Specification, Verification, Trainings & Supports | 4 | 2795,00 | 4387,50 | 63,70 |

| | | | | | | |
|---|---|---|---|---|---|---|
| **L3 CONTROL SUBSYSTEM SCT** | Tests specification, trainings | 1 | 251,00 | 750,00 | 33,47 | 1.8.2008-20.12.2008 |
| **ENTITY INTEGRATION(EI)** | Integration of System components | 1 | 448,00 | 1380,00 | 32,46 | 3.11.2008-3.7.2009 |
| **ENTITY VERIFICATION(EV)** | | 2 | 924,00 | 2085,00 | 44,36 | 13.3.2008-30.4.2009 |
| **TOTAL** | | | 16914 | 54843 | 30,84 | |

*FORMULA 1. Efficiency is calculated by the value added time (VAT) divided by the total cycle time (TT) multiplied by 100.*

*Efficiency = VAT / TT * 100 in percentage.*

The value added time is the time spent on value added work during the course of the development. However, It should be noted that the times used for the calculations are received from the T activity tracking tool of the people working of the people working in NSN.

The total cycle time is the sum of all cycle times in the value stream. The calculation is based on the official 7,5hours per day and 20 days per month work, including all paid holidays and overtime.

The efficiency values simply give us an idea of what portion of our activities are value-adding ones and also the amount of wastes in the course of the development of this particular feature.

*Using EFS as a case study;*

*Recall, efficiency = VAT / TT * 100;*

*VAT = 503hrs.*

*TT = 4800hrs.*

*Efficiency = (503 / 4800) * 100 = 10.47%*

*It is also called a process cycle efficiency.*

*The efficiency value of 10.47% means that out of the total hours of 4800 which covers a period of 2years 7months and 30days, only 10.47% of the time was used on the value added activities and 89.53% on some kind of wastes from the feature point of view.*

In a similar way, the efficiency values of other subcomponents or phases can be analysed in order to see the amount of time used for both value and non-value added activities. Only a thorough analysis of the end-to-end value stream will reveal the actual activities that constitute wastes.

### 4.1 Value Stream Map

The figure 1. Below is the value stream map of the software development cycle of the mobile network element feature.

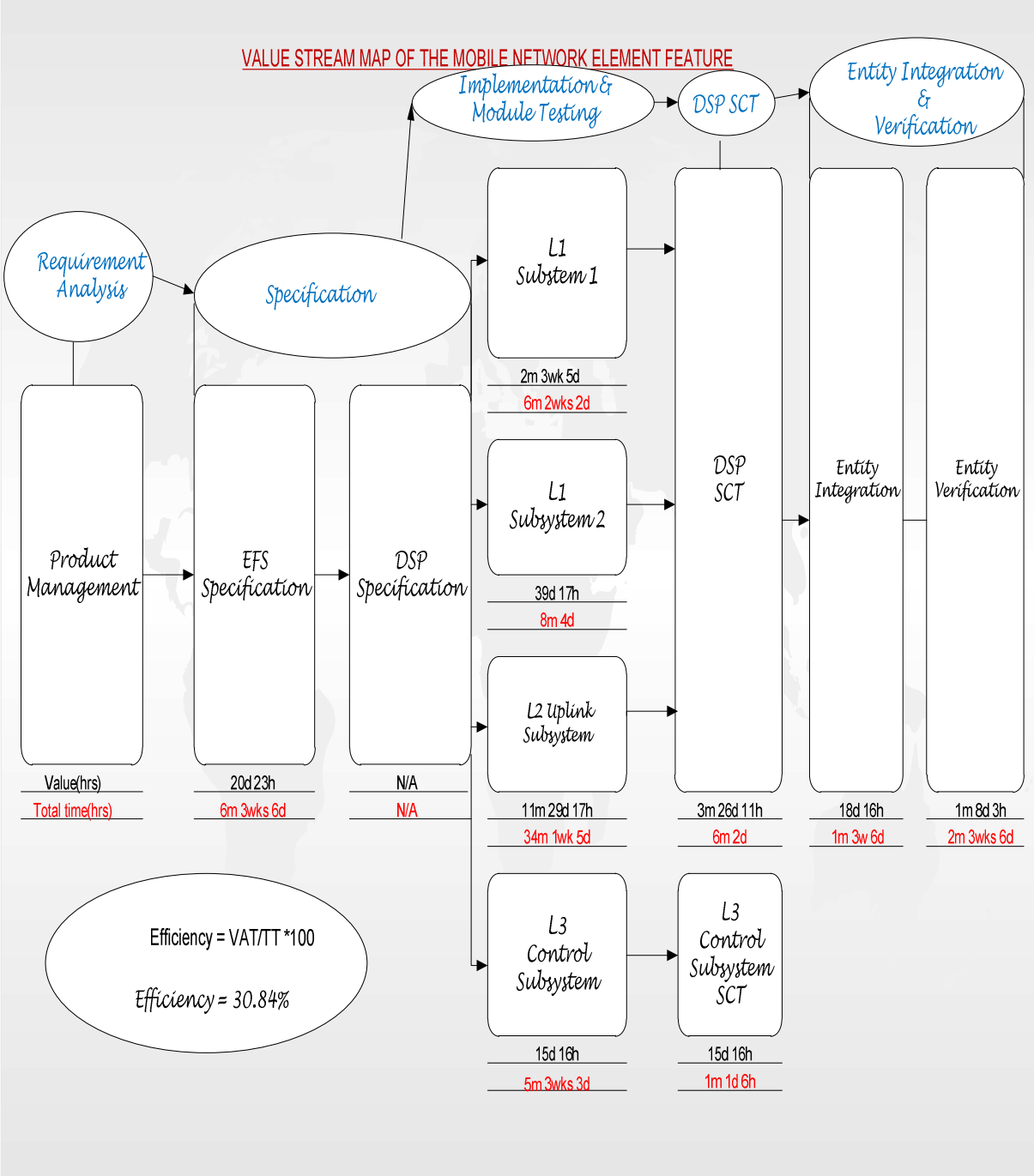


*FIGURE 1. The value stream map of the mobile network element SW feature development cycle*

Where

*M* is months,

*D* is days,

*Wks* is weeks

*H* is hours.

The diagram above is the simplified value stream map of the whole development cycle based on the data collected. The values shown on the map are both the value added time (VAT) and the total cycle time (TT). Also, the subprojects involved in the development of this particular mobile network element feature are represented on the diagram. The map was sketched in this form so as to show the sequential flow of activities right from the product management, where customer´s demand start, to the entity verification when the working software is ready to be delivered.

It should however be noted that the various subprojects have different subcomponents which are not shown on the diagram but their times were also calculated.

For example, L1 subsystem 1 Subproject has two subcomponents:

- L1 subsystem 1 Uplink and
- L1 subsystem 1 Downlink.

The L1 subsystem 1 uplink subcomponent has both a L1 subsystem 1 implementation and a module testing. Likewise the codec downlink subcomponent also has an implementation and a module testing.

The value added time is received by adding the individual VAT from each subcomponent that is;

*VAT (L1 subsystem 1) = VAT (L1 subsystem 1 uplink) + VAT (L1 subsystem 1 downlink)*

*VAT (L1 subsystem 1) = (1496 + 616) hrs*

*VAT (L1 subsystem 1) = 2112hrs*

*Similarly, the total cycle time is also received by adding the individual TT from each subcomponent as shown below:*

*TT (L1 subsystem 1) = TT (L1 subsystem 1 uplink) + TT (L1 subsystem 1 downlink)*

*TT (L1 subsystem 1) = (945 + 3760.50) hrs*

*TT (L1 subsystem 1) = 4705.50hrs*

*Recall;*

*Efficiency = VAT / TT * 100;*

*Efficiency = (2112 / 4705.50) * 100*

*Efficiency = 44. 88%*

As explained earlier, the efficiency value simply implies that 44.80% of the total cycle time was used for value added activities and the rest for some other activities. In a similar way, the subcomponents from other subprojects can be seen from the table 1 above.

Also, the overall efficiency value was calculated in a much similar manner as explained above but in this case the VAT is the sum of VATs from all the subprojects and the TT is also the sum of all the TTs from all the subprojects. The overall efficiency value of 30.84% means that only this percentage out of the total time was used for value added activities and 69.16% of the time was used for something else.

### 4.1.1 Analyses

Based on the result from the value stream mapping analysis, all the development teams agreed that the common sources of wastes are; bureaucracy which occurs for example, as a result of too many changing company policies. It takes too much time or process to effect a simple change, etc. Others are;

The pressure from the top level management to do things on time, multitasking and the general communication issues. Specifically for this feature, one of the major pressures from the upper level management was to get the peak data rate from the beginning of the development mainly for advertorial purpose which is not a proper thing to do at the early stage of the development work. The effect of this demand was a crash in the eventual software, which later requires extra task times to fix.

On the average, all the people that were interviewed were also involved in at least two other development works simultaneously. And as we know, multitasking tends to reduce speed and decrease efficiency in thought-intensive work.

The communication issue has also been a major source of waste in the development e.g. the communication between different designers in different parts of the world, the communication between the product management and R&D, the communication between the specification personnel and the development teams, These communication lapses occur not because of inadequate means of communication but because of the inability of the personnel involved to choose the appropriate channel or means of communication.

In addition to the common sources of wastes described above, the improper estimation of R&D pipeline capability and not clearly prioritized features causes lots of delays and several postponements in the product management side of the development. The others are unclear requirements, unclear specifications and documentations, both at the implementation and module testing phase and at the testing phase of this development.

I classified the remaining sources of wastes broadly into three different categories i.e. the management, the frontline development teams and the tools as discussed below:

The major sources of wastes from the upper level management were their inability to provide the requisite training for the DSP SCT team in two locations before starting the testing, which eventually affected other phases of the development, and caused delays in setting up test environment places. On the other hand, the wastes associated with the development teams are as follows:

- Too long iteration in one release caused a huge amount of extra and unproductive work in the target testing side of the L2 uplink subsystem.
- Too many parallel and unneeded software iterations due to lack of adequate and clear planning of needs and deliverables at the mobile network element SW level.
- Too many unproductive meetings.
- Too much time spent on competence transfer, i.e., transferring of skills and responsibilities of older employees to the newly employed ones.
- The process of learning how software is being developed in NSN is always a challenge for new employees.
- Lack of L2 uplink subsystem architects.

And finally, tools. The testing tools were not made available on time and the ones available were very complicated tools.

### 4.1.2 The List of Improvement Proposals

Looking from the previously discussed sources of wastes, I would recommend that NSN acquired Lean expertise to see whether the problems with bureaucracy, pressure from the upper management, unproductive meetings, communication, etc, can be removed. The first step to achieve this or make the development process Leaner is a complete change of mindset towards our approach to software development and consciously applying the Lean principles.

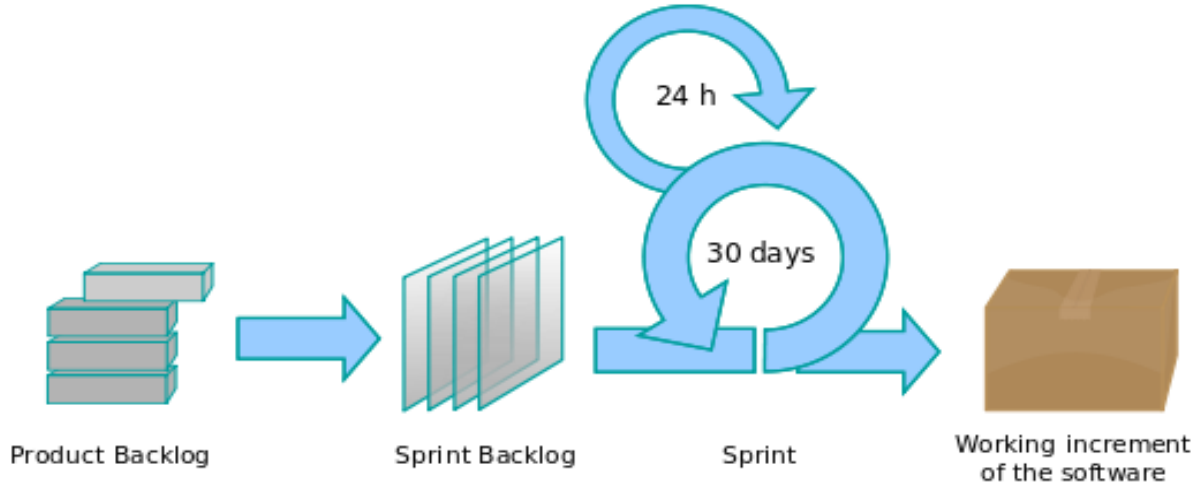


*FIGURE 2. Agile Development (Effective agile development 2009, date of retrieval 2.6.2010).*

This concept also employs the use of agile practices such as sprint planning, 10-15 minutes daily scrum meetings, short iteration so that potentially shippable features available as often as possible, increased amount of feedbacks between customers and development teams. When the amount of feedback is increased, the waste will be easier to spot and there will be less problems associated with integration unlike in the waterfall model.

In order to apply this concept correctly, I would also like to propose that three sets of groups should be involved in the process, They are the upper level management, the frontline development teams and the paying customers. The responsibilities of the upper level management based on Lean principles are as follows:

- Provide the means of educating the developers about the major Lean principles.
- Train them to be able to identify the process waste as well as perform in multifunctional teams and
- Give the developers the autonomy to make value creating decisions because they are responsible for delivering the value to the customer.

The above mentioned can be achieved by organising seminars, trainings, workshops etc.

Ideally, the development teams should more or less serve as the interface between the company and the customer if NSN really intends to make the process Leaner. This is not the case here at the moment, but as the teams are still responsible for delivering value to the customers, then R&D will work closely with the product management in order to receive regular feedback from the customers about their needs, and of course, to develop software based on the need of the customers.

The customer is responsible for outlining the functionality needed from the software and also to give regular feedback when their requirements change. In that way the development teams will know what the measure of the value is from their perspective. And also, the link between the customer, product management and developers should be flexible and dynamic through regular communication so that development teams receive customer feedback in due time. It is realized though, that active customer involvement is somewhat paradoxically challenging to arrange.

The other and more specific proposals are listed below:

- Specification and requirement analysis should be done on incremental or iterative basis. It simply implies that no upfront specification of the whole system architecture is required. It should be done in smaller batches to develop a workable piece of software, to deliver to the customers frequently and also to meet customer's changing needs. The length of iteration will vary for different projects but shorter iteration times are generally preferred. Likewise, design should also be done iteratively because software development is not easily predictable. It is better to design little and implement little, with integrity and resilience built into the system. This will allow modifications of the system with a reasonable amount of work.

- Multitasking should be reduced if it cannot be totally eliminated by developing one feature at a time from start to finish.
- Remove the communication barriers entirely by not spreading the work over a number of sites and also have clear interface between different sites.
- Clear planning at as early phase as possible. Priority should be given to the features with the highest business value, and customer needs should be clearly defined from the beginning of the development. The development sequence should be carried out from the most valuable features from start to finish before proceeding to the less valuable ones with the same resources.
- Bureaucracy should be reduced. It cannot be totally eliminated because of the size of the company. One way of achieving this, is to transfer the responsibility for the customer satisfaction to the frontline teams, empower the teams to be able to make decisions and also to be able to use customer feedback to drive change, instead of the traditional managerial problem solving methods.
- There should be adequate training for developers and testers at the beginning of any development.
- Reduce unproductive meetings as much as possible.
  Using the law of two feet;" it states that if you're neither contributing nor getting value where you are, use your two feet and go somewhere where you can.
  In other words,
  Whoever comes to the meeting are the right people.
  Whatever happens is the only thing that could have.
  Whenever it starts is the right time.
  When it is over it is over". (Open space technique 1980, date of retrieval 2.6.2010).
- Reduce the pressure from the management; the developers should be given a high degree of autonomy on how to achieve the goals set by the management. We should be completely inflexible in where we want to go but completely flexible on how to get there.
- Set up test environment and provide easy to use testing tools well in advance before the testing starts.
- Proper estimation of R&D pipeline capability should be carried out before the development starts. "Agile does not prescribe a single way for effort estimations; however, it postulated that teams should use size of work to quantify effort instead of the traditional hours based estimations. Common estimating methods include numeric sizing (1 through 10), t-shirt sizes

(XS, S, M, L, XL, XXL, and XXXL), and the Fibonacci sequence (1, 2, 3, 5, 8, 13, 21, 34, etc.). The most important thing is that the team shares an understanding of the scale it uses, so that every member of the team is comfortable with the scale's values". (Scrum Efforts Estimation and Story Points 2009, date of retrieval 2.6.2010).

- Working completely from outside the premises of NSN should be completely discouraged unless there will be adequate provision of working tools, and access to NSN internal network resources.
- Increase knowledge and capabilities to deliver according to customer´s demand.
- More pragmatic planning and upfront analysis of wastes. The first and major step in Lean development is learning to see the wastes. If something does not add value directly to the customer, it is waste. For example, part of the induction of Lean development in Toyota production system (TPS) was spent learning how to identify wastes. It is also important to note that not all wastes should be eliminated; some are necessary wastes that should be streamlined.

# 5 CONCLUSIONS

The evaluation of the mobile network element functionality development cycle was carried out in the form of interviews with the relevant stakeholders involved in the development of this feature. After the interviews,the findings were documented and the value stream map was drawn. The next step was the phase by phase analysis of the entire mobile network element software development cycle of this feature from the documented findings.The areas of wastes were identified in each phase, efficiency of each phase was calculated and improvement proposals were stated accordingly.

From the result of the evaluation of the SW development cycle of the mobile network element feature, it is quite obvious that there is dire need of better ways of doing things in order to increase efficiency and achieve software development productivity with the given resources. Lean development is not a quick fix concept; it will take a lot of effort to work effectively in NSN considering the nature and size of the company. It has to start with a change of orientation, management philosophy, mindsets, and thinking.

First and foremost, there is a need to actually understand in clear terms what the customer sees as value and how the feature is being used by the customer. It will involve making the customers a key or integral part of the software development process and probably finding a way to receive first-hand information on how the customers use the software to create value for themselves.

Also, the software development methodology has to be a contrast to the one used here, unlike the traditional methodology where people are assigned with specific job roles. In Lean agile software development,people will have their traditional primary roles based on their competencies but they will also have their secondary roles to be able to work effectively in cross-functional and self organized teams.

More so,the hierarchy at the management levels will also be affected, e.g. the line managers that have mainly been involved in administrative tasks will be given specific job roles in the development teams.

In addition, the number of sites we have around the world where development is carried out simultaneously could also prove to be a major source of challenge but I suggest we try to limit the amount of site dependency involved in the development of a particular feature and also to have a clear interface between these sites.

On a final note, Implementing Lean successfully requires a clear understanding of Lean principles and a methodical step-by-step approach. Developers must be educated about the major Lean theories and principles as well as trained to perform ad hoc problem solving, identify process waste and defects and collaborate in multifunctional teams (Semonchtchak, V. & Cook, J 2004, date of retrieval 3.6.2010).

# 6 DISCUSSIONS

It can be seen from the results of the efficiency that there is so much waste in the software development process and there is a need for better ways of doing things in order to increase efficiency and optimize the value stream.

From the perspective of both L3 control subsystem and DSP, this particular mobile network element feature is very small and as such it is not ideal for the value stream mapping analysis.

Also, a couple of other factors should be considered when using the result of this research to get a complete picture or balanced view of how the analysis was carried out and the challenges I faced when trying to analyse the value stream. Some of the challenges are explained below.

The hours recorded in the activity tracking tool are not very reliable because of a number of key factors, for example, if the task lists are not updated by managers then it is hard for team members to register their working hours. There is also the possibility that team members are working on the development of other features simultaneously which makes it difficult to receive the actual time they used for this particular feature especially if the times for the different features are recorded together. In essence, the times used for the calculations are only used to give the hectares and cannot be totally relied on to be 100% accurate. More so, the value stream map was supposed to show a detailed breakdown of, for example, how much time was spent on meetings, travelling, and reading documentations, etc. But because the stakeholders cannot explicitly say how much time they used on specific tasks, they gave a lump sum of the times they used for the whole activities they were supposed to be doing. Some people even gave rough estimates because they were also involved in the development of other features at the same time and the times were recorded together.

In addition, the value stream analysis was carried out based on the information gathered from the stakeholders in the course of the interview partly because software development is knowledge work and the frontline people have the firsthand information on the issues or challenges they faced during the course of development and also because there are no existing documents where we can get the detailed breakdown or the basics of the activities that were carried out.

In conclusion, I would like to propose that in order to make the value stream mapping analysis more accurate in the future; the stakeholders should be well informed about the concept so that they will be better prepared to attend to the interview.

# 7 LIST OF REFERENCES


1. Effective agile development, 2009. Date of retrieval 2.6.2010

***http://effectiveagiledev.com/Portals/0/800px-Scrum_process_svg.png***

2. Open space technique, 1980. Date of retrieval 2.6.2010,

http://www.directservicetribes.org/docs/06162010/Open%20Space.pdf

3. Poppendieck, M. & Poppendieck, T. 2002. Principles of Lean Thinking. Date of retrieval 31.3.2010

http://www.poppendieck.com/papers/LeanThinking.pdf.

4. Scrum Efforts Estimation and Story Points, 2009. Date of retrieval 2.6.2010,

http://scrummethodology.com/scrum-effort-estimation-and-story-points/

5. Semonchtchak, V. & Cook, J. 2004. Lean object oriented software development.

Date of retrieval 3.6.2010, http://www.allbusiness.com/management/166694-1.html