Ismo Paukamainen

Product Virtualization in Large Scale Development

Adapting System Test for a Virtualized Product

Helsinki Metropolia University of Applied Sciences

Master's Degree

Information Technology

Master's Thesis

22 April 2015

Helsinki Metropolia University of Applied Sciences

Preface

Virtualization represents the biggest change I have experienced during my 25-year career in the field of testing. It has, and will have, more significant impacts on testing compared to the impacts when moving from a product to another, or when moving from the waterfall model to the Agile methodology.

For me in my early fifties, making this study was a good way to upgrade my competence with the network functions virtualization and related information. Writing this study helped me in the learning process, and in keeping the pace.

I am grateful to all subject matter experts and stakeholders involved in the evaluation of the outcome of my study and gave professional comments about my study. Furthermore, I extend my thanks to my instructors in my company Mr Johan Kvarnström and Mr Tomas Nordman, and at Metropolia University of Applied Sciences Mr Ville Jääskeläinen and Ms Zinaida Grabovskaia.

I could say that my studies at Metropolia University of Applied Sciences are a demonstration that learning is a lifetime journey.

Ismo Paukamainen

Author(s) Title Number of Pages Date	Ismo Paukamainen *Product Virtualization in Large Scale Development: Adapting System Test for a Virtualized Product* 108 pages + 5 appendices 20 April 2015
Degree	Master's Degree
Degree Programme	Information Technology
Instructors	Ville Jääskeläinen, Head of Degree Program Zinaida Grabovskaia, PhL, Senior Lecturer Tomas Nordman, B.Sc. Section Manager, Johan Kvarnström, B.Sc. Product Owner

Today, the telecom operators' networks are populated with a large variety of proprietary hardware appliances. In the near future, by using standard IT virtualization technology, many of the network equipment types will be replaced by standard servers, switches and storage. Network Functions Virtualization (NFV) is rapidly emerging in telecom networks.

Virtualization will bring many changes in the software development. Testing as part of software development will also be influenced by these changes. This thesis analyzes these impacts on testing, concentrating on the system test area in particular.

The case company of this project is a leading ICT company that is also offering virtualized products for telecom markets. Many of its products have already been virtualized or this virtualization work is currently going on. This study was conducted at the Media Plane Development organization of the case company.

BGF (Border Gateway Function) is a product of the case company's Media Plane development organization, which is a logical node in the MRS (Media Resource System). The virtualization of BGF is done, as part of MRS. Virtualization of BGF means that instead of offering solutions, as a combination of proprietary software with a proprietary hardware platform only, also a combination of proprietary software with the industry standard hardware and software components need to be offered. These combinations have impacts on system testing.

Thus, this study addressed the impacts brought up by virtualization in the system test area for the case organization. The analysis was twofold. First, the study analyzed the current context of the system test, which means analyzing the current system test strategy, test environment and the ways of working. The second step was to analyze how the network functions virtualization requirements impact on the current context. Based on these analyses the study suggests adaptations for the current system test strategy, test environment and ways of working. Finally it also presents an action plan for deploying the changes, and recommended improvements in system test strategy.

Keywords	Network Functions Virtualization, System Test, Cloud

Table of Contents

Appendices

List of Figures

List of Tables

Abbreviations and Acronyms

3GPP	3rd Generation Partnership Project (The 3rd Generation Partnership Project (3GPP) unites [Six] telecommunications standard development organizations (ARIB, ATIS, CCSA, ETSI, TTA, TTC), known as “Organizational Partners” and provides their members with a stable environment to produce the Reports and Specifications that define 3GPP technologies.)
AS	Application Server
AAA	Authentication, Authorization, and Accounting
ACM	Association for Computing Machinery
ANSI	American National Standards Institute
API	Application Programming Interface (a set of routines, protocols, and tools for building software applications)
API	Application Programming Interface
ATDD	Acceptance Test Driven Development
ATM	Asynchronous Transfer Mode
BDD	Behavior Driven Development
BGF	Border Gateway Function
BGFv	Virtualized Border Gateway Function
BSS	Business Support System
BUCI	Ericsson Business Unit Cloud and IP
CEE	Ericsson Cloud Execution Environment
CI	Continuous Integration
COTS	Commercial of the Shelf
CPU	Central Processing Unit
CWTS	Chinese Wireless Telecommunication Standard
DevOps	Development and Operations. A software development method that emphasizes communication, collaboration information sharing, between software developers and operations professionals.
DSL	Digital Subscriber Line
DSP	Digital Signal Processor
EBS	Ericsson Blade System
ECS	Ericsson Cloud System
EM	Element Manager

EMS	Element Management System
ETSI	European Technical Standards Institute
FaaS	Failure-as-a-Service
GCP	Gateway Control Protocol
HD Video	High Density Video
HD Voice	High Density Voice
HSPA	High Speed Packet data Access (HSPA) (3GPP HSPA (n.d.)
IaaS	Infrastructure-as-a-Service
ICT	Information and communications technology
IEEE	Institute of Electrical and Electronics Engineers
IM-MGW	IP Multimedia Gateway
IMS	IMS (IP Multimedia Subsystem) is a core network solution built on 3GPP standards, enabling real-time consumer and enterprise communication services over any access technology (HSPA, LTE, Wi-Fi, Fixed). Using one core network platform, operators can offer converged mobile and fixed services over any devices. Examples of services are HD voice, voice over LTE (VoLTE), video communication, HD video conferencing, Wi-Fi calling, IP based messaging, WebRTC, and other new innovative multimedia communication services. (Ericsson (n.d.a))
IP	Internet Protocol
ISG	Industry Specification Group
ISP	In Service Performance
KVM	Kernel Based Virtual Machine
L4	The Transport Layer, Open System Inter Connection (OSI), Layer 4 (ISO/IEC 7498-1 (1994:28))
L7	The Application Layer, Open System Inter Connection (OSI), Layer 7(ISO/IEC 7498-1 (1994:28))
LTE	LTE (Long Term Evolution) or the E-UTRAN (Evolved Universal Terrestrial Access Network), introduced in 3GPP R8, is the access part of the Evolved Packet System (EPS). The main requirements for the new access network are high spectral efficiency, high peak data rates, short round trip time as well as flexibility in frequency and bandwidth (3GPP LTE (n.d.)).
MANO	Management and Orchestration
M&O	Management and Orchestration

MGC	Media Gateway Controller
MMTel AS	Multi Media Telephony Application Server
MRF	Media Resource Function
MRFP	Multimedia Resource Function Processor is a media plane node in MRS used to mix, source or process media streams for voice and video conferencing, multimedia message playing and media conversion services.
MRS	Media Resource System, which provides the converged media plane functionality in IMS networks.
MSC-S	Mobile Switching Center Server, MSC Server
MSC	Mobile Switching Center
MSS	Mobile Soft Switch
MTAS	Multimedia Telephony Application Server
N-PoP	Network Point of Presence
NFV	Network Functions Virtualization
NFV-MANO	NFV Management and Orchestration
NFVO	NFV Orchestration
NIC	Network Interface Controller
O&M	Operations and Maintenance
OS	Operating System
OSS	Operations Support System
OVF	Open Virtualization Format
PL	Payload
PNF	Physical Network Function
PRA	Preliminary Availability. The product is ready for release with limited availability.
PSTN	Public Switched Telephone Network
QoE	Quality of Experience
QoS	Quality of Service
RX	Reception
SBG	Session Border Gateway
SC	System Controller
SDN	Software Defined Network

SGC	Session Gateway Controller
SGCv	Virtualized Session Gateway Controller
SGW	Signaling Gateway
SIP	Session Initiation Protocol
SLA	Service Level Agreement
SUT	System Under Test
TTC	Telecom Technology Committee, TTC is an incorporated association that contributes to standardization activities in the field of information and communication technology (ICT) by developing and disseminating standards for information and communications networks.(TTC(n.d.))
TTCN-3	Testing and Test Control Notation version 3
TX	Transmission
VIM	Virtualized Infrastructure Manager
VM	Virtual Machine
VNF	Virtual Network Function
VNFM	Virtual Network Function Management
VoLTE	Voice over LTE
WebRTC	Web Real Time Communication
Wi-Fi	Commercial term for WLAN (wireless local area network), an abbreviation for Wireless Fidelity

1 Introduction

Network Functions Virtualization (NFV) is a trend in today's telecom business. It has impacted and will impact on the development of the telecom systems in many ways. This study focuses on exploring the impact of virtualization on system testing in the context of one department of the case company.

1.1 Case Company Background

The case company of this project is Ericsson. It is a leading ICT company that offers products for telecom markets. Many of the Ericsson products have already been virtualized or the virtualization is going on. For Ericsson products virtualization means that it should offer, in addition to solutions based only on Ericsson proprietary software with the proprietary hardware platform, also a combination of proprietary software with the industry standard hardware and software components.

One example of product virtualization is the IMS network. The architecture of the cloud platforms based IMS network is illustrated in the figure below (Figure 1):

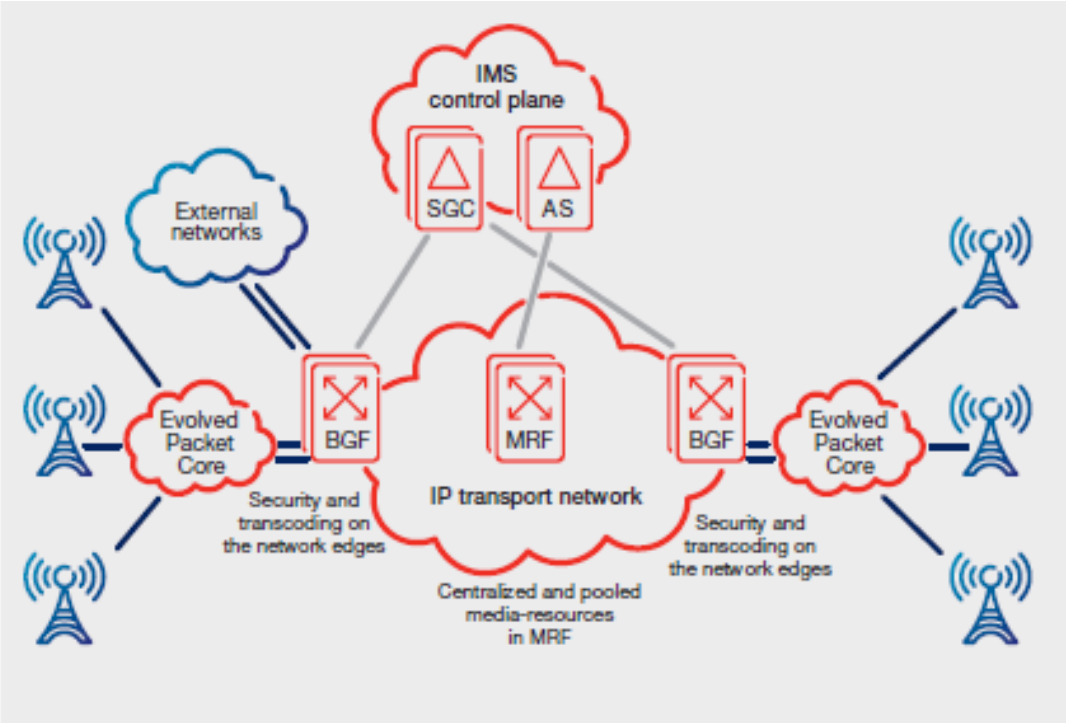

Figure 1. Architecture of an IMS network. Reprinted from Lundström J. 2013: 6.

Figure 1 above shows the network architecture that is build optimizing latency and ensuring bandwidth efficiency.

Ericsson IMS (IP Multimedia Subsystem) is core network solutions build on 3GPP standards. It is meant for real-time communication services over mobile and fixed access technologies such as HSPA, LTE, Wi-Fi, and DSL. Examples of these communication services are HD (High Definition) voice, voice over LTE (VoLTE), video communication, HD video conferencing, IP based messaging, Wi-Fi calling and WebRTC. (Ericsson n.d.a)

The converged media plane functionality in IMS networks are provided by Media Resource System (MRS). Media Resource System contains the media plane functions for IMS networks listed in the table below (Table 1):

Table 1. Media plane functions in MRS, Data gathered from Ericsson (n.d.b).

	Media Plane Function
1	The Border Gateway Function (BGF) provides the security and policy control for the media plane between IMS core network and access network
2	The Multimedia Resource Function Processor (MRFP) provides media services in IMS networks such as announcements, audio and video conferencing and media processing
3	The IP Multimedia Gateway (IM-MGW) is a connectivity layer function for IMS - PSTN network interconnection

Table 1 above lists the Media Plane functions in MRS for IMS networks. MRS can be deployed as combined node or with any combination of these logical nodes (Ericsson n.d.b).

Border Gateway Function (BGF) is a logical node in the Media Resource System. BGF is a product of the case company's Media Plane development organization. This study, therefore discusses how the BGF node is virtualized, as part of MRS, for the cloud platforms based IMS.

1.2 Network Functions Virtualization

Network Functions Virtualization is actively gaining ground in telecom industry. It will leverage modern technologies such as those developed for cloud computing, like for example, hardware virtualization by means of hypervisors, as well as, the usage of virtual Ethernet switches for connecting traffic between virtual machines and physical interfaces. The hardware used for this are industry standard high volume servers (e.g. using x86 architecture) and components, such as network Interface controllers (NIC). The software components are, for example, Intel Data Plane Development Kit, open API's for management and data plane control, such as OpenStack or OpenFlow.

Network Functions Virtualization aims for a clear separation between functional logic defined in software and the underlying infrastructure, offering an opportunity to redesign the way network functions are implemented. Instead of being implemented in vertically integrated boxes (often called "physical appliances,") network functions will be provided as virtual appliances, in other words, software is executed in a virtualized infrastructure environment. This is shown in the figure below (Figure 2):

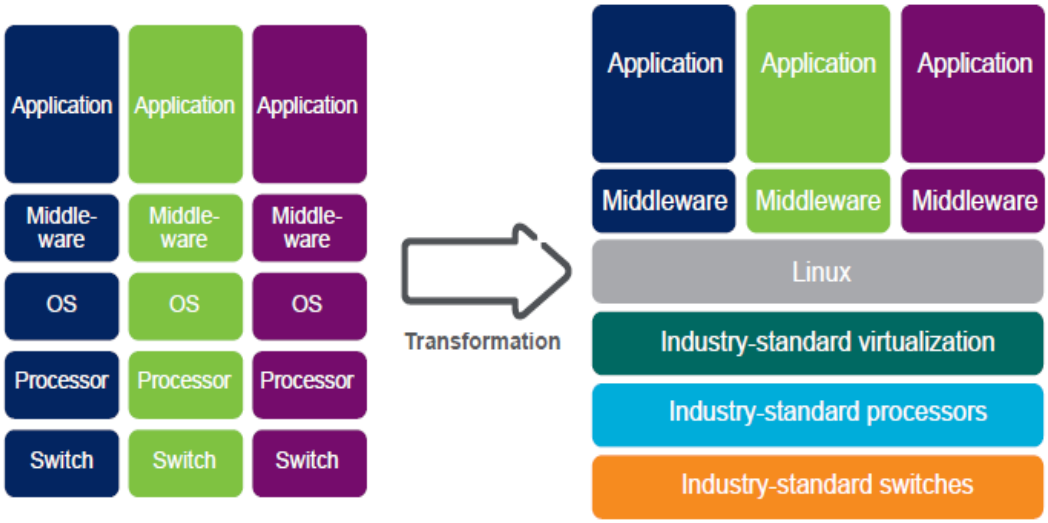

Figure 2. Decoupling applications from infrastructure. Reprinted from Ericsson 2014: 2.

Figure 2 above illustrates how Network Functions Virtualization (NFV) decouples applications, for example, Virtual Network functions (VNF), from the infrastructure. (Ericsson 2014: 2)

1.3 System Testing and Test Strategy

A term *System Testing* is defined in the following way: *"In system testing the behavior of whole system/product is tested as defined by the scope of the development project or product. System testing should investigate both functional and non-functional requirements of the testing."* (ISTQB n.d.) A *functional requirement* is a requirement that specifies new functionality, whereas a *non-functional requirement* is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors (ISO/IEC/IEEE 29119 n.d. a:29).

In the context of this study, a term *Test Strategy*, wherever used in the text, means the organizational test strategy as described in ISO/IEEE 29119 Software Testing standard (ISO/IEC/IEEE 29119 n.d. b:2,5,8,9,11,12). The organizational test strategy defines, for example, how the test scope, is divided between different activities in the organization, who is responsible of doing what. It may contain information about, guidelines or directives for working, for example, what documents are required to produce, when to make defect reports, or test reports. It may include also a strategy for test tools, defining the main tools that should be used in testing.

The organizational test strategy is often divided in two parts, one concerning the near future (typically around e.g. 1-3 years ahead) and the other part concerning the future with a longer perspective (typically for e.g. 5 years ahead). Since the organizational test strategy also serves for communication goals and ways of working, it means, it needs to be well documented and available, especially in the multi-site organization. Therefore it is important to make sure that everyone in the organization knows where to find it, what it contains.

1.4 An Oncoming Change

In the case company, like in many other telecom vendors, the virtualization means a huge change for the whole product development, including testing. The biggest impacts on testing will be in the test strategy, test environment in the targeted hardware, test tools and the ways of working. Therefore starting the virtualization will require starting with defining a new strategy and approach for testing. Today the Media Plane organization has been developing Border Gateway Function (BGF) product already for years, and its specified test strategy has been working well. Therefore, the current test strategy can be used as a starting point for the new strategy. However, to be able to set up a new strategy and approach for testing, all the impacts on testing from virtualization should be analyzed in advance. In practice this analysis and anticipation should cover the test environment, test tools, the ways of working and human aspects, such as, new skills and competencies of testers.

1.5 Research Question, Scope and Structure of the Study

The objective of this study was to develop such a system testing strategy that address the change related to the forthcoming virtualization of BGF product in product development. To put precisely, this study answered the following research question:

How to adapt the current system testing for the virtualized border gateway function (BGFv)?

This included addressing the following sub-questions:

1. To clarify what are the requirements for the virtualization
2. To clarify how the virtualized border gateway function (BGFv) operates.
3. To clarify how system testing need to be changed for this new virtualized environment
4. To adapt the present system test strategy for the new virtualized border gateway function (BGF)
5. To evaluate the proposed changes in strategy validated by the key experts/ stakeholders in Media Plane Development.

This study includes analysis of the Network Functions Virtualization requirements and material related to it, and how they impact system testing in the case company's Media Plane Development organization. Based on this analysis, proposed changes in system test strategy, triggered by the impacts are presented. The thesis was concluded by the evaluation of the proposed changes in the system test strategy, which is done by the expertise of the experts.

This report is written in ten sections. Section 2 describes the methods and the material used for this study. Section 3 describes the most important factors in the current context of the system test that help the reader to understand the needs of the change. Such important factors are, for example, ways of working, overall test strategy of the product, the importance of the feedback loops and the test environments. Section 4 describes the network functions virtualization, the requirements it has, and the network functions virtualization environment in general. Based on Sections 3 and 4, Section 5 and 6 present the impacts of the Network Functions Virtualization on the case company's system testing, and also propose changes in system test strategy for adapting these impacts. Section 7 presents a detailed action plan for the changes proposed in the previous sections. Section 8 evaluates the proposal validating it by the key experts/stakeholders in Media Plane Development organization. Based on the evaluation it also presents the upgraded plan for deployment. Section 9 introduces opportunities for improvements, and finally Section 10 concludes the study by summarizing and evaluating it.

2 Method and Material

This section discusses the research approach, research design and methods used in this study. It gives an overview about the data and data collection methods and analysis used.

2.1 Research Approach

To achieve the research goals and to contribute to solving the research problem, this study was conducted using an exploratory case study approach. This approach was selected as the most suitable for addressing the research question and the objective discussed in the Introduction.

According to Baxter P. and Jack S. (2008) the quantitative case study methodology should be considered when the focus of the study is to answer 'how' and 'why' questions and the study is aiming to cover contextual conditions. In this study the analysis of the current system test formed the case that needed to be studied. Then Baxter P. and Jack S. (2008) also instruct that after deciding to use a case study approach *the case of analysis need to be determined*. That is, to decide what is the unit of analysis, in a bounded context. In this study it was the system test in Media Plane Development organization. After determining what the case is, it needs to be considered what will not be included in the case. In order to avoid the problem having too many objectives in the study Baxter P. and Jack S. (2008) suggested that *placing boundaries* on the case that can prevent the study from losing its focus. This would mean binding a case to time and place, time and activity, and by definition the case and its context. In this study, the case thus meant analyzing the current system test in Media Plane Development organization, with the focus on the current (time) system test (activity and context) in Media Plane Development organization (place and context).

According to Baxter P. and Jack S. (2008), once the qualitative case study is selected as an approach for the research, the case and its boundaries have been determined, *the type of the case* study need to be considered. The case study may be categorized as explanatory, exploratory, or descriptive. In this study, to analyze the current system test in the Media Plane Development organization was exploratory in its type.

2.2 Research Design and Process

The research designed of this study includes the following steps. First, *the literature review* was conducted for identifying theoretical knowledge about the network functions virtualization. Second, the study analyzed *the current situation* in the case company's Media Plane Development organization, which made the case of this study. Third, the solution was suggested based on the above two. Finally, the solution was validated with the key stakeholders and case company experts for gathering feedback and further improvement suggestions. The research process is presented in the figure below (Figure 3):

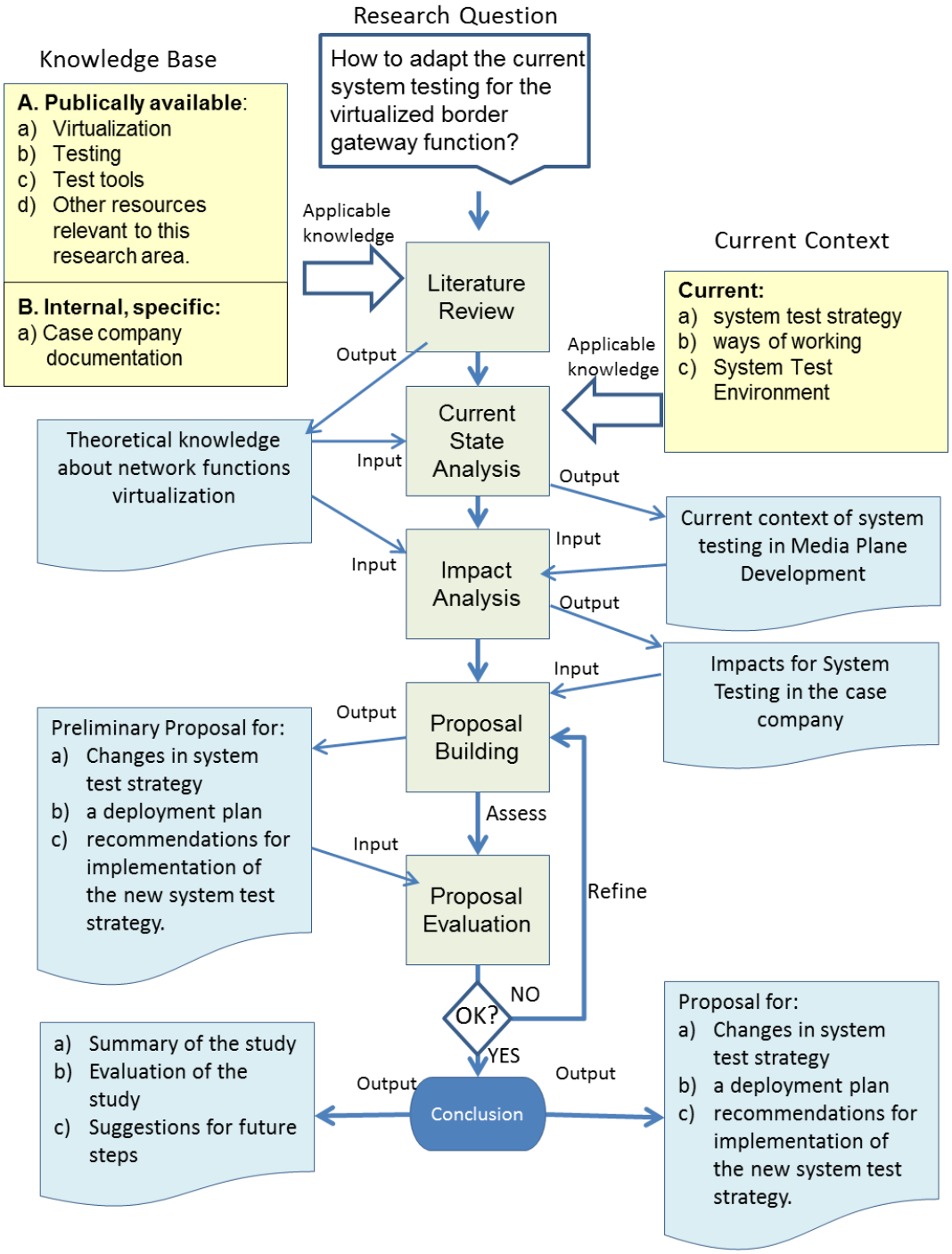

Figure 3. Research process of this study.

Figure 3 above shows the research process with all the stages. These stages are detailed in the flowing sections.

Literature Review

The study started with a literature review on virtualization and its impacts. Literature review was done by identifying, evaluating and interpreting the existing knowledge relevant to this study. It was based on the analysis of publically available scientific articles (including publicly available Ericsson articles), books and other resources relevant to this study area.

To the search for the existing knowledge and best practice, the relevant literature database sources were used, such as ACM digital library, IEEE Xplore® Digital Library and ETSI Technologies & Clusters, but not limiting the search within the above mentioned databases only. Various sources, articles, journals, white papers were found. Most of the sources were found by using key words including, for example, 'network functions virtualization', 'NFV', 'telecom cloud'.

Additionally, general information about the virtualization of telecom systems was found from ETSI (European Technical Standards Institute). There is a network operator-led Industry Specification Group (ISG) with open membership working through the technical challenges for Network Functions Virtualization. ETSI NFV SG has published several documents related to Network Functions Virtualization (ETSI 2014-10-22 :3, 4). These ETSI documents give a comprehensive view on what kind of expectations and requirements do network operators have.

In addition to publically available material also the case company's internal information about the implementation of the virtualized Border Gateway Function (BGFv) and the virtualized Media Resource System MRS needed to be analyzed to see how the implementations impact the system testing.

It should be noted that it was challenging to find information outside the company related to working methods or strategies related to test and virtualization. With this area being fairly topical at the moment, that kind of information is seen as business sensitive and thus not public. Therefore, the information for the analysis from other case company organizations was considered highly valuable and could not be abundant when it came to impacts on testing.

The Current State Analysis

Second, the study analyzed *the current situation* in the case company's Media Plane Development organization, which made the case of this study. The analysis concentrated on how the virtualization impacts the system test and it was based both on the literature findings and the observations, interviews, and case company document scrutiny as the case study.

The current state analysis was based on the data listed in the table below (Table 2.) collected and analyzed in the case company:

Table 2. Details of data collection.

	Type of data	Content	Input	
1	Internal documents	- Verification Strategy for M-MGw - Continuous Integration wiki pages	22 pages Wiki	Internal
2	Internal Wikis	Release Area wikis	~10 pages	Internal
3	Discussions	About ways of working Product Owner Johan K., Section Manager Tomas N., Senior Developer Timo K.,	~1 hour (per each)	Field notes
4	System Verification Way of working	Discussion about the setup for feature integration and non-functional system tests.	1 hour	Minutes of the meeting
4	Business Unit Cloud and IP Test Strategy Team	Workshops Discussing, planning, agreeing common test strategy for the whole business unit. (same participants as in the team below)	Two workshops 1-2 days per each.	Minutes of the meeting
5	Business Unit Cloud and IP Test Strategy Team	Team meetings and other team communication (phone calls, e-mailing) Team members from several development organizations, from several product areas, from different countries.	Every Monday 1,5 hours. Started September 2014.	Minutes of the meeting
7	Participant observation	By the researcher, in the organization	Continuous	Field notes, project reports

As seen from the table above, the current state analysis focused on the current situation in the Media Plane Development organization. It was analyzed based on the documentation, interviews and workshops. This included meetings, discussions and e-mailing with people from other Ericsson organizations that were in the same situation, or who had already got further in virtualization.

Impact Analysis

Third, based on the input collected in the literature review and from the current state analysis, the impacts on system testing in Media Plane Development were analyzed. This happened by analyzing how feasible the current system testing was for new virtualization requirements. In practice this meant analyzing needs for changes in system test strategy, such as, changes in the test scope, changes test environment or ways of working.

Proposal Building and Evaluation

Fourth, based on the impact analysis, the study proposed *changes in system test strategy*, *an action plan for implementation* of the new system test strategy, and *recommended improvements*. Fifth, as a method of evaluation of the proposed model, together with the result of the impact analysis, the study *validated* the proposal in the validation sessions with key stakeholders and experts in the case organization. At the same time, also the results of the impact analysis were evaluated. The evaluation was based on the interviews and workshops (validation sessions) as listed in the table below (Table 3.):

Table 3. Details of validation data collection.

	Type of validation data	Content	Input	
1	Meetings, workshops	Product Owner Johan K. and Section Manager Tomas N. Walkthrough of the proposed changes.	Every Friday 1 hour. January – April 2015.	Memos

2	Evaluation of the proposed changes	Experienced Developer Marko S. Evaluation of the area.	RA Media Quality	e-mails
		Senior Developer Joona V. Evaluation of the area.	RA Vulnerability	e-mails
		Master Developer Rabbe T. Evaluation of the area.	RA Characteristics & ISP	e-mails
		Product Owner Mikko P. Evaluation of the area.	RA O&M, RA Upgrade and Expansions	e-mails
		Product Owner Johan K. Evaluation of the area.	RA Robustness, RA Stability RA Signaling RA Single Traffic & Features	e-mails
		Test Environment Manager Antti A. Evaluation of the area.	Test Environment	e-mails
3	Overall evaluation of the study	Section Manager Juha K. and Section Manager Tomas N. Overall Evaluation	General view	e-mails
		Head Product Owner Tatu K. and Strategic Product Manager Johan L. Overall Evaluation	Product view	e-mails

3 The Current Context for System Test of the Native BGF

This section describes the current context of the system test for the native Border Gateway Function (BGF) product on a level that gives a general overview and helps to understand the needs for changes when the product is virtualized. Also two ‘internal factors’ in the current development in the Media Plane Development organization are presented. First, fitting system testing in Agile context by using a release area concept. Second, using the one-track development, that is an enabler for continuous deliveries.

3.1 The Current Product Level Test Strategy

Presently, the Media Plane Development is using a test strategy that covers all software testing activities in product development. This means that, also system test strategy is part of it. There are two types of system tests, functional and non-functional system tests. The first is covered by *end-to-en feature integration tests*, and the latter is covered by the release area concept. Release Area Concept is presented later in Section 3.1.3 *Release Area Concept*. And Feature integration tests are presented in Section 3.1.4 *Feature Integration*. Originally the strategy was created 2009 when the organization moved from incremental software development to using the Agile software development methodology. The strategy has evolved during the years based on the feedback from its usage and it has been continuously improved.

As said, the current test strategy covers all testing in the software development in Media Plane Development organization. That is, all test levels starting from the unit level up to network level. This kind of overall test strategy is crucial in a communication point of view. It describes the test scope on each test activity/test level, thus everyone knows what is tested and verified and where, on which activity. Overall strategy helps to plan testing so that nothing is omitted, but also in a way that there is no unnecessary and costly overlapping in testing.

3.1.1 The Current Test Scope Division

The existing test strategy corresponds to the agile methods. The Media Plane Development adaptation of the Agile Testing Quadrants is shown in the figure

below (Figure 4). It is based on the Agile Testing Quadrants presented by Lisa Crispin and Janet Gregory (Gregory J. and Crispin L. 2009: 98).

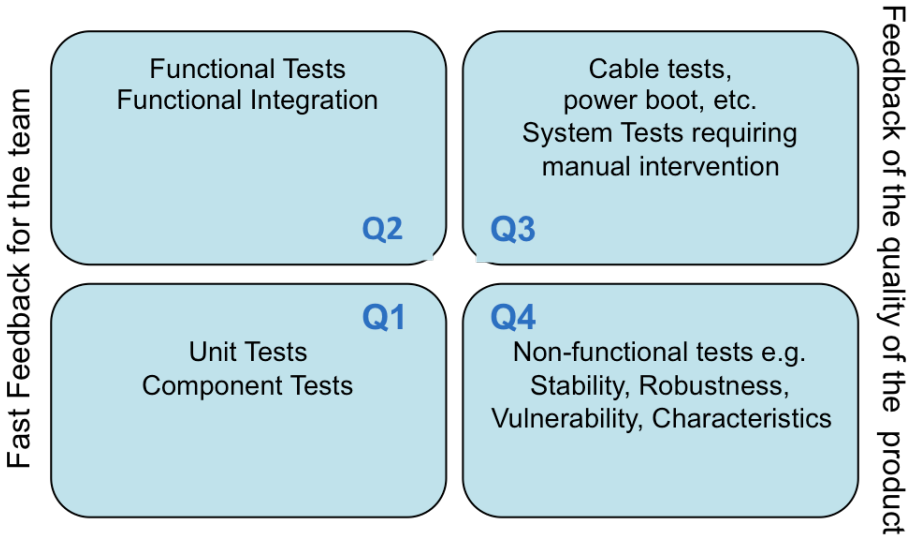

Figure 4. Agile Testing quadrants adaptation in Media Plane development.

Figure 4 above illustrates a model of Agile Testing Quadrants used in Media Plane Development. This model defines on a high-level all the tests that are needed. The model helps to communicate the test needs to everybody. The tests in Media Plane Development organization are divided in the ways illustrated in the table below (Table 4.):

Table 4. Division of test scope in Media Plane Development.

	Test Quadrants
1	The aim is that Cross functional teams would cover all the quadrants, but in practice they mainly cover Q1 and Q2 including functional tests and short load test in production environment
2	Continuous integration machinery covers mainly functional tests Q2 and small amount of system tests in production-like environment Q4
3	The independent test teams cover the non-functional tests in production-like environment Q3 & Q4.

The division of the testing quadrants between the teams is shown in Table 4 above. After these tests in the Media Plane Development there is currently IMS network Integration and Verification activity that performs, as the name says,

network level tests for the whole IMS release. As part of the IMS network also BGF releases are delivered to these tests. The team setup and activities are discussed in more details in the following sections.

3.1.2 The Current Feedback Loops in Development

Fast feedback loops is one of the basic idea in the strategy. That is, the time measured from implementing a new program code to getting a feedback about the quality should be, as short as possible. The feedback tells if the newly implemented code worked, as it should, and if it caused any problems in the previous legacy functionality. To be able to get fast feedback means in many cases that testing is reasonable to perform in the target, production-like, test environment. To get fast feedback, tests need to be implemented and executed on the integration level, which provides shorter preparation and execution time. Not forgetting that test code implementation requires time and resources, and creates always also maintenance costs. In general test automation is one way to shorten the feedback loop, especially when it comes to legacy part of the system.

When working in Agile context the development is typically done in a small increments, e.g. in two week's sprints as it is currently done in the case organization. That is, why it is very important that the developers get the feedback of their work as fast as possible to be able to make corrections also as fast as possible. There is a concept for fast feedback in the Media Plane development organization. It is called 4F-concept. It stands for "Fail-Fast-Fix-Fast". The figure below (Figure 5) shows the approximate feedback times of the current testing on different integration levels in Media Plane Development.

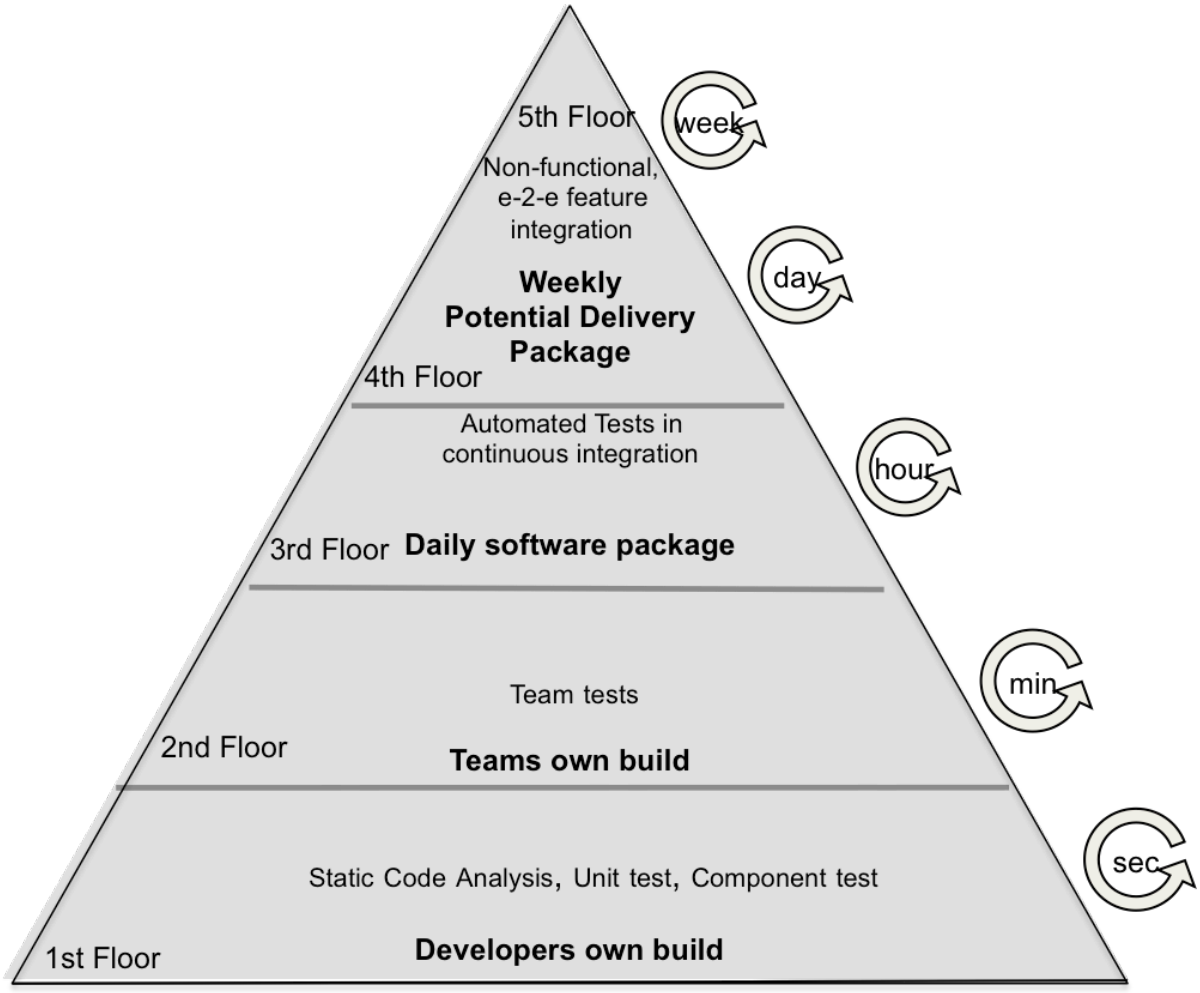

Figure 5. Feedback times in Media Plane Development.

Figure 5 above shows the feedback times, from seconds to week. To give further details the table below (Table 5.) shows different levels and feedback times in the current development:

Table 5. The levels and the feedback times.

Floor	Feedback Times per Activity
1	Work is done on various local software versions. Developers test their own code. The feedback of the quality is possible in very short time, even in seconds.
2	Cross Functional Teams may have their own team versions, but they are supposed to commit to main-track continuously after 'team tests and static analyses'. In team tests the feedback times is in minutes. Tests, which are done on teams own build before committing on the main-track must provide feedback

	in minutes. These tests may include e.g. static code analysis or quick ‘smoke test’ covering the legacy.
3	Automated tests are run (‘washing machines’) four times a day on the main-track for all committed code. These tests are run as part of the continuous integration machinery. The latest run of the day will be a new daily common software package. On this level the feedback comes in hours.
4	When there is a new common software package is available, automated system level tests are then run on it (once per day). Also cross-functional teams perform testing of new functionality on common software package. Once per week one of the software packages is selected to be a new weekly potential delivery package. Delivery Manager does the selection together with Product owners based on the input from the teams. Feedback times are still in hours.
5	Independent test teams may start testing already on daily common software package. They continue system level testing on the weekly potential delivery package. Feedback time on this level is from day to week.

Table 5 above explains the levels in the pyramid figure (Figure 5.) starting from the first floor of the pyramid. These feedback times are also used when selecting the test environment (i.e. when selecting the integration level) for tests. When selecting the environment, it needs to be considered what is sufficient environment for covering the test scope, and also capable to provide the fastest feedback, containing also the time used for preparations and implementing possible test program code.

3.1.3 Release Area Concept

Before Media Plane Development organization moved to agile, the system test was a separate test phase. It was a very late activity with a long lead-time. When moving to agile, it was clear that this kind of phase did not fit in the picture anymore. On the other hand, the fact is that a big number of faults, found in the production at the customer’s sites, are in the system legacy such as in stability or robustness, not in the new features. The reason behind this is that the new features are very well in focus when they are developed, whereas the system legacy as an area is so wide that it could not be covered fully.

Although the system test phase was not feasible anymore, the system legacy needed to be secured somehow. The former system test phase needed to replace by something else, which was more feasible for agile ways of working. The solution was that Media Plane Development moved from the system test phase

to continuous assurance of the system quality. Setting up Release Areas did this. Release areas cover the system legacy. That is, they cover non-functional requirements, which specify criteria that can be used to judge the operation of a system, rather than specific behaviors. The operations related to these requirements can be categorized representing groups such as characteristics, robustness, upgrade, security, quality of service, and stability of the system. This grouping is a base for the release areas. As said, the implementation in Media Plane Development for non-functional tests was a release area concept containing the release areas listed in the table below (Table 6.).

Table 6. Release Areas in Media Plane Development.

Release Areas in Media Plane Development		Scope of the area
1	Upgrade and Expansions	Secures that MRS node is upgradeable and expansion able between all SW tracks.
2	Operation & Maintenance	Secures that the quality of non-functional requirements related to operation & maintenance is acceptable for all product packages.
3	Signaling	Secures that different signaling configuration work including the different signaling standards TTC, ANSI, CWTS and ETSI.
4	Single Traffic & Features	Securing that single call cases used by the customer is working.
5	Media Quality	Secures media plane quality of the MRS node in all SW tracks.
6	Stability	Secures the MRS node operates stabile according to requirements in all SW tracks.
7	Robustness	Secures that the quality of non-functional requirements related to robustness is acceptable for all product packages.
8.	Characteristics & In Service Performance	Secures that all characteristics and ISP requirements are met in all SW tracks.
9.	Vulnerability	Secures that there are any vulnerability problems in the product.

The independent test teams perform the tests in the release areas listed in the Table 6. Testing is performed on potentially deliverable software (weekly) packages (sometimes also on the daily common software package) providing continu-

ous feedback of the system quality. The Feedback is provided from all the release areas by using the radiators, as shown in the figure below (Figure 6):

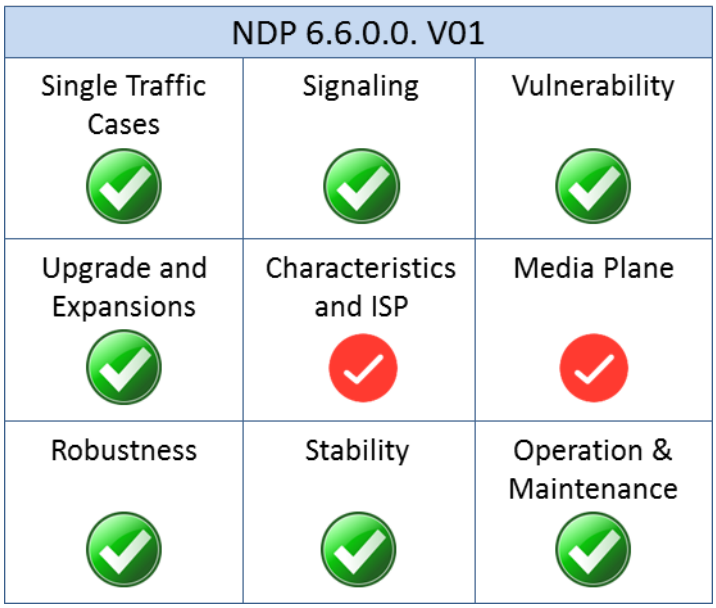

Figure 6. Radiator for all the release areas in Media Plane Development.

Figure 6 above illustrates an example how does the radiator look like for a software package, here Node Delivery Package 6.6.0.0. V01. Each of the release areas may be either red or green. The color depends if the quality is sufficient for a release, or not. The independent test team, that is responsible of the corresponding release area, will set the color based on the results of their tests. This radiator view is made visible to all employees working in Media Plane Development by showing it in the TV screens on the corridors and rooms where the teams are sitting.

3.1.4 Feature Integration Tests

Where the *Release Area concept* covers non-functional requirements of the system, the Feature Integration covers integration of new features in the system. In other words, it covers the functional requirements. Feature integration is done in Network Environment equipped with the controlling nodes Session Border Gateway (SBG) and Mobile Switching Server (MSC). Part of it is end-to-end network integration testing, which means that it is performed with real end-to-end clients. Feature integration tests contribute also to release areas Stability and Robust-

ness. This happens by performing short network level load test and some robustness type of tests, like for example, restarts while integrating new features on the network level.

3.2 The Current Ways of Working

Presently, the Media Plane Development is using Agile methodology as its main method of working. It has been used since year 2009. There are so-called cross-functional development teams working in three countries (Finland, Hungary and United States). In addition to that, there are also independent test teams supporting cross-functional teams in system test area, both in functional and non-functional system tests.

The current way of working is built around the following principles: a) one-track development, b) test automation, c) continuous integration, c) testing in cross-functional and independent test teams, d) a proper test analysis and planning. They are described in more detail below.

3.2.1 One-Track Development

Using One-Track method in the development means that all the development is done on the main-track. Cross-functional teams commit their newly made code in the main-track (i.e. main-branch) several times per day. After that all the testing is performed in the main-track. The principles of the one-track development are illustrated in the figure (Figure 7) below:

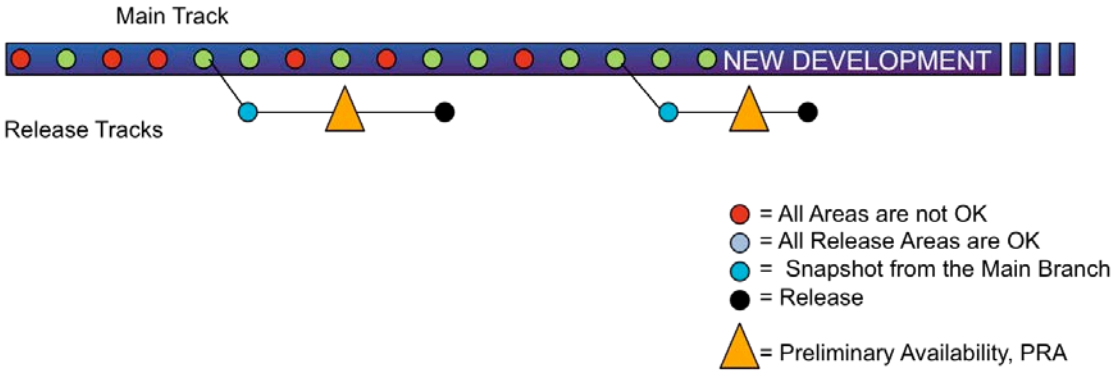

Figure 7. One-Track Implementation in Media Plane Development.

Figure 7 above shows, that the One-Track means exactly one track only, from the development point of view. For releases a parallel release-track is created, but that is only for testing purposes. The release-track is a snapshot of the main-track on certain time.

The dots in Figure 6 are defined in the following ways. First, *the red dots* in the main-track mean that the quality is not acceptable on all of the release areas (see Section 3.1.3). That is, the product quality is not sufficient for a release. Second, *the blue dots* are “snapshots” of the main-track software. It means that the final compile has been made and no changes in SW are possible anymore in the release-track. This parallel release-track is only for testing. The date/week when the release-track is made is dynamic. It’s more effective to fix the code in the main-track and take a new snapshot than fixing the defect in both the main and the release-track (Otherwise in practice it would mean “two tracks” open also for development all the time. Now the extra track is limited on test activities only.)
Finally, *the black dots* mark the stage when the release is made. In practice it means that the developed SW is put on the SW gateway and is accessible to all the customers

One-track development is an enabler for the continuous deliveries and it is a way to save in maintenance costs. So far, the benefits from One-Track development for Media Plane Development have been: 1) the possibility to release SW package once per week if needed, 2) the correction mapping (of defects) between

releases is minimized, and 3) the number of supported upgrade paths is much smaller than it used to be.

3.2.2 High Demands on Test Automation

In Agile ways of working the new features are developed in frequent time periods, in sprints. A sprint in the Media Plane Development is two weeks. Within these sprints development teams commit their new software once per day, even more often. The teams should get fast feedback after every commit about the quality of the software. That is, information about the quality of both their new code, and the quality of the legacy functionality. Quality of the legacy means that the functionality that has worked in the earlier deliveries, should still work properly. This requires frequent and fast regression test execution. Frequent regression testing requires test automation.

High automation rate of tests is a key in an Agile development. In Media Plane Development tests are automated whenever it is feasible, and what is more important, whenever there is a business case for automation. The most fertile area for test automation is in regression testing, where the same tests are repeated frequently. Business case thinking includes the maintenance costs of the automated cases.

3.2.3 Continuous Integration

Continuous integration (CI) in Media Plane Development provides a fast, automated, feedback about the quality of the developed software. One of the main rules in the Agile development is: “Do not break the legacy”. That is why the Continuous integration in Media Plane Development focuses on the legacy part of the product. The continuous integration cycle is repeated several times per day; therefore it needs to be automated.

In Media Plane Development the continuous integration machinery is fully automated. The principal flow of it is illustrated in the figure below (Figure 8):

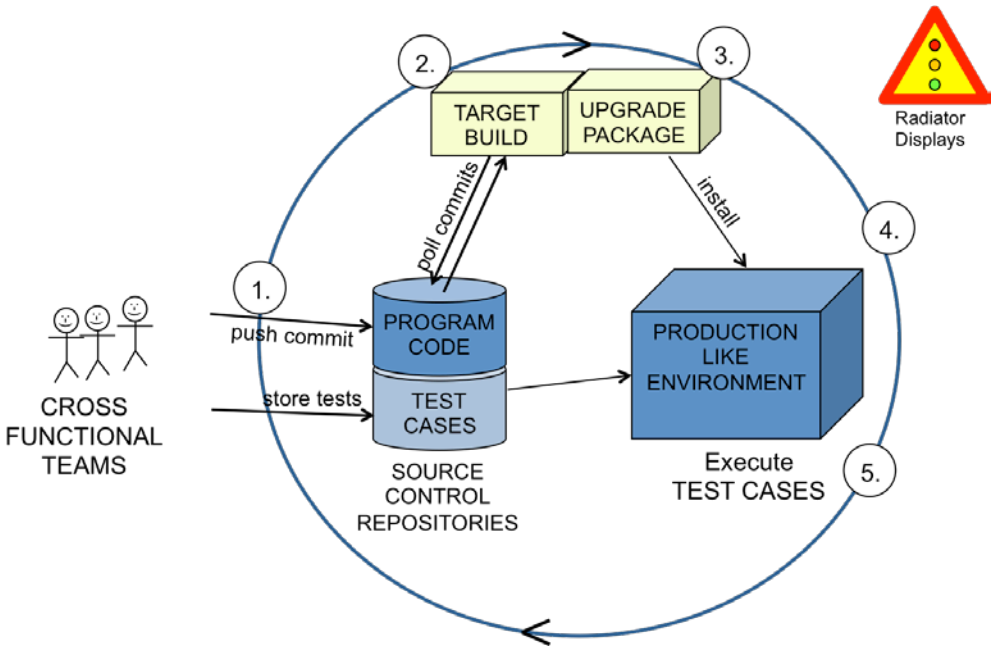

Figure 8. Principal flow of CI in Media Plane Development.

Figure 8 above shows the principal Continuous Integration (CI) flow that is built around the steps listed in the table below (Table 7.). Radiator displays are used to visualize build and test results.

Table 7. Basic Steps in the Continuous Integration.

Steps	Action
1	Checking-in, committing new source file content into version control system
2	Compiling new source code into target binaries
3	Building an installable upgrade package from new version of target binaries
4	Upgrading the test node with the newly created package
5	Executing test campaign on new software, to verify its legacy and newly added functionality working

Table 7 shows the steps in the automated Continuous Integration (CI) cycle in Media Plane development. The first step is a "human interaction" and the upcoming steps are automatically initiated after each other. The first step is the most important step in the process. If this has not taken, there is nothing to build, integrate or test, nothing continuous. Thus, frequent commitments must be part of the working culture. In other words, developers should commit a newly implemented code frequently.

To explain the flow in more detail, CI Automation is continuously polling for commits. Commit triggers target build, which is followed by creation of an upgrade package. The upgrade package is then installed in to the production-like test environment where automated tests are run. This is repeated four times per day on the main-track for all committed code. The tests in CI cover functional tests; short stability and also some selected non-functional system test cases. The functional test cases used in the automated test runs, are made by the cross functional teams. They have stored the tests (i.e. test suites) in the source control repository. A selection of all functional test cases is marked with specific tags, and test automation in continuous integration machinery executes these cases directly from the repository.

The latest run of the day will make a new common software package. Teams will perform functional tests on this package. The automated system level tests are run once per day on this daily common software package. Once a week, one of the daily packages is selected to be a weekly package (that is also a potential delivery package). Independent test teams perform their tests on this weekly package, but they are also often asked/allowed to take daily package e.g. in a case of platform changes. This is to get fast feedback about the changes.

3.2.4 Testing in Teams

Agile methodology suggests that all the testing should be performed by the development teams, cross-functional teams. In an ideal case this would work, but in large-scale software development project this is not always the case. In large-scale software development, there might be time constraints, competence and cost efficiency issues that prevents of doing this. This is also the case in the Media Plane development, where in practice; the scope of the testing in is divided between 1) *cross-functional teams* and 2*) independent test teams*.

First, in *cross-functional teams*, testing covers tests that gives fast feedback to teams themselves. This contains lower level tests, such as unit and component tests in development environment, and also system level functional testing including possible some load tests. System level functional tests are executed in the target, production-like, test environment. Second, the independent test teams

perform non-functional and functional testing (including integration aspects) in the network test environment. The functional testing focuses on the integration of the new features, whereas, the non-functional tests contain, for example, robustness tests, stability tests and characteristics tests. These non-functional tests were covered by Release Area concept (as described in Section 3.1.3).

The reason for having such independent test teams is that, otherwise these tests would belong to test scope of all or at least many cross-functional teams. In practice this would mean that each and every team should have network test environment with system testing specific tools, such as, load generators. This would then mean that all teams should have also in many cases specific competence to perform these tests. All this is very expensive and even impossible when it comes to the competence. Everyone cannot be a specialist in everything. This is why the current test environments, test tools and competencies are concentrated in independent test teams specialized on their own areas.

3.2.5 Test Analysis and Planning

Test analysis and planning are in a centric role in large-scale development. When there are many teams working for the same product, it is important that the testing is planned well to avoid overlapping testing and also that anything will not be omitted. Basis for planning is test analysis. It will make impacts on testing visible on time, such an impact is, for example, a need for a new testing tool, because of the new feature or other requirement. This kind of impacts need to be recognized as early as possible, since it may take several months to get the needed tool into the use. The current test analysis and test planning work is implemented in the Media Plane Development as described on high level in the following.

Presently test analysis is done mainly in two activities. First, high level test analysis is done *in the Early Phase Program*, where the test analysis results in high-level information about the impacts on testing caused by the development of the new feature. This includes impacts on test tools and test environment. It covers also the cost estimation, hardware forecast, effort and lead time estimates. This analysis may be triggered by various sources, but usually the Product Management does it. The analysis outcome is then included in the one-pager of the feature, which is produced by the Early Phase Program. The product owner function,

cross-functional teams and independent test teams are responsible to make test analysis in the Early Phase Program.

The second step taken in test analysis is the Feature Concept Study where features are analyzed in more detail including, for example, possible change requests, external impacts, platform impacts, and also tool, environment and HW impacts. The results of the analysis may be recorded as new user stories for tests (requirements for testing). The new user stories are then stored in the product backlog. User stories may also include HW orders, work orders for test tools or environment and network plan tasks. This part of the test analysis contains the overall test analysis of the features and cost in terms of story points. Additionally timing of the features is part of the analysis. This includes, for example, feature dependencies, test and platform dependencies, not forgetting the tools and HW lead times in case of new orders, or changes. All the results of the analysis are collected into the Feature Concept Study documentation. The product owner function is responsible to make test analysis in Feature Concept Study, but the participants from cross-functional teams and independent test teams support this analysis.

Summing up, the Test Plan is based on the overall test analysis included in the feature concept study document. The Overall Test Plan contains estimations of the overall resource needs for testing for a feature development and a release project. More detailed test planning is made in the cross-functional teams as part of sprint planning and also as part of release project planning.

3.3 The Current System Test Environment

This section discusses the current system test environment. The section starts by presenting the system test network, and continues with describing the test tools.

3.3.1 Test Network

The current test network is an MSS/IMS network with real network components, such as, a real Mobile Switching Center (MSC) and a real Mobile Telephony Application Server (MTAS). Presently, there are several similar test environments in use containing different hardware configurations of the Mobile Resource System

(MRS), the logical configuration of the current Border Gateway Function (BGF). The figure below (Figure 9.) illustrates the current system test network for BGF.

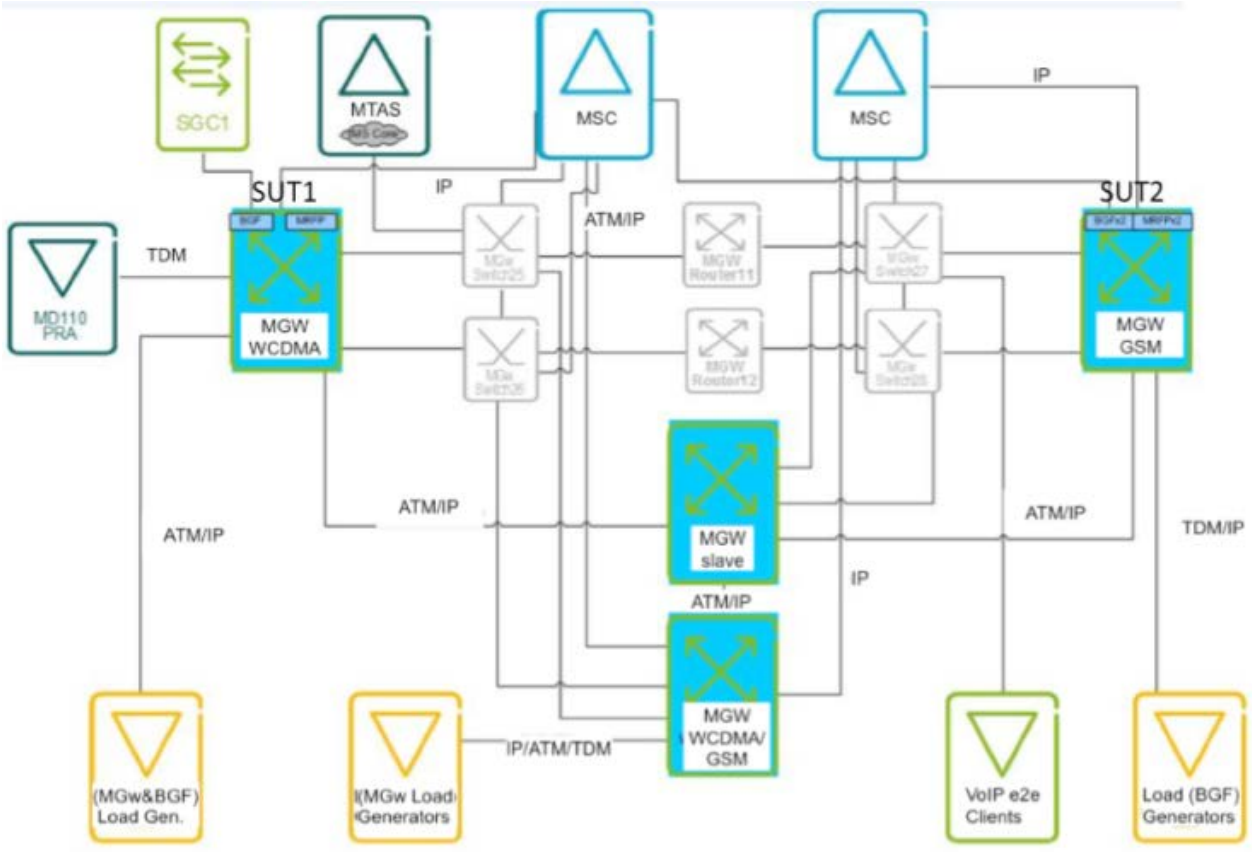

Figure 9. The current BGF system test network environment.

In the BGF system test network environment seen in Figure 9 above, all the surrounding nodes of the systems under test (i.e. BGF nodes marked with SUT1 and SUT2 in the figure) are native nodes.

3.3.2 Test Tools

Currently, the main tools in the system test are the traffic load generators. They are used to generate both control plane and user plane traffic to the network. Most of the non-functional system test cases are executed with background traffic, loading the system under test nodes about 80% of their full capacity. The same traffic generators are used also to perform stability, load and over load

tests, and characteristics tests. In addition to traffic load generators, the real VoIP end-to-end clients are used for example in the Feature Integration tests.

In the Media Plane Development, test automation has been implemented by test automation framework based on the *Testing and Test Control Notation version 3* (TTCN-3). This framework is a common test framework in the Media Plane Development, used for testing in cross-functional teams, testing by continuous integration machinery and for non-functional system tests in independent test teams. All the test generators are controlled with this framework.

3.4 Summary

As demonstrate in the analysis above, today system testing for BGF is performed mainly on node level covering also partly the network level testing. The cross-functional agile teams and independent test teams perform these tests. The cross-functional teams perform majority of the node level feature integration, whereas the independent test teams perform the network level feature integration. Tests that are covered by the release area concept, forms the majority of the non-functional tests. The independent test teams perform these tests. The role of the cross-functional teams in the area of non-functional tests is not that big.

One-track is an enable for frequent deliveries, but on the other hand, it is the Continuous Integration (CI) that makes it possible to work using one-tack method. CI allows securing the legacy functionality providing fast feedback of the quality. It is fully automated, and that is the only way to keep the pace considering to the frequent commits, the frequent new software packages.

The test environment is a complex network environment including the real IMS networks nodes, where the traffic generators generate the traffic. Majority of all tests is automated by using the common test framework. For the change to the virtualization it means that a) the test strategy, b) the ways of working, and c) the test environment will need to be changes.

4 Network Functions Virtualization: Concept and Requirements

This section discusses the findings from the existing knowledge related to network functions virtualization and its impact on system testing, as these topics are now discussed in literate and publications. The section starts with an overview of the Network Functions Virtualization (NFV) as a concept, and then continues by presenting the Network Functions Virtualization environment, its failure models and its complexity. After that the section will present the requirements that will impact system testing. These are 1) the general ETSI NFV (ISG) virtualization requirements, 2) other industrial requirements.

4.1 Concept and Overview of Network Functions Virtualization

Network Functions Virtualization (NFV) is a trend in today's telecom business. The following text from a non-proprietary white paper authored by network operators clarifies the motivation for the network virtualization concept:

> *Network Operators' networks are populated with a large and increasing variety of proprietary hardware appliances. To launch a new network service often requires yet another variety and finding the space and power to accommodate these boxes is becoming increasingly difficult; compounded by the increasing costs of energy, capital investment challenges and the rarity of skills necessary to design, integrate and operate increasingly complex hardware-based appliances. Moreover, hardware-based appliances rapidly reach end of life, requiring much of the procure- design-integrate-deploy cycle to be repeated with little or no revenue benefit. Worse, hardware lifecycles are becoming shorter as technology and services innovation accelerates, inhibiting the roll out of new revenue earning network services and constraining innovation in an increasingly network-centric connected world.* (ETSI 2014-10-22:3)

This definition illustrates the multiple problems the network operators have today related to hardware-based appliances. These appliances have become increasingly difficult; the life span of the appliances has become shorter. All this has led to need for change, and this change is Network Functions Virtualization. The

Network Functions Virtualization aims to address these problems by using standard IT virtualization technology to get different types of network equipment onto industry standard high volume servers, switches and storage. The equipment could then be located in Datacenters, Network Nodes or in the end user premises. This method is expected to be applicable to any data plane packet processing and control plane function in fixed and mobile network infrastructures. (ETSI 2014-10-22:3)

On a high-level Network Functions Virtualization can be seen as implementation of network functions as software only entities running over the Network Functions Infrastructure (NFVI). There are three working domains in NFV. They are listed in the table below (Table 8).

Table 8. The working domains in NFV. Data gathered from ETSI 2014-12 a:10.

	Domain in NFV
1	Virtualized Network Function (VNF), as the SW implementation of a network function, which is capable of running over the NFVI.
2	NFV infrastructure (NFVI), It includes three domains, a hypervisor domain, a compute domain and a network domain, including diversity of physical resources and their virtualization.
3	NFV management and orchestration (NFVO) a) covering the orchestration and lifecycle management of resources, physical and/or software, b) supporting virtualization of the infrastructure, and the lifecycle management of VNFs, and c) focusing on all virtualization-specific management tasks that are needed in the NFV framework.

Table 8 above lists the three domains that form the Network Functions Virtualization framework, which is illustrated in the figure below (Figure 10):

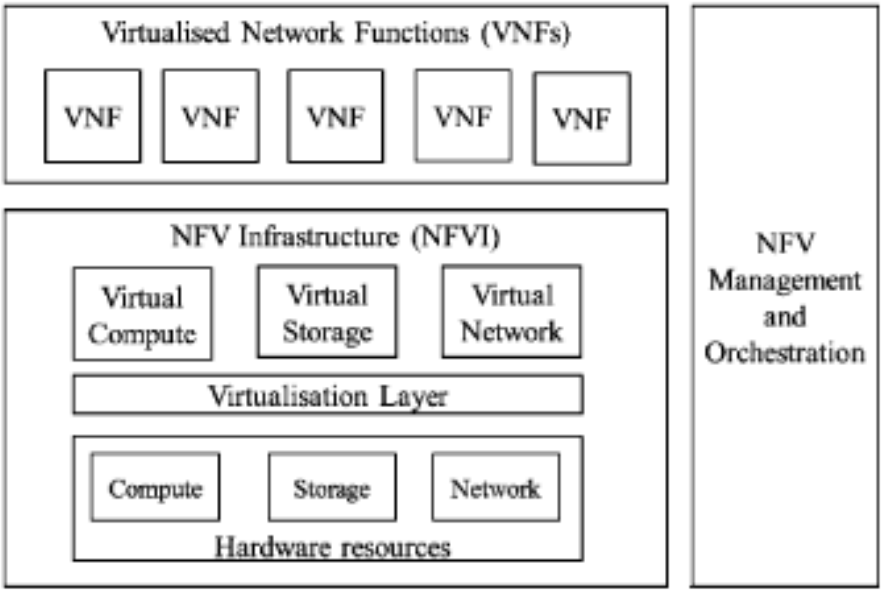

Figure 10. High-level NFV framework. Reprinted from ETSI 2014-12 a:10.

Figure 10 above illustrates the Network Functions Virtualization framework, which enables dynamic construction and management of VNF instances. The framework enables also the relationship between VNF's as for, for example, data, control, management and dependencies. (ETSI 2014-12 a:10)

Benefits of Network Functions Virtualization

Virtualizing Network Functions could potentially offer many benefits including: a) reduced equipment costs which can be achieved through consolidating equipment and utilizing the economies of scale; b) reduced power consumption which is also achieved by consolidation and scaling, c) faster Time to Market interval by minimizing the typical network operator cycle of innovation. (ETSI 2014-10-22:3)

Investments into the new functionalities are expected to be much lower for SW-based development than they have used to be for hardware-based functionalities. This is because the Network Functions Virtualization should enable network operators to reduce the needs to buy new hardware, and additionally also provide the following additional benefits, listed in the table below (Table 9):

Table 9. Additional benefits when using NFV. Data gathered from ETSI 2014-10-22:3.

	Benefit
1	Availability of network appliance multi-version allows use of a single platform for different applications. This allows network operators to share resources across services and across different customer bases.
2	Targeted service introduction based on geography or number of customers is possible. It means that the services can be scaled up/down rapidly as required.
3	Enables a wide variety of eco-systems encouraging openness. It opens the virtual appliance markets to new pure software companies, small players and academia. This will encourage bringing new innovative services and new revenue streams quickly at much lower risk.

Table 9 lists the additional benefits that are expected from Network Functions Virtualization. The first listed benefit means multi-vendor environment and many different configurations, the second leads to the scalability requirements, and the last one will bring new vendors, also not traditional telecom vendors, on the application markets. This may lead new types of situations in operation and maintenance later on.

4.2 Network Functions Virtualization Environment

This section discusses the Network Functions Virtualization environment, its failure model, its complexity and testing related aspects.

4.2.1 VNF Failure Models

Presently, there are four failure models that are introduced by moving network functions into a virtualized environment. These four models are dependent of the option selected for the Virtual Network Functions (VNF) deployment. Based on the option selected the impact of failure will vary and the restoration method has to be different. These four different deployment options are presented in the figure below (Figure 11). (ETSI 2015-01a: 36).

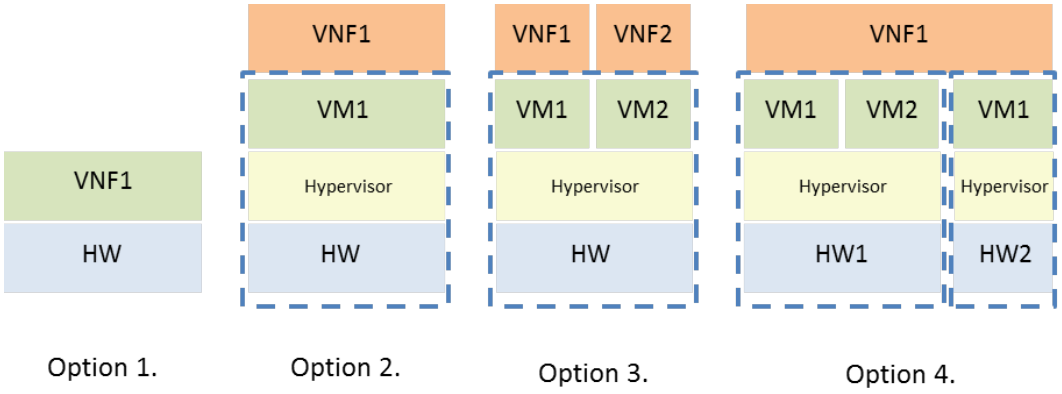

Figure 11. Deployment Options of VNF. Modified from ETSI 2015-01a:36.

The first option (Option 1.) in Figure11 above is the classical “box-model” where the function is run on the hardware. This is comparable to the native products. The second option (Option 2.) is the simplest approach to virtualization. The existing software providing a service is installed into a virtual machine (VM) image and executed on the virtual resources provided by the hypervisor. The additional software in the system adds new failure models to the system, which are, for example, failures on hypervisor level that did not exists in the box-model (i.e. in the native product). The third option (Option 3.) is based on the aim of high hardware utilization, the physical resources are sliced into a set of virtual resources providing multitude of services. That is, several VNFs can be hosted on the same physical host. This will introduce another set of failure models to the system, which are, for example, negative performance impacts, if resource isolation does not work. The fourth option (Option 4.) is that the VNF is spanned across the physical and virtual host boundaries. That is a case when a VNF is composed from multiple VNF components each having own virtual machine on the same or on a different physical host on the cloud. This will bring new failure models to the system, for example, simultaneous failures of multiple VNF components due to a failure of the underlying hardware or VNF failures due to communication failures between its components. (ETSI 2015-01a:36)

Based on these deployment options, there might be even three different failure models more compared to the native product. All the different deployment options, failure models need to be considered in testing. This will naturally impact test environment.

4.2.2 Complexity of the Environment

Virtualized Network environment is known for its complexity. The elements pointing to its complexity are described below. They were collected from the following sources: ETSI (2014-12 h), Daitan Group (2014), EZchip (2014), Spirent (2014), Nair V. and Gupta V.K. (2014), Ya-lian Pan et al (2011), Cotroneo D. et al (2014) and Kanso A. and Lemieux Y. (2013).

There are two points that may be used to reason testing in a real environment. First, compared to hardware-based solutions, which have mostly fixed *capacity, dimensioning of NFV systems is more demanding*. It means that testing in a real environment is usually easier than trying to dimension based on specifications. Second, using the real environment may also be reasoned by the fact that there are *several new components in a virtualized environment* that will require a different approach for testing compared to the environment of the native products. For example, *hypervisor* and *the generic hardware platform* can impact applications. Therefore, the NFV components need to be tested in a specific virtualization environment. (Daitan Group 2014:6)

Kanso A. and Lemieux Y. (2013) suspected if the cloud technology would be ready to host applications adhering to telecommunication-grade requirements. The issue in the cloud would be *the high availability*. Infrastructure-as-a-Service (IaaS) provided by the cloud, are agnostic of the type of applications, and therefore they do not provide protection mechanisms against failures at the application level. This means that the IaaS providers are not affected when a software application running in a virtual machine (VM) fails as long as the VM is healthy. This type of failure will probably remain undetected by the Cloud middleware. On the other hand, this type of failure directly impacts the Cloud users, since it is interrupting their services. Additionally High Availability solution should cover all the possible reasons for service downtime, which is only caused by failures. A significant portion of the service downtime is, planned outages due to *software upgrade* and *maintenance account*. This means that the High Availability solution should take into account, and enable the applications running in the cloud to continue providing their services even during their upgrade. (Kanso A. and Lemieux Y. 2013:778).

Additionally, complex load balancing increases the complexity of the environment. It is necessary to distribute the workload over multiple cores and Virtual Network Functions. This is done by load balancing. Moving the data between cores may have a negative impact on the performance. Successful delivery of all data packages is a flow that requires a stateful load balancer. This *will cause additional latency*. Additionally network devices are based on flow. An increasing number of subscribers causes that also the size of the *flow tables will grow. Finally they will exceed the cache capacity of standard servers*. The size and number of tables depends on the function and location of Virtual Network Functions that incorporate with OpenFlow switch functions (this is assumed to be a generic problem among virtual switches). (EZchip 2014:6)

Not only the load balancing and capacity are demanding, also a *performance monitoring as part of orchestration and service management is a challenge.* Large NFV networks must support effective performance monitors. This means a large number, maybe millions of stateful flow tables. The performance monitoring needs to be capable of collecting, maintaining, and possibly also analyzing a large number of counters. Additionally also *Security on VNFs* may require some attention when sensitive data is transferred to the data center or a server is shared between VNF and VMs running user applications. Many isolation techniques are required to provide secure connections and services for multiple tenants on the same network, server or core. The network and server infrastructure must support having multiple multi-vendor VNFs. In some cases the flows may require large buffers, and in other cases the flows may require low latency. In all of the cases the transmit schedulers should be capable of managing resource usage, for example, between users, groups of users, ports and channels on ports. (EZchip 2014:6, 7)

Another traffic related issue is the difficulty to ensure proper routing of traffic in a virtualized network, since the traffic flow will behave dynamically based on the configuration and the existing network load. This adds complications for testing. (Nair V. and Gupta V.K. 2014:5,12)

Practitioners see the following differences between the traditional and virtualized environments summarized in the table below (Table 10.):

Table 10. The Matrix Problem by Spirent. Data gathered from Spirent 2014: 5.

Traditional Approach	Virtualized Data Center
Dedicated Hardware	Shared Hardware
Controlled Hardware / OS Settings	Matrix of Variables
Controlled Environment	Unpredictable Environment

Table 10 shows that moving from traditional HW-based systems to SW-based virtualized systems will bring more complexity and unpredictability to the environment. The Virtualized Network environment will introduce also some *new problems in fault diagnostics.* The reliability of the services in the virtualized network environment relies on the network's capabilities to diagnose and recover faults. This might be more challenging compared to the traditional network environment, especially when the failure is caused by, for example, the virtual layers. Faults in the underlying physical network components may result in complex problems. For example, any physical link failure may cause failures of all the virtual links passing through. In general, the virtual network environment will introduce new potential fault areas compared to the traditional network. These are, for example, *more complicated functions*, *migration of Virtual Machines*, *abnormal latency or packet loss*. Usually this kind of faults can be observed as certain end-to-end network disorders. (Ya-lian Pan et al 2011:517)

For testing, the complexity of the environment puts also requirements for testing tools. For example, in a virtualized environment, network elements are distributed, that is, network elements providing same service can be placed at different physical location. This means that there is *a need for specialized testing tools, which can collect data, analyze and report exact faults points*. (Nair V. and Gupta V.K. 2014:5,12)

All said above explains the complexity of the Network Functions Virtualization environment. This complexity, or even unpredictability, may be experienced as a problem or at least as a challenge in testing.

4.2.3 Testing Aspects Related to NFVI and Network Services

Reliability of The Network Functions Virtualization Infrastructure may be an issue. It includes all hardware and software components, which build up the environment in which VNF's are deployed, managed and executed. That is, it contains common elements of cloud computing such as physical computing, network and storage resources and resource pooling mechanisms. The NFV framework comprises NFV application domain hosting VNFs, and the Management and Orchestration (M&O) domain to control and manage software appliances running on the infrastructure. (ETSI 2014-12 a). These domains are illustrated in the figure below (Figure 12):

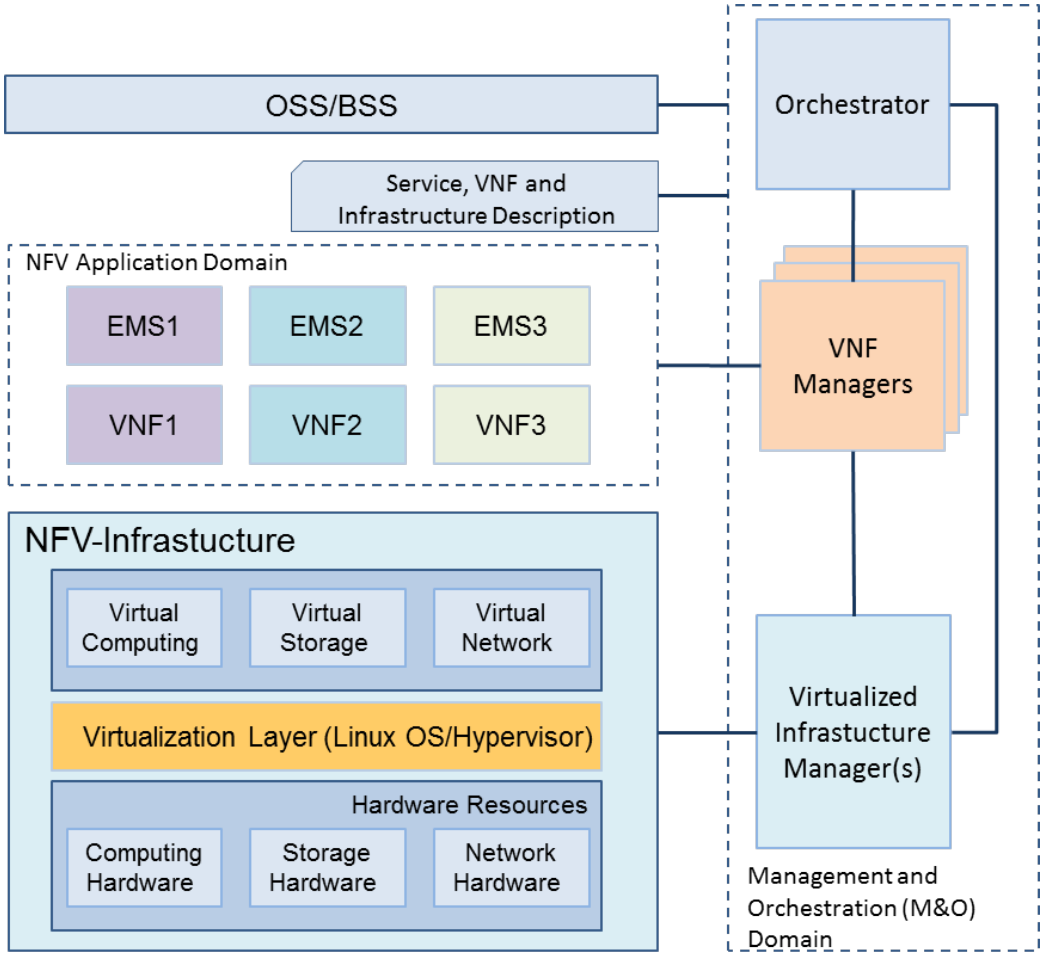

Figure 12. NFV Reference Architecture Framework. Modified from ETSI 2014-12 a:14.

Figure 12 above shows the architecture and the components of the Network Functions Virtualization Framework. This architecture will introduce potential causes of VNF failures, which are listed in the table below (Table 11.):

Table 11. Potential causes of VNF failures related to NFV Infrastructure. Data gathered from Cotroneo D. et al 2014:39.

	Potential VNF Failures
1	Hardware faults: COTS HW used in NFVI may negatively affect the VNF's running on them.
2	Software faults: at various levels, such as Host OS, Hypervisor, VM, or the VNF instance itself.
3	Operator faults: mistaken operations and configurations, for example, capacity planning, VM deployment and migration.

Table 11 lists the potential failure situations related to NFV infrastructure. In addition to these, there are also situations in the Network Services (NS) lifecycle that may require testing when performed. That is, for example, a performance testing that is to be done before the service goes live. Testing a single VNF instead of the whole Network Service is a subset case, but necessary before placing a VNF in service. (ETSI 2014-12 h).

NFV infrastructure is one of the areas that require consideration when planning testing, especially in an environment where the building blocks (e.g. Host OS, Hypervisor) may come from different vendors.

4.2.4 Benefits of NFV in Testing

Testing may also benefit of Network Functions Virtualization. This is, although all what was said above about the NFV environment may be considered challenging, or even negative. One of the benefits of the NFV for the telecom operators was the flexibility of the environment, possibility to use different HW and different cloud setups. The same applies also for testing, and test environments. In addition to the flexibility in the test environment, NFV will also make it reasonable to use virtualized test tools. When both the system under test and the test tools are virtualized they may be moved from a place to another. This brings totally new possibilities compared to testing of the non-virtualized products. By having everything virtualized it is possible to setup test environments on the need basis very quickly as Test-Environment-On-Demand. This would enable also buying or renting HW and cloud outside the local company.

4.3 ETSI NFV Virtualization Requirements Impacting System Tests

The ETSI network operator-led Industry Specification Group (ISG) for NFV has defined requirements for Network Functions Virtualization. The requirements that will cause changes in system tests are discussed on high level in the following sub-sections. These requirements are from document with title: ”Network Functions Virtualisation (NFV); Virtualisation Requirements” (ETSI 2013-10a).

4.3.1 Partially or Fully Virtualized Network Functions and Portability

Presently, it is required that the service providers or network operators should be able to *partial or fully virtualize* network functions that are needed to create, deploy and operate the services they provide. Partial means that a set of specific functions (e.g. based on a similar characteristics) within a network system or subsystem is virtualized. For example, virtualization of network functions on control plane, but not the ones in the data plane. This is valid requirement especially in transformation phase, when adding virtualized network components into the network.

Another generic requirement is a requirement of *portability*. The requirement of portability represents the philosophy behind the whole idea of virtualization. Portability means that in the NFV Framework it should be possible to load, execute and move Virtualized Network Functions (VNFs) across different but standard Network Point of Presence (N-PoP) multivendor environments. That is, the NFV target is to achieve VNF portability across multiple vendors, hypervisors, and hardware. (ETSI 2013-10a:7).

As said the requirement of *Partially or Fully Virtualized Network Functions* is valid in the transformation phase. The operators will not transform all the network components at the same time, but this is done gradually. This means that the transformation period may be long. The requirement of *Portability* is one of the key requirements in virtualization. More about portability can be found in Appendix 1.

4.3.2 Performance

For any running VNF instance, the NFV Framework should be able to collect performance related information regarding the usage of compute, storage and networking resources by that VNF instance. The framework should also be able to collect *performance* related information concerning the resource usage at the infrastructure level (e.g. hypervisor, Network Interface Controllers, virtual switch) of compute, storage and networking resources. (ETSI 2013-10a:8).

To collect and measure performance data will be challenging in the NFV environment due to the multi-vendor set-up. More about the performance issues can be found in Appendix 2.

4.3.3 Resilience, Elasticity and Service Continuity

The general NFV requirement contains a *Resilience* requirement, which relates to the fault recovery, service availability and service continuity. The NFV Framework should be able to provide a mechanism to allow network function to be re-created after a failure. Both on-demand re-creation and automatic re-creation should be supported. Resilience requirement also means the metric on the network stability (e.g. packet loss rate, latency and delay, failure rate in transactions). (ETSI 2013-10a:9)

In addition to the general NFV requirements mentioned above, ETSI has published also a specific requirement document for *Resiliency Requirements* (ETSI 2015-01a) with title: “Network Functions Virtualisation (NFV); Resiliency Requirements”. This document defines, for example, resiliency use cases, resiliency principles in NFV environment, fault management in NFV and service availability.

Requirements for *elasticity* are to be connected to resiliency requirements. Elasticity requirements are part of the scalability requirements. Automatic scaling of a VNF instance can be triggered based on the pre-defined criteria, like for example, different failure situations, or should it be called automatic re-creation instead of scaling. Anyway elasticity is also a resiliency issue. (ETSI 2013-10a:8)

One example of elasticity case was presented in Proof of Concept demonstration, where a hardware failure was used as an example of elasticity (Csatári G. and László T. 2013).

In addition to elasticity, also some of the *service continuity requirements* are to be connected to resiliency requirements. In general the service continuity is subject of Service Level Agreement (SLA), but the three requirements in the table below (Table 12.) under the service continuity extend the *Elasticity* and *Resilience* requirements (ETSI 2013-10a:10, 11).

Table 12. Service continuity requirements related to Elasticity or Resiliency. Data gathered from ETSI 2013-10a:10, 11.

	Requirement
1	In a case of a hardware failure or a resource shortage/outage, the NFV framework provides mechanisms to restore impacted VNF instance(s).
2	If VNF instance or a subset of instances needs to be migrated, the NFV will consider service continuity in the migration process. The possible impacts should be measurable.
3	In a case VNF instance or a subset of instances, for example a Virtual Machine is migrated, the communication between the migrated instances and other entities, for example, physical network element, should be maintained independently of its location and awareness of migration.

Table 12 above lists three service continuity requirements than can also be considered as a resilience or elasticity requirements for HW-fault situations or migration of Virtual Machine.

4.3.4 Security

New security aspects should be taken care by the NFV Framework. These are, for example, vulnerabilities introduced by the virtualization layer; usage of shared storage resources or network resources, new interfaces e.g. HW or management systems, access to NFV functions via APIs. (ETSI 2013-10a:9,10)

The overall security of a system is almost always dependent on the security of underlying abstractions and interfaces. The following aspects need to be considered: authentication, authorization, privacy, and auditing (e.g. logging of valid and denied accesses). (ETSI 2014-10 b)

There is a separate ETSI document about the security (ETSI 2014-12b) with title: “NFV Security; Security and Trust Guidance”. This is not a requirement, but rather guidance about the security issues for NFV development. The document brings up a new aspect in security area, like for example, that in Network Functions virtualization; the administration of the network may also be done as a multiparty activity. The controlling and maintaining of the network may be done by separate departments, or even by different companies. To be able to control a stable trust relationship there are options to divide the administration to different domains. Scenario of the domains is illustrated in the figure below (Figure 13):

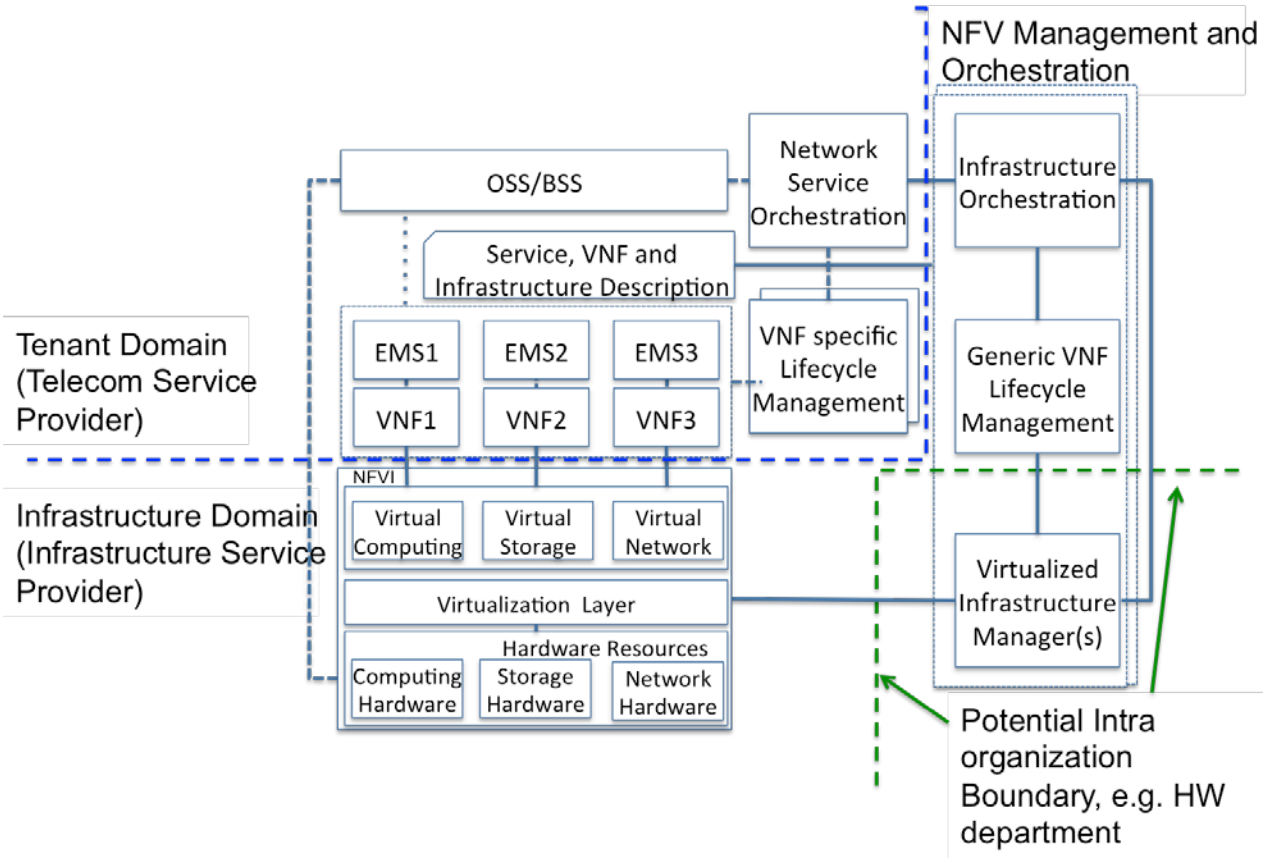

Figure 13. Administrative Domains. Modified from ETSI 2014-12b:22.

Figure 13 above illustrates a scenario of having “Tenant Domain” and “Infrastructure Domain”. The Infrastructure Domain is to administrate resources such as compute, networking and storage. The Infrastructure Domain may be divided then to sub-domains, for example, physical networking, and data-center environment. The Tenant Domain is to administrate Virtual Network Functions (VNFs).

This kind of division will impact security in a way that there need to be well defined reference points between the Tenant and Infrastructure Domain. The NFV ISG proposes a separation of the Orchestration functionality and the Virtual Network Functions lifecycle management and an infrastructure part.

In addition to the requirements mentioned here, other security related aspects are presented in Appendix 3: “Security Issues in Network Functions Virtualization”.

4.3.5 Service Assurance

Since many network functions are virtualized, the virtualization infrastructure offers service assurance, by presenting the opportunity for running virtualized functions whenever and wherever they are needed. For example, service assurance may mean possibility to diagnose problems remotely, or monitor performance. In other words, the Service Assurance requirement relates to the network problem diagnostics, monitoring and measuring, for example, monitoring performance and measuring latency.

Service assurance requirement introduces the following needs: 1) The NFV Framework should provide mechanisms for time stamping of hardware (e.g. Network Interface Controllers, NICs). These mechanisms should also provide possibility to interrogate if the particular network interface hardware provides time-stamping facilities. 2) A (set of) VNF instance(s) and/or management system should be able to detect the failure in the instance and or network reachability to that and take action to meet the fault detection and remediation. 3) A Virtual Network Function, the orchestration functionality and/or a management system should also be able to determine whether the VNF is operating properly. (ETSI 2013-10a:10, 11)

Service assurance testing should verify that Network functions are remotely accessible, monitored, and can perform diagnosis (Nair V. and Gupta V.K. 2014:13).

The requirement of Service Assurance is a complementary requirement for service continuity, which is one of the key requirements in virtualization. The aim is to make it possible to monitor this continuity.

4.3.6 Operational and Management Requirements

Operational and Management requirements are related to the Operations and Maintenance system, which is called Orchestration and Management in NFV. The whole Operations and Management system will change compared to the existing native systems.

There are fifteen Operational and Management requirements in the ETSI NFV requirements document ETSI (2013-10a). The following three examples of NFV Framework related requirements draw a picture of the multifold aspects that need to be covered by operations and maintenance. First, the framework should incorporate mechanisms for *automation of operational and management functions*, such as, creation, scaling and healing of VNF instances based on the following criteria a) described in VNF information model, b) network capacity adaptation to load, c) software upgrades and new features/nodes introduction, d) functions configuration and relocation and e) intervention on detected failures. Second, the NFV Framework should also provide *a management and orchestration for VNF and VNF instances lifecycle management*; instantiation, allocation and reallocation resources, scaling, and termination. Including monitoring and collection of information related to usage. Management and orchestration should be able to interact with other operations systems when they exist. Third, *the framework should be able to manage NFVI resources* so that resources (compute hardware, storage, network) can be shared between VNFs (ETSI 2013-10a:11,12)

Overall, the virtualization principle leads to a multi-vendor ecosystem where the different components of NFVI, VNF software, and NFV-MANO architectural framework entities exist. Most likely they will follow different lifecycles, for example, on procurement and upgrading. Therefore, *there is a need for interoperable standardized interfaces and proper resource abstraction among them*. (ETSI 2014-12 h:14). The management interfaces are listed in the table below (Table 13.).

Table 13. NFV MANO Interfaces. Data gathered from ETSI 2014-12 h:14-29.

Management Interfaces	
Network Service management (provided by)	
1	Network Service Descriptor management (NFVO)
2	Network Service Lifecycle management (NFVO)
3	Network Service Lifecycle change notification (NFVO)
4	Network Service Performance management (NFVO)
5	Network Service Fault management (NFVO)
Virtualized Network Functions management (provided by)	
1	VNF Package management (NFVO)
2	VNF Software image management (VIM, NFVO)
3	VNF Lifecycle operation granting (NFVO)
4	VNF Lifecycle management (VNFM, NFVO)
5	VNF Lifecycle change notification (VNFM, NFVO)
6	VNF Configuration (VNF)
7	VNF Performance management (VNF, VNFM)
8	VNF Fault management (VNF, VNFM)
Virtualized resources (provide by)	
1	Virtualized resource catalogue management (VIM)
2	Virtualized resource capacity management (VIM)
3	Virtualized resource management (VIM, NFVO)
4	Virtualized resources performance management (VIM, VNFM)
5	Virtualized resource fault management (VIM, VNFM)
NFVI Management (provided by)	
1	NFVI hypervisor management (NFVI)
2	NFVI compute management (NFVI)
3	NFVI Network management (NFVI)
Policy administration (VIM, NFVO and VNFM)	
Network Forwarding Path Management (VIM)	

As seen in the table (Table 13.) above, there are many Management and Orchestration (MANO) interfaces in NFV. The three components NFV Orchestrator (NFVO), VNF Manager (VNFM) and Virtualized Infrastructure Manager (VIM) provide these interfaces. These components are illustrated in the figure below (Figure 14).

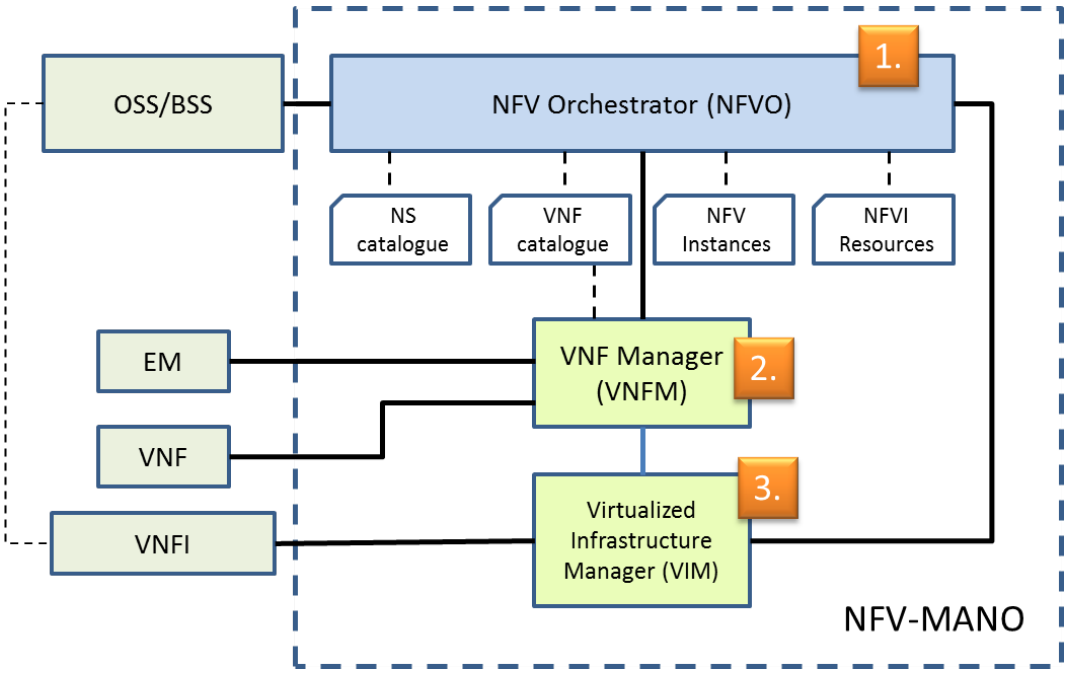

Figure 14. The NFV-MANO architectural framework. Modified from ETSI 2014-12 h:23.

Figure 14 above illustrates the architectural framework of the NFV Management and Orchestration. The table below (Table 14.) lists the functional blocks in the NFV-MANO architectural framework from Figure 14.

Table 14. Functional blocks of NFV-MANO Framework, Data gathered from ETSI 2014-12 h:25-28.

Functional Block	Description
NFV Orchestrator (NFVO)	NFVO is responsible of the orchestration of NFVI resources and the lifecycle management of Network Services.
VNF Manager (VNFM)	VNFM is responsible of lifecycle management of VNF instances.
Virtualized Infrastructure Management (VIM)	VIM is responsible for controlling and managing the NFVI resources (i.e. compute, storage and network

	resources).
NS catalogue	Network Services Catalogue is the repository of all the non-boarded Network services. It supports creation and management of Network Service Descriptors.
VNF catalogue	VNF catalogue is a repository of all the non-boarded VNF Packages. It supports creation and management of the VNF Descriptors, software images, etc. VNF packages.
NFV Instances Repository	Holds information of all VNF instances and Network Service Instances.
NFVI Resource Repository	Holds information about available, reserved and allocated NFVI resources.
EM	Element Management is responsible for configuration for network functions provided by the VNF, Fault management for the network functions provided by VNF, accounting the usage of VNF functions, collecting performance measurement results for the functions provided by VNF and security management for the VNF functions.

Table 14 above lists the functional blocks and their responsibilities in Management and Orchestration (MANO). The list shows a large amount of different functions and responsibilities that are handled by the different functional blocks. To make it even more complex, the fact is that, in the multi-vendor eco-system many of these functional blocks may be developed by different vendors.

4.3.7 Co-existence with the Existing Networks –Transition

There is a transition period, when deploying the new virtual network functions. Virtualization cannot happen at once for the whole network, it will be done gradually. It means that there will be a transition period when the virtual network functions and classical physical network functions co-exist in the same network. Therefore the NFV Framework should co-exist with legacy network equipment, and additionally it should be able to work with hybrid network composed of the classical physical network and virtual network functions. (ETSI (2013-10 a:13)

Co-existence requirement does not apply only for the network functions but also for the operations and maintenance of the networks. Therefore the NFV Frame-

work should be able to interwork with legacy management systems, for example, Operational Support System (OSS). In other words, the NFV Management and Orchestration (MANO) need to interwork with the existing OSS. (ETSI (2014-12 h:18,28)

As said above, the requirement of co-existence is for a transition period when operators are deploying the new virtual network functions. This will take place gradually, and may require a long period of time.

4.4 Other Industry Specific Requirements and Recommendations

In addition to ETSI NFV (ISG) requirements discussed above, there are also other industry specific requirements and recommendations that need to be considered when adapting system testing for NFV environment. These requirements and recommendations are discussed in this sub-section.

The Compliance to *3GPP Specifications* makes one specific requirement. The architectural changes in network functions virtualization will impact protocols and message flows across the network. These impacts may reflect to the compliance to 3GPP specifications. That is why compliance to specifications needs to be assured especially in multi-vendor eco-system. This may require exhaustive conformance testing. Assuring compliance to protocol specifications is nothing new in system testing, but in NFV the environment may be a multi-vendor environment, which means increasing needs for interoperability type of testing to secure compliance to 3GPP specifications. (Nair V. and Gupta V.K. 2014:13)

Network Testing makes other requirement area. There will also be significant changes in the network, when deploying NFV. That is why Nair V. and Gupta V.K. (2014) recommended that all the existing network services should be tested, to see that no Quality issues were introduced in the deployment. This means quite much of Network testing. The recommended tests are listed in the table below (Table 15.)

Table 15. Recommended network tests. Data gathered from Nair V. and Gupta V.K. 2014:13.

	Recommended Test
1	Integration testing of new features/services to assure smooth rollouts.
2	End-to-end testing of all the features/services in the end–to-end network with multi-vendor/multi technology environment.
3	Field trial to assure overall performance of new technology.
4	SDN controller security testing.
5	End-to-end network testing assuring that all legacy services are working fine and not impacted when introducing SDN/NFV

Table 15 above lists the recommended network level tests.1) Integration testing of all new features and services is important to avoid problems when deploying features. Considering the unpredictability (Spirent 2014:5) and complexity of the new virtualized environment, it become even more important that 2) the tests are done in network test environment having multi-vendor equipment. 3) The tests at the customer premises, field tests, will become more important; because of the differences in the environment, which may be even unique on some customers. 4) The SDN controller security testing is on the list, but it is not relevant when having only NFV solution. 5) The last item on the list is securing the legacy, which means testing the legacy functionality in the network test environment.

4.5 Software only Product

In virtualize environment telecom solutions are not necessary offered only as complete products based on the proprietary hardware and proprietary software, but also solutions as ‘software only’ that are possible to run on a commercial of the shelf (COTS) hardware. This makes *continuous deployments* possible and more attractive in telecommunication business. In development point of view this means even shorter development cycles than today. According to Martin Taylor (Tailor M. 2014:18) new software services are created, by deploying new software elements in the network, which will make it possible to bring new services to market *in weeks*. This means faster cycles in development, frequent releases, and continuous deployments. Testing should be able to rice this challenge in the future.

4.6 Summary

The network operators are aiming to solve multiple problems related to hardware-based appliances by introducing network functions virtualization, which allows them to use industry standard hardware. Besides all the benefits network functions virtualization brings also, for example, new failure model and more complexity to the environment as a side effect.

The complexity of the Network Functions Virtualization environment will bring new failure models, as well as, some level of unpredictability. This is because of the variety of different elements, software and hardware, from different vendors. The unpredictable environment will cause some new challenges, for example, dimensioning may be difficult, even impossible by reading specifications. This will impact also characteristics testing, which should be able to provide information about the dimensioning. The complexity of the environment affects also to system features, such as, load balancing, performance measurements, and capability of the system to diagnose and recover faults. All these mean new potential fault situations, or areas, in the network environment, which makes testing in network environment preferable. In addition to network level testing also the reliability of Network Functions Infrastructure (NFVI) need to be assured. This means, testing of different fault situations like hardware faults, software faults, and operator faults.

Although the complexity of the new environment will bring changes and challenges in testing like, for example, in troubleshooting, it will also bring some benefits. The new environment will bring the opportunity to organize test environment in a new ways. By virtualizing also the test tools, it makes it possible to load the test tools in the same cloud environment with the system under test. This leads to possibility to have ‘portable’ test environments, test environments-on-demand. This would enable even to run tests in a rented cloud outside the company.

ETSI NFV (ISG) has defined requirements and discussed about the possible threads introduced when deploying network functions virtualization. The requirements described in the ETSI NFV document: “Network Functions Virtualisation

(NFV); Virtualisation Requirements" (ETSI 2013-10 a) are listed in the table below (Table 16.):

Table 16. ETSI NFV Requirements. Data gathered from ETSI 2013-10 a.

Requirement
Partially or Fully Virtualized Network Functions
Portability
Performance
Resiliency
Elasticity
Security
Service Continuity
Service Assurance
Operational and Management
Co-existence with the existing networks –Transition

Table 16 above lists the Network Functions Virtualization (NFV) Requirements that will impact system testing. In addition to these requirements, there are also other issues that needs to be considered in testing like, for example, compliance to 3GPP specifications when new protocols are introduced.

When introducing new technologies such as Network Functions Virtualization (NFV), the test methodologies need also be considered. This covers knowledge and knowhow of the new technology, as well as, special testing and diagnostic skills to troubleshoot problems in this complex network environment, not forgetting the need of tools capable to all this.

Network Functions Virtualization will make it possible to sell software only products. This may lead to faster cycles in development and frequent releases, continuous deployment. This will be a future challenge for System testing.

5 Impacts and Proposed Changes in System Test Strategy

This section is the most important outcome of this study. It proposes changes in the Media Plane Development system test strategy. The changes proposed here are triggered, either by the requirements presented in the preceding section, or the factors presented in the beginning of this section. The section starts by presenting first, the external factors: *Integration with external products* and *BUCI end-to-end test strategy,* and second, it presents the internal factors, which are changes in the product architecture. After presenting these factors, the *Common Impacts on Most of the Testing* are discussed. Then there are two sub-sections that propose changes on system test scope. First, *Changes in Release Area Concept* and second, *Changes in Feature Integration*. This kind of division is natural since the scope of the system test is twofold. The legacy part of the system functions is covered by the release area concept and the new features are covered by the feature integration tests. The last sub-section will discuss about the changes in the test scope in lower level tests. To be noticed that the parts of the strategy that are not mentioned in this section are not expected to require any changes due to the virtualization.

5.1 External Factors

External factors here are activities and products outside the Media Plane organization that need to be considered in the local system test strategy. These are discussed in the following sub-sections.

5.1.1 Integration with External Products

Products that are as part of the Border Gateway Function (BGF) product but are not developed in Media Plane Development organization are considered as external products. In the native Border Gateway Function (BGF) such external product was system platform. Therefore the integrations with external products were limited in the integration of this platform. When moving to virtualized BGF product, the amount of external products will grow remarkable. There will even be products that are systems themselves consisting of several products, such as, Ericsson Cloud System with many internal applications. There will also be third party products that need to be integrated with BGF. These third party products

are, for example, hypervisor, Linux Operating System, etc. All this is new compared to the native BGF.

Proposed Changes in System Test Strategy

The BGF product development needs to be synchronized with external product deliveries. This means more planning in development, also in testing. It is about scheduling, but also thinking about the configurations of different versions that need to be covered in testing.

5.1.2 BUCI End-to-End Test Strategy

The Ericsson Business Unit, Core and IP (BUCI) end-to-end test strategy describes the tests that are performed on the network level for the complete IMS network. It describes the expectations per each development organization delivering products to this network. This BUCI en-to-end test strategy is renewed for virtualized network functions. The biggest change compared the previous strategy is a common continuous integration activity before delivering to the network level tests. A network level continuous multi-application integration will be set-up 1) to get fast feedback for the delivering development organizations and 2) to prevent unnecessary troubleshooting in network tests. The new continuous multi-application integration activity requires collaboration between development organizations.

For Media Plane Development organization this new continuous multi-application integration activity offers fast feedback about the BGF quality from the IMS network level tests. Additionally it provides also a possibility to have the latest version of the Session Border Gateway (SBG) application in the test network when running system tests of BGF.

Proposed Changes in System Test Strategy

To adapt local system testing to this multi-application continuous integration requires *collaboration* with the SBG development organization and *changes in the system test environment including the test tools*, which need to be aligned with the multi-application continuous integration test environment. This will cause changes in the ways of working. They cannot fore see yet, but will be recognized when adapting test practices in the multi-application continuous integration.

The multi-application continuous integration will require resources from the Media Plane Development organization. This is for planning, coordination, collaboration, and daily test activities, such as, trouble shooting and test support. Maybe also for test automation will require some work. If these resources are from system test area, or if they are separate resources, needs to be decided. It must be studied if the activities related to the multi-application continuous integration are possible to merge with the existing system test activities to form a more solid solution.

5.2 Internal Factors

This section discusses the product changes that will impact System Testing. The changes are introduced because of the new architecture of Media Resource System (MRS). The new architecture will require changes in the implementation of Border Gateway Function (BGF) product, which will then impact system testing. This section discusses about some of the changes on high level. These changes are interesting from test strategy point of view, but when defining test scope a proper test analysis is needed.

5.2.1 Network Functions Virtualization in MRS

The virtualized Media Resource System (MRS) structure follows the common NFV framework presented earlier in the Section 4.*1 Concept and Overview of Network Functions Virtualization (NFV)*. The new high-level distributed and integrated architecture of the virtualized MRS is shown in the figure below (Figure 15.)

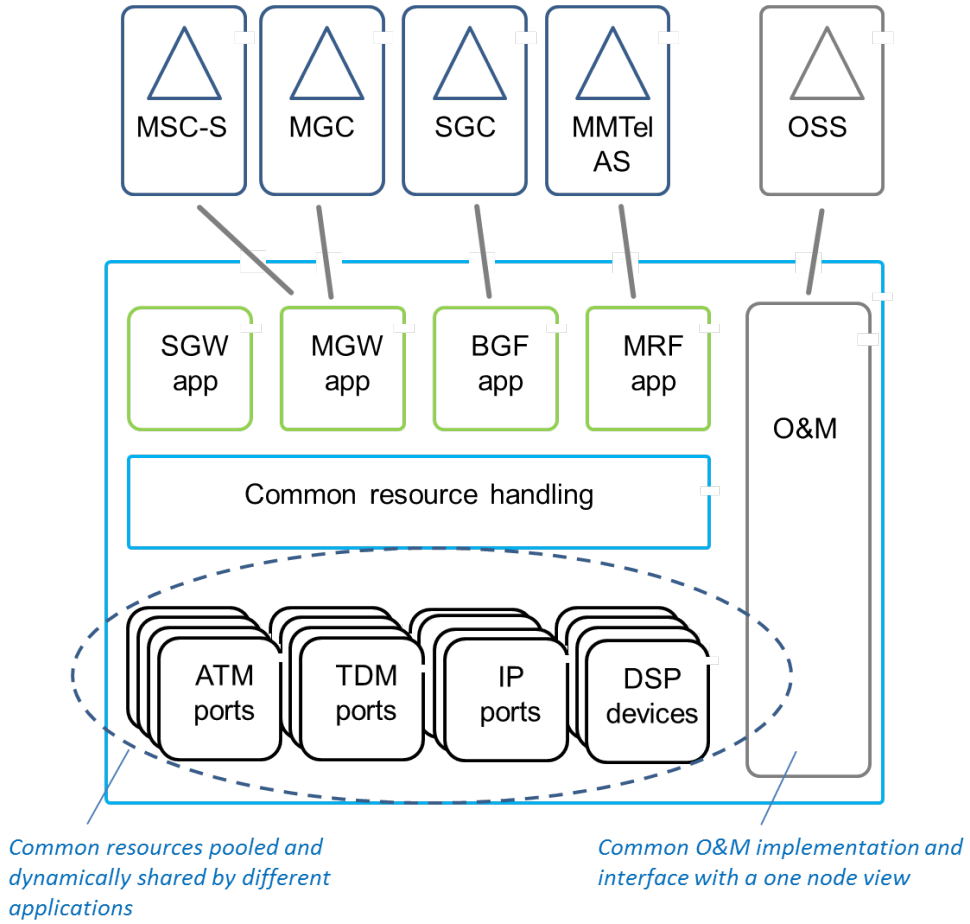

Figure 15. Architecture of Virtualized MRS. Reprinted from Lundström J. 2013:3.

Figure 15 above shows the structure of the virtualized MRS on high-level. In the Ericsson Cloud System the O&M part is the *Ericsson Cloud Manager*, which includes necessary functions required for handling both telecom and IT processes end-to-end. These functions may be, for example, applications to monitor network efficiency, service analytics or billing. Different signaling ports (ATM, TDM, IP) and Digital Signaling Processor (DSP) devices represent the common hardware resources in NFV infrastructure. (Lundström J. 2013:3).

The virtualized BGF (BGFv) as a part of Media Resource System (MRS) consists of HW, which may be Ericsson Blade System, or a third party hardware, OpenStack as a Virtual Infrastructure Manager (VIM) and KVM (Kernel Based Virtual Machine) as a hypervisor.

5.2.2 Ericsson Cloud System

The Ericsson Cloud System includes both the *Ericsson Cloud Execution Environment* (CEE) and the *Ericsson Cloud Manager*. Ericsson Cloud Execution Environment includes hardware virtualization for multiple applications sharing the same infrastructure. The execution environment supports coexistence and migration of the existing network functions. It is based on OpenStack cloud management and KVM (Kernel Based Virtual Machine) hypervisor. (Ericsson n.d. c).

The Ericsson Cloud Manager includes features like, *Self-Service Portals* providing on-demand control to the operator, *Orchestration* coordinating automated processes and manual tasks to provision services, *Configuration Management Database* consolidating network data of the virtual infrastructure at both the physical and logical levels. The Cloud Manager is able to manage both legacy (physical) and virtual infrastructure while supporting multiple hypervisor technologies and supports privacy, regulatory laws, and resiliency against cyber-attacks.
It provides also Metering to keep track of resource usage. The Ericsson Cloud Manager is available either as a standalone product or as an integral part of Ericsson Cloud Infrastructure. (Ericsson n.d.d)

A Virtual Machine (VM) consists of a *Guest Operating System (OS)* and its application. The VM executes on a *hypervisor* running on a physical compute host (blade). The scheduler in the cloud infrastructure determines the host where each VM is to execute on. A hypervisor is a program enabling to share a single hardware host between multiple virtual machines. Virtual machines have own host CPUs, own memory, and also other resources. These are controlled and allocated by hypervisor on a need basis, and makes sure that the VMs do not disrupt each other. Both the Guest Operating System and hypervisor may vary depending on the NFV environment. According to Ericsson (n.d. c) the Guest Operating System (OS) in Ericsson Cloud System (ECS) is Linux with Real-time capabilities, and the Hypervisor is Kernel Based Virtual Machine (KVM).

Proposed Changes in System Test Strategy

It is not possible to adapt general test strategy to variation of software products and different hardware in the environment, because they are unpredictable. Therefore it is important to make decisions of the test scope based on the test

analysis, when the software and hardware components used in/with the product are known. The analysis might be easier in case the products included in *Ericsson Cloud System (ECS)*, *Ericsson Cloud Execution Environment (CEE)* or *Ericsson Cloud Management System CMS)* are used.

5.3 Common on Most of the Testing

The changes in the environment brought by the Network Functions Virtualization are common for all testing. As it was discussed in the Section *4.2 Network Functions Virtualization Environment*, the complexity of the environment will grow. This means totally new failure models, which are dependent on how the system is build. In any case, it will be a multi-vendor environment with variation of different elements, such as, hypervisor, Operating System, and commercial of the self (COTS) servers. The customer environment will become unpredictable, which makes analyzing of test needs; especially test environment and tool needs difficult. Additionally there will be new types of faults in network environment *requiring more end-to-end testing and possibly new tools*.

The capability to diagnose and recover fault situations, reliability of the system, in unpredictable environment may become cumbersome. Solving the problems will be dependent on the ability of multi-vendor equipment understand each other's, being able to communicate faults, for example usage of different fault codes or fault messages. This will be c*hallenging, but also require new skills in troubleshooting*.

Test automation requirement is not only because of the virtualization, but virtualization will impact on the frequency of software deliveries. Therefore, it is increasingly important to automate as much as feasible.

Proposed Changes in System Test Strategy

Virtualization will bring architectural changes and many new components in the environment. These changes will impact testing in many ways. Test scope need to be adjusted, competence needs will change, and also test environment and test tools may require changes. These changes may be challenging, when considering the unpredictability of NFV environment. The unpredictability can be seen here as a variety of the different components in multi-vendor environment.

That is why the final impacts on system testing cannot be included in the strategy, but they need to be analyzed case by case, when making test analysis.

A thoroughly analysis of the product needs to be performed for the test scope, and test environment and test tools needs. Obviously, several different test network set-ups are needed to cover the most common customer network set-ups. It will be impossible to cover every possible variation of HW and SW combinations. Decision about the test network setups need to be done in close co-operation with Product Management. There will be many different set-ups, therefore there need to be a controlled way to maintain information of the HW and SW and their versions. *Traceability between run tests or characteristics measurements and test environment used is needed*.

Automation must be implemented such a way that the automated cases can be utilized in continuous integration (possibly also in multi-application continuous integration). Additionally there is a need for *new competencies related to the test environment, tools and troubleshooting.* This need to be taken care well before much more people will move to work with virtualized products.

5.4 Impacts and Changes in the Release Area Concept

The release area concept in the Media Plane Development has been experienced effective and it should be continued. The following sections will describe the issues per each release area that need to be considered when adapting them for Network Functions Virtualization.

5.4.1 Release Area Upgrade and Expansions

The present upgrade and expansions release area is responsible for securing that the Media Resource System (MRS) node is upgradeable and expansion is possible between all software tracks, i.e. testing of different upgrade paths between the existing software tracks.

In the virtualized product the upgrade will have big changes. There will not be upgrade paths in the ways there were for the native BGF. The expansion in the native node meant hardware expansion. This is no longer needed in the virtual-

ized BGF testing. Instead of the expansion testing, there is a need to test *manual scalability*. There are two types of scalability for virtualized network functions, automatic and manual. The automatic scalability is related to load regulation and failure situations whereas the operator handles the manual scalability. Additionally also *portability* requirements need to be covered in this area.

Proposed Changes in System Test Strategy

The scope of the Upgrade and Expansions release area should be changed to cover upgrades and manual scalability. That is why the new release area could be named as '*Upgrade and Manual Scaling'* release area, or something that better describes the new scope.

Upgrade testing should verify the management of images life cycle. This is *Portability* that needs to be covered by creating, moving and terminating virtual machines (VM). The manual scalability testing should verify that the NFV framework is capable of scaling VNFs (scale up and scale down) and moving its components from one computing resource to another.

5.4.2 Release Area Operations and Maintenance

The present Operations and Maintenance release area is responsible for securing that the quality of non-functional requirements related to operation and maintenance is acceptable for all product packages.

The whole Operations and Maintenance (called Management and Orchestration (MANO) in NFV) part of the product will change. NFV Management and Orchestration will handle NFV Infrastructure (NFVI), Virtual Network Functions (VNF) and Network Services (NS). This includes the fault and performance management, Policy management and testing of Network Services. In addition to that, MANO should also be able to handle native and virtualized applications in the same network. That is, it should be able to interwork with the existing OSS/BSS.

O&M is mainly used for manual actions, managing Virtual Machines, creating and terminating them, but there are also automated O&M mechanisms. These mechanisms, such as creation, scaling and healing of VNFs based on pre-defined criteria, did not exist in the native system. There are also requirements for Service

Assurance in O&M, which brings new monitoring functions in-to the system that are remotely accessible, and can perform diagnosis.

Proposed Changes in System Test Strategy

The Operations and maintenance area is one of the most impacted release areas. That is because of the huge changes in the architecture of the O&M. The name of this area is proposed to be changed to '*Management and Orchestration'*. That is the term used for operations and management in relation to Network Functions Virtualization.

The functionality of all the functional blocks in NFV Management and Orchestrator (MANO) needs to be covered in tests. That is, NFV Orchestrator (NFVO), VNF Management (VNFM) and Virtualized Infrastructure Manager (VIM). These may be from different vendors and require a different approach. For example, there are some known issues in OpenStack, in case it is used as VIM (more information about the OpenTack can be found in Appendix 4.). In addition to these also NFV profile handling (i.e. OVF repository), interworking with Operational Support System (OSS) need to be covered.

Management and Orchestration should also be able to operate and manage a hybrid network. That is, to operate and manage a network where NFV framework co-exists with the legacy network. Therefore, this kind of co-existence tests need to be covered as transition phase tests in Management and Orchestration release area.

Additionally this release area should also verify the capability of the NFV framework to automatically create, scale and heal of VNFs based on pre-defined criteria, and also that Network functions are remotely accessible, monitored, and can perform diagnosis. In other words, this area should test Service Assurance in operations and management.

5.4.3 Release Area Signaling

The present signaling release area is defined to secure that different signaling configuration of the node function, including the different signaling standards TTC, ANSI, CWTS and ETSI.

In the all IP network this is no longer valid. There is no need to test different signaling standards.

Proposed Changes in System Test Strategy

This area can be removed. It is not needed.

5.4.4 Release Area Single Traffic & Features

The present Single Traffic and Features release area is responsible for securing that single call cases used by the customer are working.

The virtualization will require more *interoperability and integration type of testing* to verify that the NFV framework is capable to host, optimize, and load Virtualized network functions (VNF) in a standardized multivendor environment.

Additionally the transition from native, non-virtualized, to virtualized networks will require *co-existence* and transition testing, which is testing that the NFV framework co-exists with the legacy network and supports transition phase. This will also impact the test environments requiring a Hybrid network.

Proposed Changes in System Test Strategy

Interoperability and Integration testing are needed to verify that the NFV framework is capable to host, optimize, and load Virtualized network functions (VNF) in a standardized multivendor environment.

Co-existence and transition testing should verify that the NFV framework co-exists with the legacy network and supports transition phase. This is a so called hybrid network.

The test scope needs to be balanced between this release area and Network level Feature Integration. For example the co-existence with legacy network could be part of Network level Feature integration, as well as, most of the interoperability cases. In some cases, for example for performance, there will be a need to compare between native and virtualized BGF. Whether this is done in the Single Traffic & Features release area or in the Network level Feature Integration, needs to be agreed. Also the test environment needs to be prepared for this kind of testing purposes. It should be also noticed that the multi-application continuous integration setup should enable usage of the latest Session Border Gateway (SGB) when testing single traffic cases.

5.4.5 Release Area Media Quality

The present Media Quality release area is defined to secure media plane quality of the Media Resource System (MRS) node.

The test scope in this area needs not to be changed since there are no new area specific requirements. The Media Plane Quality is expected to be on the same level as today with the native product. But when thinking about the NFV environment, it seems that this area will have big challenges. The Network Functions Virtualization environment will bring many new layers in the system, with also unwanted phenomenon such as, latency, packet loss, etc.

Proposed Changes in System Test Strategy

To keep the present Media Plane quality makes this area very important from the testing point of view. The key here is that testing needs to be able to give fast feedback about the Media Plane Quality to enable corrective actions on time.

The test environment requires consideration. What are the different configurations that are needed to cover? Additionally the metrics and how the system provides them will change. This means that the counters and the collection of the data need to be re-organized including tools used for testing.

5.4.6 Release Area Stability

The present Stability release area is responsible for securing that the MRS node operates stabile according to requirements in all software tracks.

As it was with media quality release area, the quality is expected to be on the same level as it has been before virtualization. There are no new requirements on stability area as such, but there is a requirement for automatic scaling which needs to be covered also in stability tests. It is about scaling the resources up and down based on the traffic load intensity.

Proposed Changes in System Test Strategy

As it was in the case of the media quality release area, this area might also be challenging because of the complex Network Functions Virtualization environment. The changes in the scope are related to automatic scaling. Both scaling up and scaling down need to be included in the stability tests. The functionality of scaling criteria must be covered in the stability tests. This release area may require some extra effort in the area of test automation to be able to provide fast feedback enabling corrective actions on time.

The metrics and how the system provides them will change. This means that the counters and the collection of the data need to be re-organized including tools used for testing.

5.4.7 Release Area Robustness

The present Robustness release area is responsible for securing that the quality of non-functional requirements related to robustness is acceptable for all product packages.

The NFV Resiliency requirements may be mapped very well to this release area. *Resiliency requirements* are related to the system's capability to recover after failure situations. One of such requirement is the service continuity requirement (Section 4.3.3). It requires that the NFV framework must be able to restore services (i.e. to recover VMs, provide alternative solution). Additionally also part of the Scalability requirements may be mapped in the robustness release area. That

is, Scalability, also called as Elasticity, in cloud in a failure situation, e.g. in HW failure.

The Resiliency requirements focus on the availability of the products and the User-perceived dependability because; a) unreliable services will most likely to be discarded by users, and b) the cost of system failures can be high. Potential causes of failures may be related to Hardware, e.g. the commercial of the self (COTS) server may be a root cause of failure. Failures may also be related to software at various levels, such as host Operating System (OS), hypervisor, Virtual Machine (VM), or the Virtual Network Function (VNF) instance itself. Additionally failures may also be caused by the operators, e.g. mistaken operations and configuration, or capacity planning, Virtual Machine (VM) deployment and migration. There are a number of different areas and elements where things may go wrong, and still the system should survive, or at least these unwanted occasions may not lead any uncontrolled situations, and the system should recover in reasonable time.

Proposed Changes in System Test Strategy

The scope of the release area Robustness does not need to be changed a lot; it should still cover user experiences of service continuity and the system's ability to recover error situations. Anyhow, to make it easier to communicate about tests in this release area with the customers, it is proposed that the name of the area is changed to Release Area Resiliency.

Resiliency testing should verify that Network functions are capable of recovering after failure and the NFV framework is able to classify Network functions according to resiliency and facilitate the resiliency scheme in both the control plane and user plane. As part of resiliency tests *Service continuity* testing should verify, for example, that in the hardware failure or a resource shortage/outage the NFV framework is able to restore impacted VNF instance. Additionally also *testing of Scalability, and Elasticity in cloud* in a failure situation, (e.g. in HW failure situations) need to be covered in this release area.

As it is with all other release areas, also in Resiliency Release Area, Test Automation must be favored as much as feasible. To be able to automate failure situations in Network Functions Virtualization environment might require new ap-

proach. One possible approach could be to use a so called fault injection method, or as Haryadi S. et al. (2011) called it, failure as a service, Faas. Haryadi S. et al (2011) states that every cloud service that promises fault-tolerance must ensure also recovery, and by using Faas, development could focus more on feature development and less on recovery testing. Some examples about the fault injection can be found in the Internet. For example, the IEEE: Network Function Virtualization: Challenges and Directions for Reliability Assurance (Cotroneo D. et al 2014:40) lists different types of resiliency tests done by using fault injection. These are a) Fault Injection Testing of Virtual machines random termination of virtual machines, b) Fault Injection Testing of Cloud Management Software and Fault Injection Testing of Hypervisors. The fault injection is discussed further in Section 9. *Opportunities for Improvements*.

Resiliency area testing also assures that the fault management works properly. This will be challenging in a multi-vendor environment. Network problem diagnostics, monitoring performance, actions taken for fault detection and remediation, and interworking with Management and Orchestration are all dependent on the implementation of the equipment in the multi-vendor environment. In many cases the results of the robustness tests may not be predicted forehand, which makes automation of the cases difficult. This is not new in this release area, but it will become even more difficult when the environment is unpredictable too. Some issues related to Virtual Infrastructure Manager OpenStack are presented in Appendix 4 *Issues in OpenStack.* This information can be utilized when developing Resilience test cases.

5.4.8 Release Area Characteristics & In Service Performance

The present Characteristics and In Service Performance (ISP) is responsible for securing that all characteristics requirements are met in all software tracks.

From the performance point of view everything is expected to work as today with the native product. In practice the NFV brings many changes in the characteristics area. The biggest issue there is the configuration used for testing. When today there are two different hardware configurations for native MRS, the number of different hardware configurations will explode. This means a change in ways of

working. How to select the reference configurations, will there be any. What are the measures needed?

Another issue is scalability. There will be cluster internal scalability, i.e. scaling the ratio of the number of System Controller (SC) and Payload (PL) blades within the cluster. There will be also another internal scalability in a sense of the size of Virtual Machine (VM) (i.e. number of cores allocated per VM). There will also be scalability in a sense of allocating more blades to fulfill growing resource needs. How will all this be taken care of in characteristics measurements?

Additionally the way of measuring characteristics will change. NFV will introduce new Service Quality Metrics related to 1) *Virtual Machine Service Metrics*, 2) *Virtual Network Interface Service Quality Metrics* and 3) *Orchestration Service Quality Metrics*. These are the metrics that operators may use to measure Service Level Agreement (SLA). (ETSI 2014-12 d:1-27).

Proposed Changes in System Test Strategy

This area requires deeper analyzing per each network configuration.
Performance testing should verify that the NFV framework is independent of HW used and framework is capable to collect performance related information.

The metrics and how the system provides them will change. This means that the counters and the collection of the data need to be re-organized including tools used for testing.

When making characteristics measurements it is important to be able to tie the results and the configuration used to each other's. In the Network Functions Virtualization environment there will be more variation in the configurations. Test tools and test scripts/programs, are part of the configurations. This means that the test tools and test script/program lifecycles must also be managed in professional manner. That is, test tooling need to be handled as products.

Furthermore, the input for dimension must change. In the new Network Functions Virtualization environment it is almost impossible to dimension the system based on the specifications, but need to be measured from the system (Daitan Group 2014). So, the measurement information for dimensioning purposes needs to be

feed to the characteristics team in system development. This should be done fast and frequently, and may require visualization, to be easier to follow-up. This will require automation, keeping in mind that also in this area it is important to provide fast feedback enabling corrective actions on time.

5.4.9 Release Area Vulnerability

The present Vulnerability release area is responsible for security testing on all software tracks. This has covered both control plane and user plane.

In NFV environment the NFV framework protects the network from end-to-end vulnerabilities (new HW interfaces and third party entities) and provides authentication, authorization, data encryption, data confidentiality and data integrity. This will also cover a separation of Tenant and Infrastructure Domains in Orchestration and Management.

The documents presented in Appendix 3 are to be used as an input for vulnerability analysis.

Proposed Changes in System Test Strategy

Release area Vulnerability should verify tend-to-end vulnerabilities, authentication, authorization, data encryption, data confidentiality and data integrity.

The new vulnerabilities introduced by the new layers and components need to be considered, for example, in the virtualization layer there are shared storage resources, network resources and new interfaces e.g. new hardware resources or management systems, and access to NFV functions via APIs. In addition to that the Management and Orchestration system is completely new. Both the control plane and user plane need to be covered as in the native system.

Comprehensive vulnerability analysis is recommended to define the new test scope.

5.4.10 Not Mapped Impacts

There are two ETSI NFV requirements that were not mapped directly in any of the release areas. First, A requirement of *partially or fully virtualized network functions* (Section 4.3.1.). Second, A *requirement for portability (*Appendix 1. Portability of VNFs). The portability requirement makes it possible to load, move and execute VNF's across different multivendor Network Point of Presence (N-PoP) environment.

Proposed Changes in System Test Strategy

These requirements need to be considered when defining the test scope of the release areas. The *partially or fully virtualized* network may be covered by building a test network that will make it possible, for example, to run test cases where the control plane is not virtualized, but the user plane is. That is, cases with virtualized BGF and native SBG. The *portability* requirement may be covered in continuous integration (CI) where the package (image) is created, loaded in the cloud environment and tested. In the CI automation the selection of the different cloud is made based on the availability. The CI machinery will load a new image in a cloud that is not reserved for any other testing at the time, and performs tests in there. Portability aspects are also covered in upgrade testing.

5.5 Impacts and Changes in Network Level Feature Integration

This section is about the network level integration of new features. As it was said in the beginning of Section 5, the new features in the system testing are covered in feature integration, whereas the release areas cover the non-functional system requirements.

All the changes in the environment brought by the Network Functions Virtualization will impact feature integration especially on the network level, also on end-to-end network. These tests are related to the integrity of the Media Resource System (MRS) product. These are, for example, network level interoperability tests between product releases.

There are new elements in the network environment that need to be considered in feature integration, such as, multi–vendor environment, or the Management

and Orchestration, which replaces the existing Operations and Maintenance system. Both the new Management and Orchestration and multi-vendor environment will require more end-to-end type of integration, including multi-vendor integration and interoperability testing. Additionally, it is as important with the virtualized product, as it was with the native product, to perform extra feature integration and interoperability tests in the case of changes in the protocols used like, for example, 3GPP protocols.

Feature integration is an area where also the comparison of the functionality of the new features in the native and virtualized environment is needed. This is valid as long as the development of the native product will be continued. Additionally also tests in the hybrid network are needed. This is needed for a transition period when operators may have both native and virtualized products in the same network.

Proposed Changes in System Test Strategy

The scope of the network level feature integration needs to be adapted on the selected configuration(s) based on the more detailed test analysis. In general the number of tests is expected to increase, because there is a need to focus more on inter-operability testing, multi-vendor integration, end-to-end type of integration and also testing in hybrid network. All this will require more testing in the network environment. Part of the solution for this would be to start integration on the network test environment already in the cross-functional teams. More about this possibility is discussed in the next sub-section.

Network level integration will require a test environment where it is possible to perform feature integration both for virtualized and native product, and also run these tests in the hybrid network. The environment should support variation of different multi-vendor configurations.

5.6 Impacts on Test Scope on Lower Level Tests

This section discusses the impacts and needs for changes in node level function testing, because of the changes made in system test.

Due to the pressure to add more system level integration testing in the network environment, there is a need to find a balance between the system level integration in different activities. Today the independent test teams perform the feature integration testing on network and end-to-end test environments. When adding more testing, it might be difficult resource wise. Therefore it could be reasonable to think about different options to increase testing in the network environment, even in an end-to-end environment.

Proposed Changes in System Test Strategy

An increased amount of testing in network environment could be divided between the different testing activities. One solution would be to start integration in the network environment already in cross-functional teams in functional tests. In practice this could mean, for example, adding a real SBG node in the function test environment used by cross-functional teams. Using the real SBG nodes would change in the simulated interface. That is, simulating the session Initiation Protocol (SIP) towards SBG instead of simulating Gateway Control Protocol (GCP) towards BGF.

This might be a competence issue, since the test tools and the network test environment are different than those used in node level functional tests. To build up the required new competencies in the cross-functional teams will require resource planning, and perhaps also new ways of working, such as, using the Specification by Example method, as described in Section 9.1, or maybe using DevOps teams, as described in Section 9.1.2.

6 Impacts and Proposed Changes on Ways of Working

This section discusses the impacts and changes needed in the ways of working. That is, changes in the processes, methods and human resources, new skills and competences. Additionally the changes proposed for the test environment are also discussed in this section.

6.1 Characteristics Measurements

Estimation of characteristics, the capability of the virtualized BGF will become difficult, even impossible, based on the specifications. That is because of the variety of different equipment from different vendors.

Proposed Changes in System Test Strategy

The only way to get a reasonable estimation of the capability of different configurations is to measure. This means a change in the ways of working in the characteristics release area. When working with virtualized BGF, the characteristics release area needs to be able to provide fast feedback information of the characteristics measurements to the characteristics team in system development. This requires a change also in the tooling. There is a need for a tool that would make fast characteristics feedback possible, even on real time.

6.2 Test Automation

The test automation rate is rather high in the native BGF testing, but there is still room for improvement. In the native BGF there are tests that have not been automated because they have not been possible or feasible to automate, for example, cases that require unplugging cables, or other type of manual intervention.

Proposed Changes in System Test Strategy

Virtualization will change the test environment. This may bring also new opportunities for automation. For example, part of the manual robustness test cases could be automated by using fault injection (Section 9.4 *Fault Injection*). This needs to be studied further, since a sufficient coverage of the System tests needs to be included in the continuous integration machinery.

The frequency for regression tests is becoming higher and higher all the time. Test automation is partially a tool issue, but also a ways of working issue. Finding new ways for automation is crucial to keep the pace in system testing.

6.3 Multi-Application Continuous integration

The Ericsson Business Unit, Core and IP (BUCI) end-to-end test strategy will introduce a new multi-application continuous integration (CI) activity to provide fast feedback for the delivering development organizations and to prevent unnecessary troubleshooting in network tests. To participate in this kind of external activity will require actions also in the Media Plane Development organization. In addition to the fast IMS network level feedback about the quality of BFG, multi-application CI will provide possibility to get the latest Session Border Gateway (SBG) software in the end-to-end system test environment in the Media Plane Development organization. This possibility needs to be considered when planning tests and also tooling.

Proposed Changes in System Test Strategy

Multi-application CI will require resources from the Media Plane Development organization for support and also to align test environment, test tools and ways of working in to it. This means preparation work, changes in daily work, and perhaps also new competences. This will be clearer after the first trials of multi-application CI.

6.4 Competence and Skills

The Network Functions Virtualization will bring requirements for new competencies needed in system testing. The competencies are related to the new system architecture and new system components. These are, for example, Management and Orchestration (MANO), Kernel Based Virtual Machine (KVM), virtual switch, Hypervisor, Virtual Infrastructure Manager VIM) e.g. OpenStack and architecture in general including API's, test tools, test use cases, end-to-end/network test environment(s). In addition for people working in system testing, the new competencies may be also necessary for people working in cross-functional teams, especially if these teams will perform part of the end-to-end level system integra-

tion. Furthermore, the new architecture and components will bring changes in the needed testing, diagnostics and troubleshooting skills.

Proposed Changes in System Test Strategy

Building up the new competencies should start from the architecture and components area, and then continue in testing and diagnostic skills. These skills need to cover also the test tools. After this, people should be able to trouble shoot problems, localize fault, trace system, including the tools used. One solution to share and build-up competence about testing in end-to-end network environment would be using Acceptance Test Driven Development method, or as it is also called, Specification by Example. Another solution goes even further, and that is Development and Operations, DevOps. These concepts are discussed in Section 9. *Opportunities for Improvements*.

6.5 Test Environment

This section discusses the impacts and changes in system test environment for testing virtualized MRS product, starting with the test network, and continuing with system test tools.

6.5.1 Test Network

The system test network will be the most impacted area in the whole system test context.

Proposed Changes in System Test Strategy

The existing test network can be expanded with virtualized Session Gateway Controllers (SGCv) and Virtualized Border Gateway Function (BGFv). The new virtualized components are all located in a cloud, which needs to be managed and orchestrated by a cloud manager. The logical configuration of the virtualized Border Gateway Function (BGFv) system test environment is illustrated in the figure below (Figure 16):

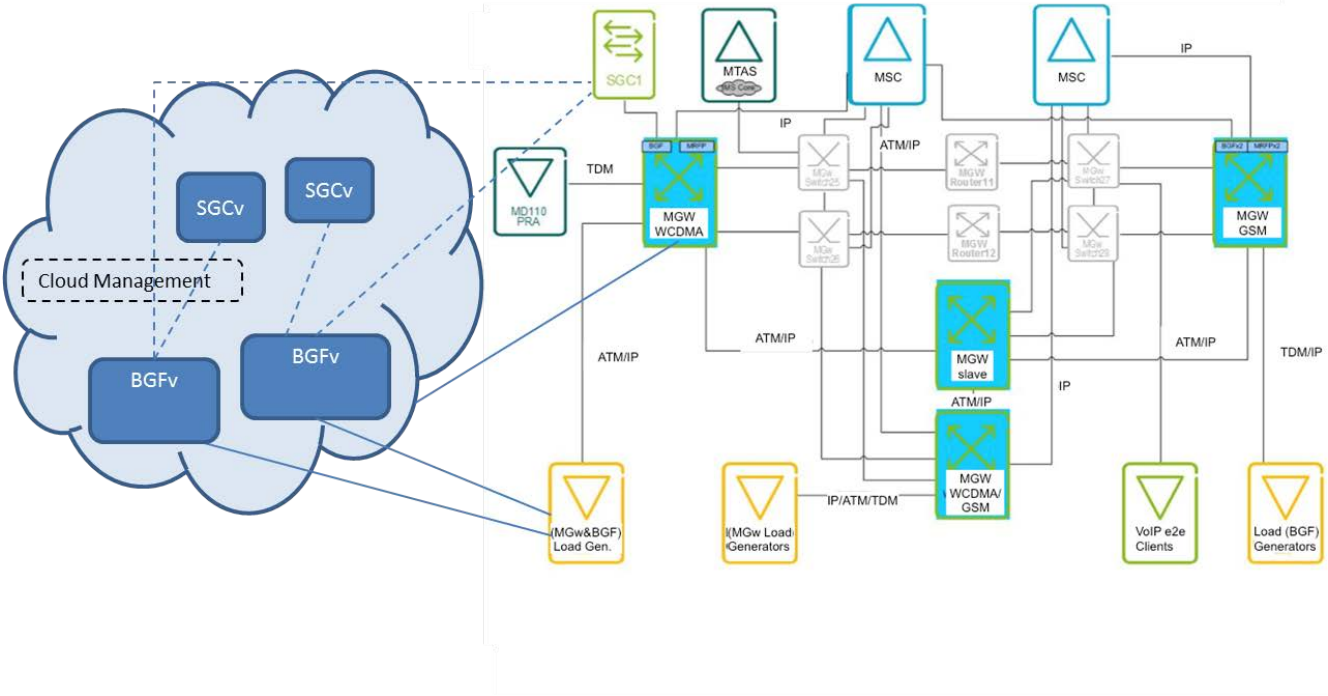

Figure 16. System Test environment for BGFv

Figure 16 above shows how the new test environment may be built as an expansion to the current test system. The change might be clearer when comparing Figure 16 above to Figure 9 *The current BGF system test network environment* in Section 3.3.1.

The virtualized Border Gateway Function (BGFv) should have connections from both virtualized Session Gateway Controller (SGCv) and native Session Gateway Controller. That is to test partially virtualized network functions cases where the media plane is virtualized, but not the control plane. There are also native and virtualized BGF nodes in the network to enable testing in hybrid network, where NFV framework co-exists with the legacy network. This kind of environment is needed, for cases such management and orchestration and feature integration testing. Native nodes need to be tested also in the future because the native BGF will exist in the product portfolio. Native nodes are also needed for comparison of for example performance of the native and virtualized nodes in Characteristics tests.

Reference Configuration

It should not matter what the cloud platform used in the environment is, but there is a need to have a reference configuration, or configurations of the test environment, which are based on one (or more) cloud platform. This is needed to be able to compare measurements and test results between the releases.

6.5.2 Test Tools

The existing load generators can be used also to test virtualized BGF. However, to be able to adapt the multi-application continuous integration also Session Initiation Protocol (SIP) interface towards SBG needs to be loaded. To accelerate the speed of new deliveries, also system tests need more automation. Today there is still a mass of manual tests or semi-manual tests in the system test area. To get rid of these manual or semi-manual tests is a goal for the future.

When it comes to the Network functions virtualization environment there will be more variation in the configurations. Additionally, test tools, test scripts and programs with their versions are part of these configurations.

Proposed Changes in System Test Strategy

The test automation rate must be increased. One option to implement automation for the existing manual or semi-automated test cases could be fault-injection, which is discussed in Section 9.4 *Fault Injection*. Another thing, common for all test tools, is that they should be handled as a part of the NFV environment configurations. This means that the test tools and test script/program lifecycles must also be managed in a professional manner. That is, the test tools need to be handled as products.

Furthermore, alignment to multi-application Continuous Integration has to be done. That will mean change in the simulation interface. Now the Gateway Control Protocol (GCP) H.248 between SBG and BGF is simulated. In the Multi-Application CI environment there is a real SBG. Therefore instead of simulating H.248, the incoming SIP towards SBG needs to be simulated.

6.6 Summary

The extent of the changes needed varies a great deal between the different testing areas. Some areas need more testing, some areas need only adjustments in the test scope. One of the biggest changes related to system test scope is in the Upgrade and Expansions release area, because of no upgrade paths are needed, and there are no expansions in the virtualized BGF. The manual scaling tests can replace the expansion test. In general the new Network Functions Virtualization environment triggers the biggest changes. From the environment point of view the biggest changes are in the *Operations and Maintenance* since the whole system will change. This area will need a lot of testing for the first releases. The amount of testing may then later be decreased. The Network Functions Virtualization environment will impact also the *Media Plane Quality* and *Characteristics and ISP* areas. The amount of testing in those areas is dependent on the decisions as to how many different configurations are needed to be covered in testing. The new environment will also bring new vulnerabilities, which need to be covered in the *Vulnerability area*. *Robustness* is an area that will be impacted by the new NFV environment. The amount of testing is not expected to increase, but the scope and the test automation need big changes. The scope of the *Stability* tests needs to be changed to cover automatic scalability, scaling up and down. *Single Traffic & Features* and *Feature Integration* areas both need more testing in network and end-to-end environments. Depending on how they will be carried out, they will affect the extent of the adaptation needs. There might be possibilities even to merge these two areas. Additionally one option would be to start network level integration already in functional tests in cross-functional teams by changing the test setup to include also SBG. The *Signaling* release area may be removed because of the All-IP network. A summary of the adaptation needs is presented in the table below (Table 17.):

Table 17. The extent of the adaption needs in the system testing

Area in System Test		Test Scope Adaption	Number of Tests
1	Upgrade and Expansions	Medium	Decrease
2	Operation & Maintenance	Large	Increase
3	Signaling	Removed	Removed
4	Single Traffic & Features	Large	Increase

5	Media Quality	Medium	Equal
6	Stability	Medium	Increase
7	Robustness	Large	Equal
8.	Characteristics & ISP	Large	Equal
9.	Vulnerability	Large	Increase
10.	Feature Integration	Medium	Increase

Table 17 above presents the extent of the adaption needs in the system testing in the Media Plane Development organization. It simply lists the extent of the adaption needs on the test scope and are the test expected to increase, decrease, or stay equal compared to the present.

No new areas were found, but renaming of three areas was proposed. First, a new name *Management and Orchestration release area* for the *Operations and Maintenance* release area was proposed*. Second, a* new name '*Upgrade and Manual Scaling release area'* was proposed for the current Upgrade and Expansions area. Third, the present Robustness release area was proposed to name as *Resiliency release area.* The purpose of these new names was to get better match with the scope of the area after adaption.

There are two big changes in *ways of working*. First, the characteristics estimations need to be changed to include more measurements. Second, the Multi-application continuous integration requires changes in the test environment and test tools, maybe also in the implementation of test automation. Test network should contain cloud where SBG and BGF are installed. Using real SBG will require changes in simulation interfaces. Depending on the case, SIP interface towards SBG may need to be simulated instead of GCP interface towards BGF. The existing load generators may still be used. Multi-application continuous integration also requires alignment of test automation, test environment and test tools. This should include the ways how the automated cases are build and what are the interfaces used.

There is a need to automate more non-functional tests in general. In *Robustness* release area (resiliency) one option to increase Test Automation could be fault injection. This is an area proposed to be studied further.

All this will require new competencies from people. Competencies as for the new architecture, new components, new terminology, fault localization, troubleshooting, and test tools are all subjects for learning.

7 Proposed Changes and Action Plan

The proposed changes with the actions for the deployment are collected in the table below (Table 18).

Table 18. Proposed Changes and Action Plan

Subject to Change	Change	Magnitude	Actions for Deployment of the changes.
Release Areas			
Upgrade and Expansions	Upgrade paths and expansions do not exists in BGFv.	**Medium** The test scope adaptation requires Implementing new test cases. The number of Test cases is expected to decrease in total.	Remove upgrade paths and expansions from the test scope of virtualized product.
	New requirements for Manual scalability & portability.		Include Manual scalability and portability (VM lifecycle management).
	Renaming of the area.		Rename to: '*Upgrade and Manual Scaling'*
O&M	New architecture for O&M	**Large** The test scope will change a lot. The number of test cases is expected to increase. The Test analysis and implementing new test cases will require much resources and time.	Cover all the functional blocks in NFV Management and Orchestrator (MANO) in O&M tests.
	Transition requirement.		To be considered: Covering of Management of Hybrid Network in tests.
	New automated O&M mechanisms.		Verify the NFV framework's capability to automatically create, scale and heal of VNFs based on pre-defined criteria.
	New Service Assurance requirement for O&M.		Test that Network functions are remotely accessible, monitored, and can perform diagnosis.
	Renaming of the area.		Rename to: '*Management and Orchestration'*

Signaling	Virtualization will not call for any changes in this area, and after moving to all IP networks testing of different signaling standards is not needed.	**Removed Area**	Remove this area and all the tests in it.
Single Traffic & Features	Multi-vendor NFV environment	**Large** The new test scope needs to be set. The number of test cases is expected to increase.	Analyze new needs for Interoperability and Integration testing
	Transition requirement.		To cover co-existence and transition testing with legacy network
	Increased need for network level integration.		Balance the test scope between this release area and Network level Feature Integration.
Media Quality	Multi-vendor NFV environment	**Medium** The test scope needs not to be changed. And the number of test case is expected to be equal compared to the native BGF. Test tools need to be analyzed more deeply. They may require changes.	Analyze changes needed in test environment
	New counters and ways to collect counter data.		Analyze the needs to re-organize counters and the collection of the counter data including tools used.
	Both above.		Analyze possibilities and needs in test automation and implement changes.
Stability	New scaling functionality introduced in NFV.	**Medium** The ne test scope needs to be set. The number of test ceases is expected to increase.	Include both scaling up and scaling down in the stability tests.
			Cover the functionality of scaling criteria the stability tests.
			Changes in Test automation

Robustness	Multi-vendor NFV environment	**Large** The new test scope needs to be set. The number of test ceases is expected to be equal compared to the native BGF. The Test analysis and implementing new test cases will require much resources and time.	Include *Service continuity* testing (e.g. HW failure or a resource shortage/outage or *Scalability, and Elasticity in cloud* in a failure situation
			Include testing of functionality of Fault management in multi-vendor environment.
	Renaming of the area.		Rename to: '*Resiliency*'
Characteristics & ISP	Multi-vendor NFV environment	**Large** The test scope needs to be set. The number of test ceases is expected to be equal compared to the native BGF. The analysis of tool needs and setting up such environment that is capable to provide input for dimensioning even on real time, requires resources and time. (This is common activity for ways of working area.)	Verify that the NFV framework is independent of HW and framework is capable to collect performance related information.
			Tie the measurements and the configuration (including tools) used to each other's.
	Unpredictable environment. It is impossible to make estimations for dimensioning by reading specifications.		Provide Measurements as input for dimensioning.
Vulnerability	Multi-vendor NFV environment.	**Large** Vulnerability analysis is needed. The test scope may be adjusted based on it. The number of test cases is expected to increase.	Perform comprehensive vulnerability analysis Due to big changes in the environment to define the new test scope.

Feature Integration			
Feature Integration	Multi-vendor NFV environment.	**Medium** The new test scope needs to be set. The number of test cases is expected to increase.	Analyze the need for Interoperability testing, multi-vendor integration, end-to-end type of integration and also testing in hybrid network. Balance Feature Integration tests with functional tests performed by cross-functional teams and Single Traffic & Features Release area.
Way of working			
Characteristics Measurements	Unpredictable environment. It is impossible to make estimations for dimensioning by reading specifications.	**Medium** This requires implementing a new working process including Tool that is capable to provide input for dimensioning even on real time. (This is common activity with Characteristics & ISP area.)	Adapt to need for Fast feedback information of the characteristics measurements for dimensioning. It will require change in tooling and ways of working. Fast characteristics feedback should be provided even on real time.
Test Automation	More automated test cases are needed in non-functional tests.	**Medium** This requires analysis and study of new tools, including piloting and trials.	Study possibilities to use Fault Injection in Resiliency tests. Try to find and analyze also other possible ways to automate more non-functional tests.
Multi-application CI	Multi-application CI needs resources for support and also to align test environment, test tools and ways of working in to it. This is, resources for preparation work.	**Medium** This requires resource planning for both human and machine. The actual support and the amount of work needed for aligning tools, test environment and work processes will be discovered later when few trials with Multi-application CI are done. Incl. Study of merging with system test activities.	Prepare to allocate resources from Media Plane Development organization. To support CI activities To align test environment, test tools and ways of working in to it. To study if it is possible to merge Multi-application CI activities to present system test activities.

Competence and Skills	New competence needs related to Network Functions Virtualization are needed.	**Large** This requires proper competence planning, including learning objects and the learning order. All the actions in this area (i.e. People) are related directly or indirectly in competence management.	Analyze the competence needs to build up the new competencies.
	The competencies related to the new system architecture and new system components.		Prepare competence build up in the area of Architecture and components.
	New test tools, test use cases, end-to-end/network test environment(s).		Prepare competence build up in the area of Testing tools and environment.
	New testing and diagnostics, troubleshooting skills are required.		Prepare competence build up in the area of Testing and diagnostic skills.
	In addition for people working in system testing the new Competencies may be necessary also for people working in cross-functional teams, especially if they will perform part of the network level feature integration tests.		Try new ways for learning need to study: a) Specification by Example b) Development and Operations, DevOps. (See Section 9. *Opportunities for Improvements*)

Test Environment			
Test Network	Update test environment for Network Functions Virtualization.	**Large** This requires planning and building of test network, including ordering of the new hardware. This work has a long lead-time, because of the delivery times and also due to network building time.	Expand the existing test network with virtualized Session Gateway Controllers (SGCv) and Virtualized Border Gateway Function (BGFv). The new virtualized components are all to be located in a cloud, which needs to be managed and orchestrated by a cloud manager. Study possibility use Reference Configurations (Section 6.5.1) Study the needs for hybrid network. (e.g. O&M, comparison of performance)
Test Tools	Test tool needs requires deeper analysis.	Tool analysis is related to other actions above. They are listed here only to make them more visible.	Adapt/Align test tools for Multi-application CI.
			Implement more automation
			Analyze tools needs for Media Quality
			Analyze tools needs for characteristics measurements
	Production like tools (lifecycle management) and traceability needed.	Analysis of the needs in ways of working requires and preparation of new guidelines some effort.	Develop ways of working for tools and scripting/test programs. Production like tools (lifecycle management) , traceability between tools/scripts/test programs and test configuration is needed.

The changes proposed in the test strategy and the actions for deployment are presented in Table 18 above. The magnitude of the change is estimated in three levels: Small, Medium and Large. These estimations are based on how much the test scope will change, and whether the number of test cases is expected to increase, decrease or stay the same compared to the existing tests. These are rough estimations. Thorough test analysis is required to make these estimations more precise.

8 Evaluation of the Proposed Changes

The proposed changes in the system test strategy for adaptation were evaluated. This section discusses the evaluation, and its results. The section starts with presenting the evaluation and, based on the evaluation results, presents next the upgraded list of proposed changes and deployment plan.

8.1 Evaluation

The persons who participated in the evaluation and the division of the areas of evaluation are presented in Table 3. The persons selected for the evaluation are subject matter experts of the release areas and people working in BGF product related positions in variety of roles. Additionally a couple of section managers from the system test area participated in the evaluation. The group of people selected represents key stakeholders and experts in the case organization. The evaluation per each release area and the evaluation of the overall view were done individually. In addition, the walkthrough of the changes was done in weekly meetings with instructors of the study, whom are also experts in testing area.

8.2 Upgraded Proposal for the Changes and Action Plan

The evaluation showed that there were two proposed changes that needed to be modified. First the area of *Characteristics and ISP* did not cover sufficiently the In Service Performance (ISP) aspects, and second the *Upgrade and Expansions* area requires re-consideration.

The *Characteristics and ISP* area needs to be updated to cover also the In Service Performance (ISP) requirements. There are two such requirements at the moment and they both are related to upgrade functionality 1) Minimal ISP impact at SW Upgrade and 2) Downtime in upgrade less that 60 seconds. The In Service Performance (ISP) requirements require deeper investigations of ISP requirements.

The release area *Upgrade and Expansions* was proposed to be changed to *Upgrade and Manual Scaling.* There were two issues in the proposed changes in this area. First, the name was not perfect. There are no upgrade paths, and the whole upgrade method changes in general, in a virtualized product, it might be

unnecessary to use *Upgrade* in the name of the area. Second, manual scaling was discussed during the evaluation of this area. Is it necessary to test manual scaling in any release area, if it was tested already in functional tests in cross-functional teams. In other words, it was discussed whether it were possible to cover manual scaling already in the cross-functional teams. Based on this, there would be only testing of *Portability* left in the scope of the Upgrade and Manual Scaling. Bringing the manual scalability in the functional tests performed by cross-functional teams will create a new need for the teams. They need to be able to use traffic load generators to create the needed background traffic load. They should also be able to perform these tests in the network test environment. This is a competence issues, but also a resource issue. There must be such test environments with traffic generators for background traffic available for many cross-functional teams. Additionally this requires planning and coordination between the teams, so that not all the teams should perform manual scaling for every sprint. Nevertheless, this is an issue that needs to be studied further.

The name of the area had to be changed. If the manual scaling is not covered in this area and the only issue left would be portability, the area could be called *Release Area Portability.* Anyhow, the Portability would not be that describing name for the release area. In practice the Portability means Lifecycle management of Virtual Network Functions (VNF). Therefore the area could be re-named as *VNF Lifecycle Management.* If the manual scaling is decided to be kept in this area the proposed name, *VNF Lifecycle Management* could still be descriptive, since manual scaling also means creating and termination of Virtual Network Functions.

The upgraded Proposed Changes and Action Plan is presented in the table below (Table 19):

Table 19. Upgraded Proposed Changes and Action Plan

Subject to Change	Change	Magnitude	Actions for Deployment of the changes.
Release Areas			
Upgrade and Expansions	Upgrade paths and expansions do not exists in BGFv.	**Medium** The test scope adaptation requires Implementing new test cases. The number of Test cases is expected to decrease in total. **From evaluation:** To be studied where to locate manual scaling. If moved in cross-functional teams also competence issues need to be considered, since these tests require traffic load and use of load tools.	Remove upgrade paths and expansions from the test scope of virtualized product.
	New requirements for Manual scalability & portability.		Include Manual scalability and portability (VM lifecycle management). Consider moving manual scaling to functional tests performed by cross-functional teams.
	Renaming of the area.		Rename to: '*VNF Lifecycle Management*'
O&M	New architecture for O&M	**Large** The test scope will change a lot. The number of test cases is expected to increase. The Test analysis and implementing new test cases will require much resources and time.	Cover all the functional blocks in NFV Management and Orchestrator (MANO) in O&M tests.
	Transition requirement.		To be considered: Covering of Management of Hybrid Network in tests.
	New automated O&M mechanisms.		Verify the NFV framework's capability to automatically create, scale and heal of VNFs based on pre-defined criteria.
	New Service Assurance requirement for O&M.		Test that Network functions are remotely accessible, monitored, and can perform diagnosis.
	Renaming of the area.		Rename to: '*Management and Orchestration*'
Signaling	Virtualization will not call for any changes in this area, and after moving to all IP networks testing of different signaling standards is not needed.	**Removed Area**	Remove this area and all the tests in it.

Single Traffic & Features	Multi-vendor NFV environment	**Large** The new test scope needs to be set. The number of test cases is expected to increase.	Analyze new needs for Interoperability and Integration testing
	Transition requirement.		To cover co-existence and transition testing with legacy network
	Increased need for network level integration.		Balance the test scope between this release area and Network level Feature Integration.
Media Quality	Multi-vendor NFV environment	**Medium** The test scope needs not to be changed. And the number of test case is expected to be equal compared to the native BGF. Test tools need to be analyzed more deeply. They may require changes.	Analyze changes needed in test environment
	New counters and ways to collect counter data.		Analyze the needs to reorganize counters and the collection of the counter data including tools used.
	Both above.		Analyze possibilities and needs in test automation and implement changes.
Stability	New scaling functionality introduced in NFV.	**Medium** The ne test scope needs to be set. The number of test ceases is expected to increase.	Include both scaling up and scaling down in the stability tests.
			Cover the functionality of scaling criteria the stability tests.
			Changes in Test automation
Robustness	Multi-vendor NFV environment	**Large** The new test scope needs to be set. The number of test ceases is expected to be equal compared to the native BGF. The Test analysis and implementing new test cases will require much resources and time.	Include *Service continuity* testing (e.g. HW failure or a resource shortage/outage or *Scalability, and Elasticity in cloud* in a failure situation
			Include testing of functionality of Fault management in multi-vendor environment.
	Renaming of the area.		Rename to: '*Resiliency*'

Characteristics & ISP	Multi-vendor NFV environment	**Large** The test scope needs to be set. The number of test ceases is expected to be equal compared to the native BGF. The analysis of tool needs and setting up such environment that is capable to provide input for dimensioning even on real time, requires resources and time. (This is common activity for ways of working area.) This requires own analysis, but is not expected to be a big issue in resource wise.	Verify that the NFV framework is independent of HW and framework is capable to collect performance related information.
			Tie the measurements and the configuration (including tools) used to each other's.
	Unpredictable environment. It is impossible to make estimations for dimensioning by reading specifications.		Provide Measurements as input for dimensioning.
	ISP requirements		Analyze what are the In Service Performance (ISP) requirements to be covered.
Vulnerability	Multi-vendor NFV environment.	**Large** Vulnerability analysis is needed. The test scope may be adjusted based on it. The number of test cases is expected to increase.	Perform comprehensive vulnerability analysis Due to big changes in the environment to define the new test scope.
Feature Integration			
Feature Integration	Multi-vendor NFV environment.	**Medium** The new test scope needs to be set. The number of test cases is expected to increase.	Analyze the need for Interoperability testing, multi-vendor integration, end-to-end type of integration and also testing in hybrid network. Balance Feature Integration tests with functional tests performed by cross-functional teams and Single Traffic & Features Release area.

Way of working			
Characteristics Measurements	Unpredictable environment. It is impossible to make estimations for dimensioning by reading specifications.	**Medium** This requires implementing a new working process including Tool that is capable to provide input for dimensioning even on real time. (This is common activity with Characteristics & ISP area.)	Adapt to need for Fast feedback information of the characteristics measurements for dimensioning. It will require change in tooling and ways of working. Fast characteristics feedback should be provided even on real time.
Test Automation	More automated test cases are needed in non-functional tests.	**Medium** This requires analysis and study of new tools, including piloting and trials.	Study possibilities to use Fault Injection in Resiliency tests. Try to find and analyze also other possible ways to automate more non-functional tests.
Multi-application CI	Multi-application CI needs resources for support and also to align test environment, test tools and ways of working in to it. This is, resources for preparation work.	**Medium** This requires resource planning for both human and machine. The actual support and the amount of work needed for aligning tools, test environment and work processes will be discovered later when few trials with Multi-application CI are done.	Prepare to allocate resources from Media Plane Development organization. To support CI activities To align test environment, test tools and ways of working in to it.

Competence and Skills	New competence needs related to Network Functions Virtualization are needed.	**Large** This requires proper competence planning, including learning objects and the learning order. All the actions in this area (i.e. People) are related directly or indirectly in competence management.	Analyze the competence needs to build up the new competencies.
	The competencies related to the new system architecture and new system components.		Prepare competence build up in the area of Architecture and components.
	New test tools, test use cases, end-to-end/network test environment(s).		Prepare competence build up in the area of Testing tools and environment.
	New testing and diagnostics, troubleshooting skills are required.		Prepare competence build up in the area of Testing and diagnostic skills.
	In addition for people working in system testing the new Competencies may be necessary also for people working in cross-functional teams, especially if they will perform part of the network level feature integration tests.		Try new ways for learning need to study: c) Specification by Example d) Development and Operations, DevOps. (See Section 9. *Opportunities for Improvements*)

Test Environment			
T**est Network**	Update test environment for Network Functions Virtualization.	**Large** This requires planning and building of test network, including ordering of the new hardware. This work has a long lead-time, because of the delivery times and also due to network building time.	Expand the existing test network with virtualized Session Gateway Controllers (SGCv) and Virtualized Border Gateway Function (BGFv). The new virtualized components are all to be located in a cloud, which needs to be managed and orchestrated by a cloud manager. Study possibility use Reference Configurations (Section 6.5.1) Study the needs for hybrid network. (e.g. O&M, comparison of performance)
Test Tools	Test tool needs requires deeper analysis.	Tool analysis is related to other actions above. They are listed here only to make them more visible.	Adapt/Align test tools for Multi-application CI.
			Implement more automation
			Analyze tools needs for Media Quality
			Analyze tools needs for characteristics measurements
	Production like tools (lifecycle management) and traceability needed.	Analysis of the needs in ways of working requires and preparation of new guidelines some effort.	Develop ways of working for tools and scripting/test programs. Production like tools (lifecycle management) , traceability between tools/scripts/test programs and test configuration is needed.

The upgraded Proposal for Changes and Action Plan for deployment is presented in Table 19 above. The changes made after the evaluation are marked with yellow background color.

8.3 Summary

The evaluation of the proposed changes and actions was based on the expertise of the experts. Two issues were found in the evaluation. First, portability and manual scaling requires re-thinking in the release area *Upgrade and Expansions*. Second, In Service Performance (ISP) view was missing from the release area *Characteristics and ISP*. This area requires to be analyzed further from the ISP point of view requirements.

Additionally the name of the release area *Upgrade and Expansions* needs to be re-considered. The name '*VNF Lifecycle Management'* was suggested.

All the changes were considered to be of the size of *Medium* or *Large* of their magnitude. The only small change is the removal of the *Release Area Signaling*. At this point no need for new release areas was discovered. Changes are big; some of the changes will have a long lead-time, as for example, competence build up. That is why, new ways of building new competencies were proposed in this study.

9 Opportunities for Improvements

This section discusses the opportunities for improvements that are possible because of the virtualization, or that may be required for a success of efficient testing in the future. The next two sub-sections are related to people, resource allocation and competence build-up. First, a method to define test cases as collaboration with stakeholders, *Specification by Example*, is discussed. Second a concept of Development and Operations, *DevOps* is presented. They are both ways to spread competence and information within, and even between the teams. Third sub-section is about the opportunities virtualization brings in *the area of test tools*. The fourth sub-section continues with the test tools by discussing *the fault injection*.

9.1 Specification by Example

As it seems, more testing should be performed in the network environment. It means that also part of the functional testing performed by the cross-functional teams should be executed in network environment with the same tools that are used for network level feature integration. To bring functional tests in the network environment requires understanding of the functionality on the network level, and knowledge about the tools in that environment. To make it easier to create test cases on the network level, Acceptance Test Driven Development would be a considerable option. Not all cross-functional teams have competence to perform functional test in the end-to-end network environment. The acceptance test driven development as a method would ease up building the needed competence.

Specification by Example is also known as Acceptance Test Driven Development (ATDD) or Behavior Driven Development (BDD). It is based on the following five ideas. First, it uses real examples to build a shared understanding. Second, it uses a selected set of these examples for acceptance tests. Third, these cases are automated. Fourth, it focuses on the acceptance tests. Fifth, these automated acceptance test cases are used to facilitate discussion about possible change requests in the future. (Adzic G. 2009:31,32).

When a cross-functional team and a possible supportive independent test team do not participate in the designing of the specifications, it means that the teams

need to be informed about these specifications separately. This may lead to possible misunderstandings and lost details. Therefore, the outcome from the teams is possibly not as expected. They will develop something that is not what it was meant to be, and even test that it works, as they understood it. To correct this kind of misunderstood functionalities is all unnecessary rework. In the Specification by Example method, the idea is to specify the solution collaboratively with people with wide range of diversity using different heuristics to solve problems. Technical experts know the system; testers know how the system may break and where the potential issues might be hiding. All this is knowledge is important for the successful specification. Making specification collaboratively makes it possible to share the knowledge and experience, and getting people contribute and more involved in the whole delivery process. (Adzic G. 2011:Ch 2, Ch 6)

This is feasible also for network integration testing. Already in the sprint planning cross-functional teams should discuss the test cases and agree on the acceptance test cases that will be run in network test environment during the sprint to show what was implemented in the sprint works. Planning of the test cases on a high level would not take many hours, but it would require participation also from product owners, technical experts on a need basis, and possibly also from release areas. Discussing the test cases on the network level would give common understanding about the new feature, and draw a big picture on people's minds as to how the feature is meant to work. If the cross-functional team is lacking competence to manage with functional tests in the network test environment, supportive test teams should also help in the planning session to find the best test cases for the acceptance. The aim should also be to automate these acceptance test cases. They can then later be included into the regression test suite in continuous integration to verify the legacy functionality.

9.2 DevOps

DevOps, Development and Operations, is a topical trend in software development. According to Roche J. (2013) developers work mostly on code, whereas operations work mostly with systems. The DevOps is a mix skill set of Developers (including quality assurance) and operations person. The figure below (Figure 17.) illustrates the mix of different skills of DevOps.

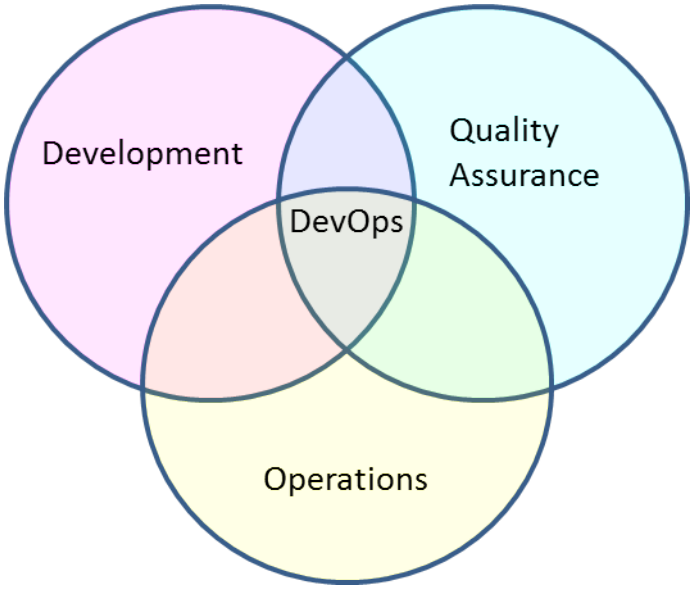

Figure 17. The mix of different skills of DevOps.

Figure 17 above illustrates DevOps as the intersection of development, operations and quality assurance. Testing can be seen as a part of the development and quality assurance. System testing as it is described in this study is merely on the Quality Assurance area, although part of the system testing is done in Development.

To be able to start network level feature integration, cross-functional teams need to have additional competencies compared to what they do have today. To bring people from today's test teams in cross-functional teams would bring the teams closer to DevOps teams. To aim towards DevOps teams would be a more permanent change than using specification by example method (discussed in Section 9.1). If the target is to move towards DevOps teams, Specification by Example could be used as a vehicle to accelerate transition from cross-functional teams to DevOps teams.

Whether t is wise to move people from test teams (Quality Assurance) to cross-functional teams or not, is hard to say. There are also people who think the separate test teams are needed. There has been long lasting debate over the years in

Agile movement around this subject. There is more about the team division in Appendix 5.

One thing to consider on a way to DevOps is the utilization of Test Environment resources. Will the need grow? Will the resources be used effectively?

9.3 Test Tools and Virtualization

Network Functions Virtualization will add flexibility also in the test environment. It will enable using different HW and different cloud setups. Utilizing virtualization also in test tools will bring value to testing. If test tools are virtualized they can be located in the same environment with the system under test. This means portability of testing in a sense, that the whole test environment may be setup on the need basis very quickly as Test-Environment-On-Demand. There might be possibilities to e buy or rent HW and cloud outside the local company.

Virtualization of test tools may also have a positive effect on test automation. Having test tools as virtualized applications in the test environment could simplify the test automation environment and bring some advances also for test automation efficiency, and make test automation more product alike.

9.4 Fault Injection

Fault Injection or Failure-as-a-Service (FaaS) is definitely an area that needs to be studied further. FaaS is a virtualized application that is run in the test environment. It creates fault situations, for example, failure modes that could happen simultaneously on a large number of components, failures such as disc failures, network failures (packet loss/delays), termination of VMs (Haryadi S. et al. 2011). It will enable testing of different kinds of failures, fault situations, in the network by using automated tests. Normally these kinds of tests would require manual intervention. Therefore this would provide a way to automate more Resiliency test cases.

10 Discussion and Conclusions

This section makes the conclusion of the study. It contains the summary and evaluation of the study with discussion about the validity and reliability of it. It also proposes steps for the future.

10.1 Summary of the Study

The objective of this study was to develop a system testing strategy that will address the change related virtualization of BGF product. This study was supposed to answer the research question: *How to adapt the current system testing for the virtualized border gateway function (BGFv)?*

The current context of the system test in Media Plane Development organization was analyzed and used as a base for this study. The analysis was made to ease understanding the needs for changes. The next step was literature review, which was made to collect information and knowledge about the Network Functions Virtualization. During the review the requirement documentation produced by ETSI Network Functions Virtualization Industry Specification Group was found very valuable. Therefore these ETSI documents were used the most. Based on the literature review, the *Network Functions Virtualization concept was presented* including the changes and the new requirements it will introduce in the telecom products and networks. These requirements were analyzed next together with the information about the current context of the system test in Media Plane development organization, to see whether they had impact on the system test or not. After this impact analysis, the changes in the current system tests of Border Gateway Function (BGF) were proposed. The proposed changes form the adaptation of the system test strategy for the virtualized Border Gateway Function (BGFv). These changes are in test scope, ways of working, test environment and test tools.

The Network Functions Virtualization environment triggers the biggest changes in system testing, precisely on non-functional area of the system. Therefore most of the changes in the system test strategy are in the non-functional tests that are covered by release area concept in Media Plane Development. Naturally the biggest changes will be in the areas where the environment will change the most. One of these areas is *Operations and Maintenance* area, where the whole archi-

tecture will change. Another such area is *Upgrade and Expansions*. The areas of *Robustness* (Resiliency) and V*ulnerability* will require rather big changes in test scope. The areas of *Media Quality* and *Characteristics and In Service Performance* need also changes, but in these areas the scope should not change. In these areas the changes in testing will be more on tooling and test environment. The area of *Stability*, need only to enlarge the scope to include also scalability (both up and down). The area of *Signaling* may be removed totally. There is no need to test different signaling standards in all IP network. The area of *Single Traffic and Features* needs to be adapted to the increasing needs of network level integration and multi-vendor environment. This should be done considering also other feature integration activities.

All the changes above are a high-level view of changes in test scope. The proper test analyses need to be done per each supported network configuration. All this depends on what are the components used, for example, what is the cloud platform, what is the Virtual Infrastructure Manager (VIM), what is the hypervisor used. The test strategy is generic and cannot include this detailed information.

10.2 Evaluation of the Study

The evaluation of the outcome of the study compares the research objectives defined in the beginning of this study against its final outcomes. It includes also evaluation of the validity and reliability of this study.

10.2.1 Outcome vs. Objectives

The objective of this study was to develop a system test strategy for Media Plane Development Organization that will address the change related virtualization of BGF product. In practice this was an adaptation of the current system test strategy for the virtualized BGF.

The outcome of this study is two folded. First, the study presents a list of requirements and issues collected from the analyzed input documents (Section 4). Second, it proposes changes in the system test strategy of Media Plane Development, ways of working and test environment, triggered by the requirements and issues listed in the previous phase (Section 5 and 6).

The study included addressing of five sub-questions. The *First sub-question* was to clarify what are the requirements for the virtualization. To address this first sub-question the study presents the concept of Network Functions Virtualization and the ETSI NFV (SG) requirements for it. The *Second sub-question* was to clarify how the virtualized border gateway function (BGFv) will operate. This was addressed by presenting the internal factors, the changes in the Media Resource Subsystem architecture and the Ericsson Cloud System together with the Network Functions Virtualization concept. The *Third sub-question* was to clarify how system testing needs to be changed for the new virtualized environment. The Fourth, the adaptation of the present system test strategy for the new virtualized border gateway function (BGFv) was expected in the fourth sub-question. *The third and fourth sub-questions* are addressed by proposing changes in System Test Strategy, ways of working and test environment in the Media Plane Development. The *Fifth sub-question* was the evaluation of the proposed changes in strategy validated by the key experts/ stakeholders in Media Plane Development. This was addressed by evaluating the proposed changes in the System Test Strategy. Additionally also the proposed changes in ways of working and test environment were evaluated.

Summing up, it can be considered that the outcome of the study answered the research question and all the all the sub-questions. The study fulfills the objective to address the change needed to adapt the current system test strategy to virtualized Border Gateway Function (BGFv).

10.2.2 Validity and Reliability

The study started with a literature review for the relevant input material. The next step was to analyze the current context of the system testing for native Border Gateway Function (BGF). This analysis was done using the exploratory case study approach. The relevant input material was analyzed together with the information about the current system test context in Media Plane Development, to see what are the requirements and issues that will impact the current system test context. It can be said that the study followed qualitative research methods.

In qualitative research, *the validity determines, whether the study truly measures what it was intended to* (Golafshani N. (2013): 599). As it was said in the preceding section the study addresses the research question and all its sub-questions. In this study the research question and the sub-questions were selected so that they helped to structure the study in a way, which then led to the relevant results. The subject matter experts, and relevant stakeholders evaluated the outcome. Based on the evaluation the results are considered to be relevant.

The results of the study are not aimed at generalizing it to other contexts, if not thinking the release area concept as such, but to a more general, non-functional testing on system level. The changes proposed in the study are specific and based on the case company's context as well as the general ETSI Network Functions Virtualization requirements. The non-functional system tests are general on the strategy level but specific in product related issues in details. Therefore, the outcome of the study is meant as learning from the case rather than generalizing the outcomes to other organizations, without making any specific adjustments. Due to the same reasons, no comparison or benchmarking was possible, between different companies.

In general the study worked well according to the plan. The only thing that caused some extra work, was that the changes due to the Network Functions Virtualization environment was to lead the study too much on test analysis type of analysis, which was not the plan. There was also problems in finding information about testing or ways of working from the companies that were in the same situation, or even further in virtualization. Therefore it would not be possible to do any comparison, benchmarking, between different companies. Additionally the writing work was more challenging that expected.

Reliability of this study can be evaluated by assessing 1) the reliability of the used input material, 2) if it was possible to get same results by repeating this study. When it comes to the reliability of the used input material, most of it is produced by ETSI NFV Industry Specification Group. There are also articles from IEEE and ACM. Additionally also some articles written by consultant companies were used, but these are used merely only to get some practical input information from companies that have experiences from virtualization. The study would most probably

provide the same result if done again wit the same input information in the same case organization, but due to the fact that this is evolving area of technology, it is not very likely that the study would be done with the same input information, even done in the same organization. Therefore also the results would be different. This does not decrease the reliability of this study, or its results. This study was done based of the latest available information, and the selected method was experienced to be valid.

The outcome of this study for the case company is the evaluation of impacts that can be utilized in the case company. Additionally, the learning from this specific case may be used in the field of system testing in other organizations, when they face the same challenging situation, since the outcome is written on high-level from tests strategy point of view. On the level, that is not that much dependent on the product or the context.

10.3 Future Steps

In the evaluation of the study two issues for the further studies showed up. 1) To plan where to locate testing of Portability and Manual scaling. 2) To analyze what other than upgrade related ISP requirements there are and plan how to include them in Characteristics and ISP release area. Additionally during the study some issues were discovered to be worth of future steps. These issues are described in the Section 9, *Opportunities for Improvements*. Two of them are related to people and the ways to support learning of new things in the teams. The first one is Specification by Example method, to add communication between all the stakeholders, and the second is DevOps teams. DevOps team set-up is a way to expand the competence in the teams to cover also testing in production like environment and maintenance support. Two other improvements are related to test tools and virtualization. By virtualizing test tools it would be possible to reach more effective way to utilize test environments, Test-environment-on-demand. Using Fault Injection tools could expand test automation in resiliency tests. The Fault Injection would require more studying. According to Haryadi S. et al. (2011), there are online failure-injection frameworks, like for example, Netflix's Chaos Monkey and Amazon GameDay. These are most probably not feasible for resiliency tests for virtualized BGF, but good starting point for the further studies in the area.

References

3GPP HSPA (n.d.). 3GPP Acronyms, HSPA. Available: http://www.3gpp.org/technologies/keywords-acronyms/99-hspa [Accessed 2015-03-31]

3GPP LTE (n.d.). 3GPP Acronyms, LTE. Available: http://www.3gpp.org/technologies/keywords-acronyms/98-lte [Accessed 2015-03-31]

Adzic G. (2009). Bridging the Communication Gap: Specification by Example and Agile Acceptance Testing, Gojko Adzic, January 5 2009, Neuri Limited

Adzic G. (2011). Specification by Example: How Successful Teams Deliver the Right Software. Gojko Adzic, June 6 2011, Manning Publications Co

Baxter P. and Jack S. (2008). *Qualitative Case Study Methodology: Study Design and Implementation for Novice Researchers*, Pamela Baxter and Susan Jack, McMaster University, West Hamilton, Ontario, Canada, The Qualitative Report Volume 13 Number 4 December 2008, pages 544-559. Available from: http://www.nova.edu/ssss/QR/QR13-4/baxter.pdf [Accessed 2015-01-25]

Capgemini (2014). *World Quality Report, 2013-2014*, Capgemini, Sogeti and HP, September 12, 2013, Fifth Edition. Available from: http://www.capgemini.com/thought-leadership/world-quality-report-2013-14 [Accessed 2014-10-23]

Cotroneo D. et al (2014). IEEE: *Network Function Virtualization: Challenges and Directions for Reliability Assurance.* Authors: D. Cotroneo, L. De Simone, A.K. Iannillo, A. Lanzaro, R. Natella Critiware s.r.l. / Federico II University of Naples, Italy and Jiang Fan, Wang Ping Huawei Technologies Co., China. Available from: http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6983797 [Accessed 2015-01-08]

Csatári G. and László T. (2013). IEEE Conference Publications: *NSN Mobile Core Network Elements in Cloud, A proof of concept demo*, Authors: Gergely Csatári, Tímea László MBB/VIPT, T&S/Research Nokia Siemens Networks Budapest, Hungary, Publication Year: 2013 , Pages: 251 – 255. Available from: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6649238 [Accessed 2015-01-02]

Daitan Group (2014). *Attesting to the Benefits of NFV, Daitan White Paper, Building Integrated Cloud Communication Services*, Daitan Group 2014. Available from: http://www.DaitanGroup.com [Accessed 2015-01-06]

Ericsson (n.d.a). Ericsson Product Catalogue, IMS (IP Multimedia Subsystem), Available from: http://www.ericsson.com/ourportfolio/products/ims?nav=productcategory002 [Accessed 2014-10-22]

Ericsson (n.d.b). Ericsson Media Resource System MRS, Available from: http://www.ericsson.com/ourportfolio/products/media-resource-system-mrs?nav=productcategory002%7Cfgb_101_432 [Accessed 2014-10-22]

Ericsson (n.d. c). Ericsson Press release, 2013-02-13, *Ericsson defines the cloud evolution.* Available: http://www.ericsson.com/news/1677723 [Accessed 2014-10-10]

Ericsson (n.d.d). Ericsson Product Catalogue, Ericsson Cloud Manager. Available at: http://www.ericsson.com/ourportfolio/products/cloud-manager [Accessed 2015-02-10]

Ericsson (n.d.e). Ericsson Product Catalogue, Infrastructure as a Service. Available at: http://www.ericsson.com/ourportfolio/products/infrastructure-as-a-service [Accessed 2015-02-18]

Ericsson (2014). Ericsson White paper Uen 284 23-3249 | December 2014, *Wide area network Traffic Engineering, meeting the challenges of the distributed cloud.* Available from: http://www.ericsson.com/res/docs/whitepapers/wp-wan-traffic-engineering.pdf [Accessed 2015-01-02]

ETSI (2013-10 a). ETSI: *Network Functions Virtualisation (NFV); Virtualisation Requirements* V1.1.1. (2013-10). Available from: http://docbox.etsi.org/ISG/NFV/Open/Published/gs_NFV004v010101p%20-%20Virtualisation%20Requirements.pdf [Accessed 2014-12-30]

ETSI (2013-10 b). ETSI: *Network Functions Virtualization (NFV); Use Cases* (V1.1.1 (2013-10)). Available from: http://docbox.etsi.org/ISG/NFV/Open/Published/gs_NFV001v010101p%20-%20Use%20Cases.pdf [Accessed 2015-01-02]

ETSI (2014-06). ETSI: *Network Functions Virtualization (NFV); NFV Performance & Portability Best Practises* V1.1.1 (2014-06). Available from: http://docbox.etsi.org/ISG/NFV/Open/Published/gs_NFV-PER001v010102p%20-%20Perf_and_Portab_Best_Practices.pdf [Accessed 2015-01-02]

ETSI (2014-10-22). ETSI: *Network Functions Virtualisation An Introduction, Benefits, Enablers, Challenges & Call for Action* (October 22-24, 2012). Available from: http://portal.etsi.org/NFV/NFV_White_Paper.pdf [Accessed 2014-10-10]

ETSI (2014-10 a). ETSI: *Network Functions Virtualisation (NFV); NFV Security;Problem Statement* V1.1.1 (2014-10). Available from: http://docbox.etsi.org/ISG/NFV/Open/Published/gs_NFV-SEC001v010101p%20-%20Security_Problem_Statement.pdf [Accessed 2015-01-02]

ETSI (2014-10 b). ETSI: *Network Functions Virtualisation (NFV); Infrastructure; Methodology to describe Interfaces and Abstractions* V1.1.1 (2014-10). Available from: http://docbox.etsi.org/ISG/NFV/Open/Published/gs_NFV-INF007v010101p - Methodology_to_describe_Interfaces_and_Abstractions.pdf [Accessed 2015-01-05]

ETSI (2014-10-17). ETSI: *Network Functions Virtualisation (NFV) Network Operator Perspectives on Industry Progress (*October 14-17, 2014). Available from: http://portal.etsi.org/NFV/NFV_White_Paper3.pdf [Accessed 2014-10-25]

ETSI (2014-12 a). ETSI: *Network Functions Virtualisation (NFV); Architectural Framework* V1.2.1 (2014-12). Available from:

http://docbox.etsi.org/ISG/NFV/Open/Published/gs_NFV002v010201p - Architectural Framework.pdf [Accessed 2015-01-17]

ETSI (2014-12 b). ETSI: *Network Functions Virtualisation (NFV); NFV Security; Security and Trust Guidance* V1.1.1 (2014-12). Available from: http://docbox.etsi.org/ISG/NFV/Open/Published/gs_NFV-SEC003v010101p - Security and Trust Guidance.pdf [Accessed 2015-01-17]

ETSI (2014-12 c). ETSI: *Network Functions Virtualisation (NFV); Virtual Network Functions Architecture* V1.1.1 (2014-12). Available from: http://docbox.etsi.org/ISG/NFV/Open/Published/gs_NFV-SWA001v010101p - Virtual Network Functions Architecture.pdf [Accessed 2015-01-17]

ETSI (2014-12 d). ETSI: *Network Functions Virtualisation (NFV); Service Quality Metrics* V1.1.1 (2014-12). Available from: http://docbox.etsi.org/ISG/NFV/Open/Published/gs_NFV-PER001v010101p-%20Perf_and_Portab_Best_Practices.pdf [Accessed 2015-01-17]

ETSI (2014-12 e). ETSI: *Network Functions Virtualisation (NFV); Infrastructure Overview* V1.1.1 (2015-01). Available from: http://docbox.etsi.org/ISG/NFV/Open/Published/gs_NFV-INF001v010101p - Infrastructure Overview.pdf [Accessed 2015-01-17]

ETSI (2014-12 f). ETSI: *Network Functions Virtualisation (NFV);Infrastructure; Compute Domain* V1.1.1 (2014-12). Available from: http://docbox.etsi.org/ISG/NFV/Open/Published/gs_NFV-INF003v010101p - Infrastructure - Compute Domain.pdf [Accessed 2015-01-17]

ETSI (2014-12 g). ETSI: *Network Functions Virtualisation (NFV); Infrastructure Network Domain* V1.1.1 (2014-12). Available from: http://docbox.etsi.org/ISG/NFV/Open/Published/gs_NFV-INF005v010101p - Infrastructure - Network Domain.pdf [Accessed 2015-01-17]

ETSI (2014-12 h). ETSI: *Network Functions Virtualisation (NFV); Management and Orchestration* V1.1.1 (2014-12). Available from: http://docbox.etsi.org/ISG/NFV/Open/Published/gs_NFV-MAN001v010101p - Management and Orchestration.pdf [Accessed 2015-01-17]

ETSI (2015-01 a). ETSI: *Network Functions Virtualisation (NFV); Resiliency Requirements* V1.1.1 (2015-01). Available from: http://docbox.etsi.org/ISG/NFV/Open/Published/gs_NFV-REL001v010101p%20-%20Resiliency%20Requirements.pdf [Accessed 2015-01-17]

ETSI (2015-01 b). ETSI: *Network Functions Virtualisation (NFV); Infrastructure; Infrastructure; Hypervisor Domain* V1.1.1 (2015-01). Available from: http://docbox.etsi.org/ISG/NFV/Open/Published/gs_NFV-INF004v010101p - Infrastructure - Hypervisor Domain.pdf [Accessed 2015-01-17]

EZchip (2014). EZchip Technologies Inc. (April 29. 2014) : *NFV Acceleration with the EZchip NPS-400 Network Processor, White Paper* . Available from: http://www.ezchip.com/files/drim__7510.pdf [Accessed 2015-01-06]

Golafshani N. (2013). Golafshani Nahid. December 2003 [597-607]. Understanding Reliability and Validity in Qualitative Research, University of Toronto, Toronto, Ontario, Canada. Available from: http://www.nova.edu/ssss/QR/QR8-4/golafshani.pdf [Accessed 2015-03-13]

Gregory J. and Crispin L. (2009). *Agile Testing: A Practical Guide for Testers and Agile Teams,* Authors: Janet Gregory and Lisa Crispin, January 2009, Addison-Wesley

Gregory J. and Crispin L. (2014). *More Agile Testing, Learning Journeys for the Whole Team*, Authors: Janet Gregory and Lisa Crispin, October 2014, Addison-Wesley

Haryadi S. et al. (2011). Failure as a Service (FaaS): A Cloud Service for Large-Scale, Online Failure Drills. EECS Department, University of California, Berkeley Technical Report No. UCB/EECS-2011-87. Available from: http://www.eecs.berkeley.edu/Pubs/TechRpts/2011/EECS-2011-87.html [Accessed 2014-11-23]

ISO/IEC 7498-1 (1994). ISO/IEC 7498-1:1994. Information technology -- Open Systems Interconnection -- Basic Reference Model: The Basic Model Available from: http://standards.iso.org/ittf/licence.html [Accessed 2015-03-31]

ISO/IEC/IEEE 29119 (n.d. a). *ISO/IEC/IEEE 29119 Software and systems engineering Software testing Part 1:*Concepts and definitions. IEEE Xplore. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=6588537&queryText%3Diso+29119 [Accessed 2015-01-16]

ISO/IEC/IEEE 29119 (n.d. b). *ISO/IEC/IEEE 29119 Software and systems engineering Software testing Part 3:Test documentation*. IEEE Xplore. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=6588540&queryText%3Diso+29119 [Accessed 2014-12-28]

ISTQB (n.d.). ISTQB (International Software Testing Qualifications Board) Exam Certification. Available: http://istqbexamcertification.com/what-is-system-testing/ [Accessed 2014-10-11]

Jellema B. and Vorwerk M. (2014) Ericsson Review: *Communications as a cloud service: a new take on telecoms* (2/2014). Available from: http://www.ericsson.com/res/thecompany/docs/publications/ericsson_review/2014/ericsson-review-2-2014.pdf [Accessed 2015-02-10]

Ju X. et al (2013). Ju, X., Soares, L., Shin, K.G., Ryu, K.D., De Silva, D.:*On Fault Resilience of OpenStack.* Available from: http://dl.acm.org/citation.cfm?id=2523622&dl=ACM&coll=DL&CFID=470420918&CFTOKEN=82983121 [Accessed 2015-01-08]

Kanso A. and Lemieux Y. (2013). IEEE Sixth International Conference on Cloud Computing Pages 778-785. Achieving High Availability at the Application Level in the Cloud. Available from: http://dl.acm.org/citation.cfm?id=2515046 [Accessed 2015-02-18]

Lundström J. (2013) Ericsson Review: *Media Processing in the cloud: what, where and how* (April 11, 2013). Available from: http://www.ericsson.com/res/thecompany/docs/publications/ericsson_review/2013/er-voice-video-cloud.pdf [Accessed 2014-10-10]

Nair V. and Gupta V.K. (2014). Aricent Group, Vikram Nair, Vinod Kumar Gupta: *Software Defined Network and Network Functions Virtualization, An Inevitable Evolution for Communication Networks.* Available from: http://www.aricent.com/sites/default/files/whitepapers/Aricent-SDN-NFV-Whitepaper.pdf [Accessed 2015-01-07]

OpenStack (n.d.): *The Open Source Cloud Operating System.* Available: http://www.openstack.org/software/ [Accessed 2015-01-08]

Pek G. et al (2013). A Survey of Security Issues in Hardware Virtualization. ACM Computing Surveys (CSUR) Surveys Homepage archive Volume 45 Issue 3, June 2013 Article No. 40 Available from: http://dl.acm.org.ezproxy.metropolia.fi/citation.cfm?id=2480741.2480757&coll=DL&dl=ACM&CFID=627981846&CFTOKEN=55682721 [Accessed 2014-12-18]

Roche J. (2013). Roche James. Magazine Communications of the ACM Adopting DevOps practices in quality assurance. Volume 56 Issue 11, November 2013. Pages 38-43 Available from: http://dl.acm.org.ezproxy.metropolia.fi/citation.cfm?id=2524713.2524721&coll=DL&dl=ACM&CFID=634114509&CFTOKEN=18012421[Accessed 2015-03-06]

Spirent (2014). *New Testing Methodologies for NFV and Service Chains*, Ethernet Summit 2014. Available from: http://www.ethernetsummit.com/English/Collaterals/Proceedings/2014/20140501_2A_Bojak.pdf [Accessed 2015-01-06]

Tailor M. (2014). Metaswitch Network, Taylor Martin: *A Guide to NFV and SDN* . Available from: http://www.metaswitch.com/sites/default/files/Metaswitch_WhitePaper_NFVSDN_final_rs.pdf [Accessed 2015-01-07]

TTC (n.d.). The Telecommunication Technology Committee home page. Available: http://www.ttc.or.jp/e/ [Accessed 2015-03-31]

Ya-lian Pan et al (2011). *Fault Diagnosis in Network Virtualization Environment* , Ya-lian Pan, Xue-song Qiu, Shun-li Zhang, State Key Laboratory of Networking and Switching Technology, Beijing University of Post and Telecommunications. Telecommunications (ICT), 2011 18th International Conference. Available from: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5898980 [Accessed 2015-01-07]

Portability of VNFs

There are ETSI documents that describe portability more in details. They are not requirements, but may very well be used when thinking testing in details. Portability of VNFs is identified the following ETSI documents:

ETSI: Network Functions Virtualization (NFV); Use Cases (ETSI 2013-10 b). Identifies the need for portability of VNFs in Section 9.2 (virtualized IMS use case) and smooth migration of virtual machines between locations in Section 13.5 (virtualization of fixed access network functions)

ETSI: Network Functions Virtualisation (NFV); Architectural Framework (ETSI 2014-12 a) in Section 7.3.2 describes the scope of NFV to include the ability to guarantee an hardware independent lifecycle, performance and portability requirements of the VNF; and in Section 8.2 describes VNF portability as a study topic in the context of the virtualization layer.

ETSI: Network Functions Virtualisation (NFV); NFV Performance & Portability Best Practises (ETSI 2014-06). Gives many examples of portability and performance issues. For example, the Section Annex C 1.2.3 describes VM migration concept in more details.

Performance Issues in the NFV Environment

In addition to the actual requirements document ETSI 2014-06) discuss the performance issues in the NFV environment. The figure below (Figure 18.) shows the summary of the different workloads:

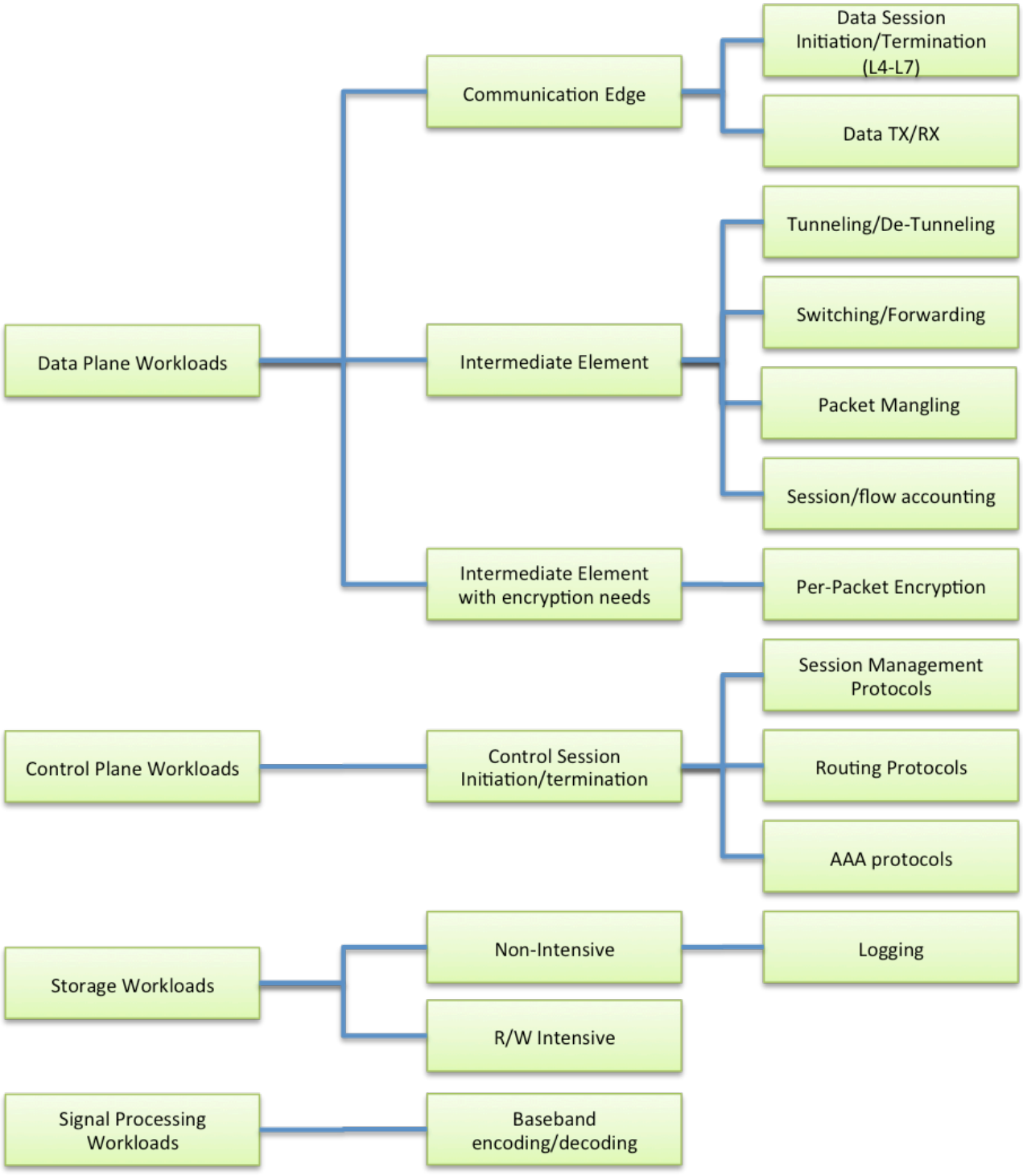

Figure 18. Workload classification. Reprinted from ETSI 2014-06:15.

Figure 18 illustrates the workload types; Data Plane workloads, Control Plane workloads, Signal processing workloads and Storage workloads that may be present in Virtual Network Functions. The referred document “NFV Performance & Portability Best Practises” gives numerous examples of portability and performance issues like, for example, workloads and relations to NFV use cases.

In the same document the section Annex C. 4 “Benchmark Performance Metrics” gives a list of measurable performance metrics for benchmarking. These metrics are categorized on high level in two groups a) Quality of Service (QoS) metrics, which are measuring the network impact on the quality of the service, and b) Quality of Experience metrics, which are to capture the subjective aspects associated with human experience. Examples of the metrics are listed in the table below (Table 20.):

Table 20. Measurable performance metrics. Data gathered from ETSI 2014-06:51-54.

Example	Quality of Service (QoS) Metrics
1	Throughput
2	Latency
3	Frame Loss Rate
Example	Quality of Experience (QoE) Metrics
1	Video Quality Metric (VQM) - Blurring, global noise, block and color distortions.
2	Peak Signal to Noise Ratio (PSNR) - Mean Square Error (MSE) between the original and the received Image.

Table 20 above lists performance metrics that are measurable. In addition to these also the following aspects need to be considered as part of performance measurements: 1) requirements (e.g. for throughput and latency), 2) monitoring of performance over interfaces, and 3) redundancy. (ETSI 2014-10 b:23).

Security Issues in Network Functions Virtualization

There are ETSI documents describing security problems in Network Functions Virtualization with titles: “Network Functions Virtualisation (NFV); NFV Security; Problem Statement” (ETSI 2014-10a) and “Network Functions Virtualisation (NFV); NFV Security; Security and Trust Guidance” (ETSI 2014-12b).

There is also a ACM Computing Survey:” A Survey of Security Issues in Hardware Virtualization” (Pek G. et al 2013:1-9). It describes the vulnerabilities in hardware virtualization. It describes on a general level the three separate software layers; *guest VMs*, *the host operating system*, and *the hypervisor*.

Hardware virtualization allows the sharing of hardware resources via hypervisors, which are software components that intercept all hardware access requests of the VMs and mediate these requests to physical devices. All VMs are isolated from each other, while they use the same virtual platform provided by the hypervisor. Hypervisors provide host-only (internal-to-host) virtual networks that allow for network communication between guests on the same virtual server. This is realized by means of *virtual switches* inside the hypervisor. *Storage virtualization* abstracts physical storage components and provides a single storage device that can be reached directly or over the network. A *host operating system* is able to manage VMs and control hardware resources either directly or via the hypervisor. There is a *management interface*, which allows operators to create, delete, and modify both virtual machines and virtual infrastructures. All the elements mentioned in this paragraph may have vulnerabilities, security problems. This is of course depended of the product, the elements used, and the configuration. (Pek G. et al 2013:1-9)

Issues in OpenStack

OpenStack is a cloud operating system that controls large pools of compute, storage, and networking resources throughout a datacenter, all managed through a dashboard that gives administrators control while empowering their users to provision resources through a web interface (OpenStack n.d.). The OpenStack Cloud Operation system is illustrated in the figure below (Figure 19):

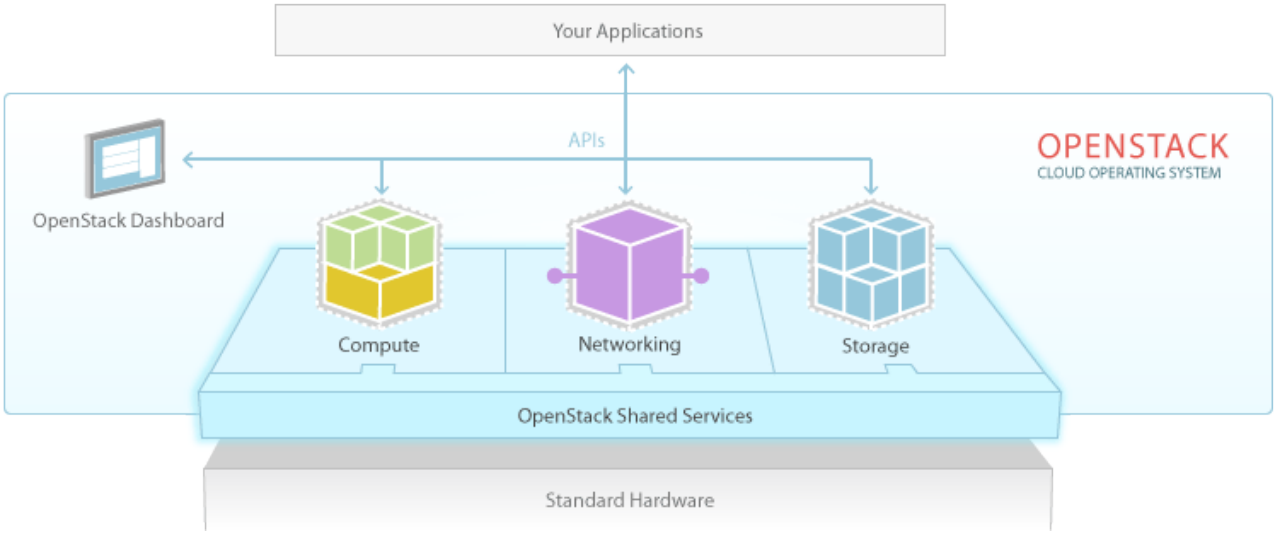

Figure 19. OpenStack Cloud Operating System. Reprinted from OpenStack (n.d.).

Figure 19 illustrates OpenStack, which is one option to be used as Virtual Infrastructure Manger (VIM). In testing point of view cloud operating system is a new layer introducing more complexity in the system, making trouble shooting more difficult, and giving new requirements for the test scope.

The users of cloud-management stacks have discovered the following fault resilience issues in OpenStack. For example, in a case of failure, creation of Virtual Machine may fail or take extremely long time. Despite of that, the VM creation may be marked successful although it is lacking critical resources (e.g. IP addresses) and remain for that reason unusable (Ju X. et al 2013:Ch 1). It seems that the OpenStack does not provide detailed and comprehensive specifications on system behaviors or state transitions for handling of external requests (Ju X. et al 2013:Ch 6). This may be a problem in a case of troubleshooting.

There were some issues found in the referred study (Ju X. et al 2013) that are related to OpenStack. These issues are summarized in the table below (Table 21):

Table 21. OpenStack related resilience issues. Data gathered from Ju X. et al 2013:9-12.

	Issue related to Resilience
1	Timeout mechanism that is used extensively by OpenStack in fault situations. Setting the timeout values for OpenStack and external supporting services is known to be difficult.
2	Periodic checking: to monitor service aliveness, to resume interrupted executions, clean up garbage, and to prevent resource leakage. Mostly related to creation of VMs.
3	State Transition: OpenStack maintains a large number of states in its databases, with indicated state-transition diagrams among them. Problematic state transitions have been experienced.
4	Return Code Checking: related to API and/or shell commands. E.g. in a case of shell command, the return code was not checked, it was just assumed to be successfully executed.
5	Cross-layer Coordination: OpenStack relies on various supporting services to maintain its functionality and supports interaction with multiple services in each external service category via a set of layers of abstraction. The coordination of these layers is complex, and incorrect interpretation of behavior of one layer may lead to faults in another layer.
6	Library Interference: The extensive use of external libraries in large-scale software systems may lead to unexpected library interference.

The issues listed in Table 21 need to be considered when testing resiliency. This kind of issues might be common within many cloud-management stacks. Many of them share a common high-level scheme, having similar service groups; they rely on the similar external supporting services, and have similar communication mechanisms.

Cross-functional vs. Supportive Teams

One of the biggest question in the ways of working in system testing has been: who should do the system test? The original aim in Agile methodology has been that all the testing would be done by the cross-functional teams. This would include also the system test. This has changed. Janet Gregory and Lisa Crispin wrote already 2009 (Gregory J. and Crispin L. 2009) about usage of the supportive teams in testing. They got even further with this thought 2014 (Gregory J. and Crispin L. 2014:284,285). They stated that one option when multiple teams are working on the same product is to use system test team. The system test team should also participate in a project inception or high-level planning at the start of the project. The main purpose of the system test team is to continually test the "potentially shippable" product delivered at the end of the each iteration, or possibly more frequently. This should be an integral part of the development process, testing the entire system end-to-end in a production-like environment, which may not be available to each individual team. (Gregory J. and Crispin L. 2014:284,285). The test belonging to such system test teams are mainly tests in Quadrants three and four in the figure below (Figure 20):

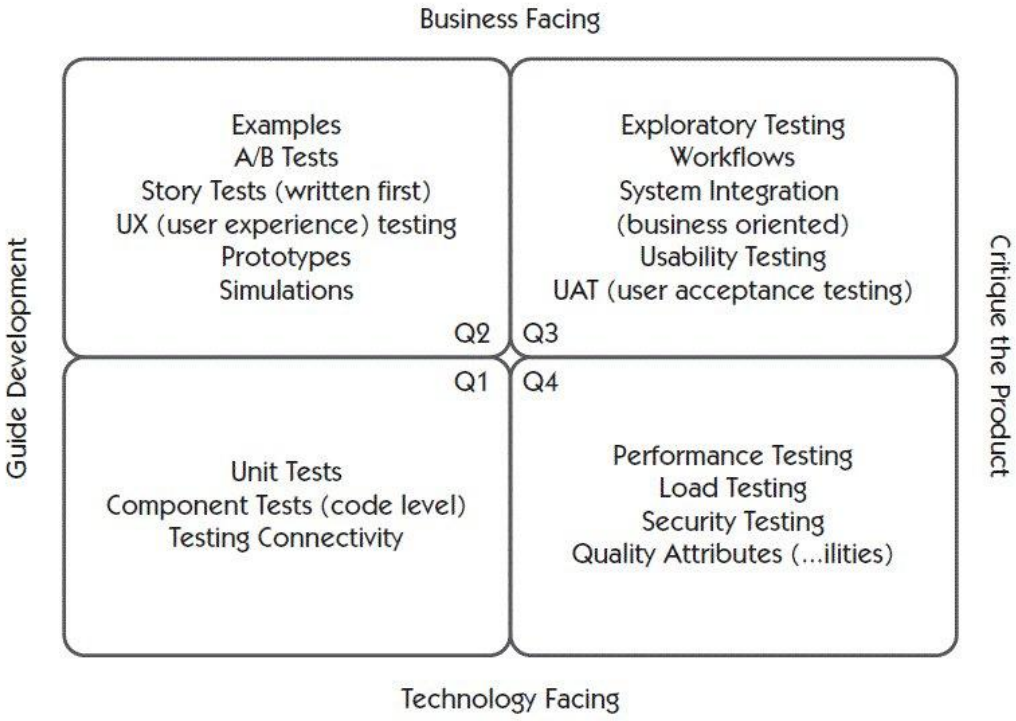

Figure 20. Agile testing quadrants. Reprinted from Gregory J. and Crispin L. 2014:102.

Figure 20 illustrates the new Agile testing quadrants, where the quadrants one and two represent tests that guides development teams, whereas the quadrants three and four represent tests that critique product, which are mainly the non-functional system tests.

Additionally, there are also other sharing the opinion of having separate system test teams. The world quality report, 2013-2014, follows the same thinking. This report is based on the market analysis and commentary on the state of enterprise application quality and testing practices from 1,500 interviews with senior executives across 25 countries. The report states that certain testing tasks, such as end-to-end integration, are best performed as a separate phase after the agile iteration, because these types of testing often demand longer preparation time with cross-domain interaction (Capgemini 2014:36-39).

The situation is cumbersome, on the other hand people are talking about having separate system test teams, then in the other hand people are talking about having combined teams, with even more tasks than in ordinary cross-functional team, having DevOps teams covering even customer support. Moreover all the opinions are very well reasoned.