UiO**:Department of informatics**
The Faculty of Mathematics and Natural Science

Project Report – 5th of november 2015

TeliaSonera Norge

ESPEN BERNTSEN - espenub

EIRIK LILLEJORDET - eiriklil

HÅKON ANTONSEN – haakoaan

TONY MATHIESSEN - tonym

WANQIU LIU - wanqiul

A look at the information infrastructure at TeliaSonera Norge

Table of Contents

1. Introduction

1.1 About the case

We have studied the information infrastructure at TeliaSonera Norway, Norway's second largest supplier of mobile services (TeliaSonera, n.d.). TeliaSonera Norway is the Norwegian daughter company of TeliaSonera AB (NetCom, n.d.-a), which is the fifth largest telecom operator in Europe (TeliaSonera, 2014). TeliaSonera AB has 26166 employees in 17 countries (TeliaSonera, 2014), and service 72.8 million mobile subscribers (TeliaSonera, 2014). In TeliaSonera Norway there are approximately 1400 employees (NetCom, n.d.-a), and 2.6 million subscribers. TeliaSonera is represented by multiple brands in Norway, including NetCom, Chess, OneCall and MyCall (TeliaSonera, n.d.).

1.1.2 Scope

Because of the size of the organizations of TeliaSonera and TeliaSonera Norway, we have decided to limit our study to the organizational branch called Customer Channels, and more specifically the Distribution Service B2C (DS) department for the lead brand of TeliaSonera Norway, NetCom (TeliaSonera, n.d.). As NetCom is a cell phone operator, and one of the leading operator brands in Norway, looking at the information infrastructure of the whole of the organization would be too time consuming for one semester. During our study, we have found approximately 40 different software systems that are directly necessary for DS at NetCom, and this number would increase a lot if we were to describe the systems used by technical support staff, developers, business analysts and so on. Not to mention the actual GSM system that runs Netcom's cell phone network.

1.1.3 Organization

We will now briefly describe the organization of TeliaSonera Norway, how it is placed in the organizational structure of TeliaSonera AB, and then focus on the position of DS.

TeliaSonera Norway is headed by CEO Charlotta Rehmann (NetCom, n.d.-b). Directly under her are the leaders of the different divisions in TeliaSonera Norway. Notably, Chess, OneCall and MyCall have their own subdivisions within TeliaSonera, with separate services like customer service and IT, while NetCom share these functions with the rest of TeliaSonera Norway (NetCom, n.d.-b). DS is located in the Customer Channels division of TeliaSonera Norway, which is headed by Ove Mathias Lind (NetCom, n.d.-b). DS is headed by Ronni Abusland, and is a part of a subdivision named Support.

DS is responsible for handling inbound traffic from Netcom's different dealers, including NetCom Shops which are stores that are a part of TeliaSonera Norway, and external dealers like Elkjøp and Expert, two of Norway's largest electronics retailers.

1.2 Method

We have based our data collection on interviews and observation. We have performed three interviews at TeliaSonera Norge, interviewing an employee working at DS, the leader of the Technical System Support for Customer Channels, and two employees in that department. We have also performed an observation of one of the staff members at DS, at a night shift. We first interviewed the employee at DS, to get an overview of what systems she deemed important in her work flow, and how she uses them. After this, we performed an observation of another employee in the same department, to try to get a more fulfilled list of the software systems that are used in this department daily, and to see if we could find any workarounds and such. Lastly, we interviewed the different staff members in TSS CC, to get an overview of the technical aspect of the infrastructure, the systems in use that were not visible to the end-users, and to get information about fall-back procedures and so on.

1.3 The Case Elements

In this section we will describe the elements that form the information infrastructure at TeliaSonera. We will go into some of the more important elements in some depth, while briefly mentioning other parts.

1.3.1 Sales Clients: Salsa, Tango and POS

Salsa, Tango and POS are the sales systems used by external dealers, Customer Channels at TeliaSonera, and internal dealers respectively. They are used to register orders which are subsequently processed by NEO (described later). Salsa is by itself exclusively an ordering system, that collects the information needed to update a subscription with the info NetCom needs in order to activate hardware and pay out provisions to the stores, while POS and Tango can also be used to take payment from customers in different ways. POS is the most comprehensive of the three, and is also used to keep track of a shops storage supplies. Salsa was until May 2015 the only sales system in use by Netcom's physical dealers, but was then replaced by POS with the internal dealers.

1.3.2 Fokus

Fokus is a software suite used by large parts of TeliaSonera Norway to administer a diverse array of tasks. It is developed by an external corporation, and bought as a commercial off-the-shelf product by TeliaSonera Norway. Fokus is a huge part of the total information infrastructure at TeliaSonera Norway, and is the main gateway from most other systems into the deeper parts of the GSM network. The amount of functionality and usage of Fokus is too large to describe here, so we will focus on the two applications that are mostly used at DS. Fokus Customer Care is the most used system at the whole of CC, and also at DS. This is where orders registered in the sales clients are eventually applied. Support staff at DS use this software for many of their daily tasks, such as looking up customer information, looking up subscriptions and billing information, and carrying out changes in all of these areas. It is also used to move subscriptions and phone numbers between users and organizations, registering and activating new SIM cards, adding or removing features and services and so on. There are few technical restrictions on what can be done in Fokus CC. Since Fokus is also the gateway to the GSM systems (like HLR and CNDB etc.) changes done by support staff here are pushed into the network instantly. The other application that is used by DS, is Fokus Inventory Control, where SIM cards can be (re-)opened for registration.

1.3.3 Middleware: NEO, NSL, CDS and NoBill

There are different middleware layers between the sales clients and the customer database, and the GSM network. As described above, Fokus is the main gateway into both of these for most software systems. In addition, there is another gateway for prepaid subscriptions, called NoBill. This system handles functionality specific for prepaid subscriptions, such as their balance, refills, etc. Functionality in common with post-paid subscriptions is taken care of by Fokus.

Directly below the sales clients there are two separate systems, NEO and CDS. CDS is a live data replication system, which contains an up-to-date replica of the customer database from Fokus. This is where most other systems get their customer data from. CDS data is updated on change of user information, as well as a full replication of the database twice a day. NEO is a collection of components used for order processing. Each of the components are more or less specialized for a small purpose during the order process. The largest component is called OrderManager, and handles every order after it is sent from the sales clients. Most of the business logic validation is done within NEO. The other components of NEO provide a mix of data validation, information and functionality. Examples of these are Payment Handler which handles different methods of payment, Price Master and Product Manager which contains information about prices, and validation of which products can be bought together. Information is passed between NEO components and other systems through APIs implementing either SOAP or REST.

The last middleware system is NSL, or Ninja Service Layer. This is the part of the system that orders are handed to after NEO, and which pushes the order into the Fokus database. NSL is an emulated Fokus client that works as a "clicker agent", emulating the steps that would have to be done by an employee to enter information info the system. There is also some order validation done in NSL.

1.3.4 Backend GUIs: Copa and Prepaid Customer Care

There are two associated GUIs to two of the middlewares. NEO (or more specifically OrderManager) has a GUI called Copa, where one can view orders that are being processed, and have previously been processed. There are different views where it is possible to view which changes are associated with the orders, who ordered it, which subscription it is for and more. It is also possible to only view failed orders from different channels etc. Orders can be cancelled or reprocessed, however after an order has been sent into NSL it cannot be fully cancelled from Copa and the introduced changes must be manually removed from Fokus.

NoBill has a GUI called Prepaid Customer Care, where prepaid subscriptions can be viewed, along with a limited set of customer information. It is also possible to view the prepaid balance, and do refills or balance adjustments.

1.3.5 Queue Processing: CallGuide and eTray

CallGuide and eTray are the two main systems used by staff at DS, and also all of Customer Channels to process their queues. CallGuide is used for call queues and chat, while eTray is used for written queues.

CallGuide is developed in-house by TeliaSonera AB, and used as the call centre software in most of TeliaSonera's subsidiaries. From the perspective of a DS employee, it is a system that allows them to log into different call queues, accept calls, and transfer them to other departments. In the backend, CallGuide is also a routing system, where the customer database CDS is replicated each night, and a routing table is generated for every customer of NetCom, deciding which queues they should be sent to. This means that business customers are routed to the business call centre, private consumers are sent to the consumer call centre, and so on. DS also use CallGuide for chat functionality.

eTray is an off-the-shelf software used to process written queues, which means fax, email and letters. It was acquired to replace an old system named EDKS, that was previously used to process letters and faxes, and to take the email queues out of Microsoft Outlook. There are different queues for different inquiries, which are "owned" and processed by different departments. Faxes and letters are scanned on a dedicated scanner and stored on a file server. Faxes and letters are stored as image files, while emails are stored as message files. eTray also contains a web interface that can be used to view cases, and also register cases to a few queues, such as ordering SIM cards or bill reprints.

1.3.6 Other support systems

The systems described above are the main systems used at DS, but there are a host of other, smaller and to a certain degree more specialized systems that are also in more or less daily use. In our data collection we found more than 40 different systems used only at DS. They range in size from trivial to large, but common for all is that they are only used to a small degree by the staff at DS. There are a few worth mentioning briefly; TeleOpti CCC MyTime is used by managers to enter information about when employees are supposed to work, and also used for planning staffing. Employees use it to view their shifts. The Microsoft Office suite of applications are used to varying degrees, especially Outlook which is used for communication across departments and with customers, and Lync used between colleagues. KSWiki is a web based wiki where information about routines for customer channels reside, as well as information about price plans, subscriptions etc.

On top of these, there are small applications like standalone credit checks, iPhone unlock software, Spotify support tool and a tool for checking BankID.

1.3.7 The case focus: Distribution Service B2C

Distribution Service B2C consists of a team which purpose is to service NetCom's dealers for private consumers. It was formed during a recent reorganization of TeliaSonera Norway where the earlier division Dealer and Channel Support was split into Distribution Service B2C and B2B, where the latter now services business dealers, and Dealer and Channel support used to service both. The department consists of a team leader, with approximately 10-15 employees. The employees work shifts, to make sure that the phone and chat queues that are owned by the department are manned throughout the opening hours. The shifts are divided into three categories; day shift, afternoon shift and weekend shift. Depending on the expected traffic, there can be between 10 and 3 people at work at any given time. A typical work day consist of 5 to 6 hours of accepting phone calls, as well as 1 to 2 hours servicing the chat queues. In addition, the employees work on queues in eTray in-between calls. Most software systems are used during calls from dealers, to either look up information, or make changes. A typical use case is when a dealer wants to upgrade a subscription that is currently in a commitment period. The DS employee then needs to look up the number in Fokus, navigate to the subscription page, open the commitment dialog, and remove the commitment. In addition, the employee needs to check that there are no leases or commitment services on the subscription, and in that case remove them as well. There are business rules for when a commitment can be removed, these are however not validated through Fokus, so it is up to the DS employee to make sure that the operation is allowed. Lastly, after all changes has been made, the dealer needs to reload the subscription she is changing in her sales client, and enter the information needed.

1.3.8 Dealers

The dealers can be roughly grouped on two axes, consumer/business and internal/external. The internal dealers are stores owned by TeliaSonera, and there are two of these. There is *NetCom Bedriftssenter* (English: NetCom Business Centre) on the business side, and NetCom Shop on the consumer side. DS only services the consumer brand, and Distribution Service B2B services the business brand. The NetCom shops are the biggest brand DS has to deal with in terms of inquiry volume. They are also currently the only brand using POS instead of Salsa.

The external dealers are stores that have an agreement with TeliaSonera to sell their products. Some of these stores sell only consumer products, some sell only business products, and some sell both. The dealers in this category on the consumer side are mostly large electronics warehouses that stock phones and sell subscriptions for most of the major cell phone carrier brands in Norway. They all use Salsa as their order registration system.

There is one last category, which consists of dealers only handling prepaid subscriptions. These include some kiosks and grocery stores, where consumers can buy prepaid "starter kits", as well as refills for their prepaid cards. What sets these dealers away from the others are the fact that they do not generally deal with technology, and that they only sell prepaid products. None of these has access to Salsa or any other order systems, and if customers have problems or questions regarding their subscriptions, the dealers will have to call DS, regular customer support, or send the customers to a nearby dealer with access to Salsa or POS.

1.3.9 Technical System Support

Technical System Support at Customer Channels is the department concerned with the day-to-day support for the systems used by Customer Channels. They handle inquiries from all departments within CC, and try to find and fix problems if they can. If not, they are responsible for escalating problems to the right departments. TSS is responsible for supporting most of the systems described in this assignment, such as Salsa, POS, Tango, Fokus, eTray, CallGuide and so on.

1.3.10 Other stakeholders within TeliaSonera

There are more stakeholders directly involved with the II we are looking at within TeliaSonera. Many of the systems are developed in whole or in part in-house, and the development teams for each of these plays a large role in what gets implemented, how and in what way. The components developed in-house include, but are not limited to, Salsa, Tango, POS, and most of the components in NEO. The development teams get input on what needs to implemented in part by the employees using the systems, but also largely they come from the business division of TeliaSonera Norway. The business division will create business rules, price plans, bundles and other sales and customer related requirements that the developers need to address. The different teams are also stakeholders for each other in the sense that new features in for example Salsa requires changes in some NEO components, and changes to NEO components can propagate into other NEO components or other parts of the system.

1.3.11 External stakeholders

Some of TeliaSonera's systems are developed in part or wholly by external consultants or companies. Fokus and eTray are examples of two systems that are developed in full by external companies. For these, the users have little direct influence over which features are implemented, as the systems are developed for a large group of users, not just TeliaSonera. In the case of Fokus, the company that delivers the software have some consultants placed in TeliaSonera Norway's building, working with implementing changes and configurations that are requested, within the confines of the COTS system.

Another important external stakeholder is the Norwegian government, most clearly represented by the Norwegian Communications Authority (Nasjonal Kommunikasjonsmyndighet, or NKom) called Post and Telecommunications Authority (Post- og Teletilsynet) until 2015. This government authority develop requirements and guidelines on what is allowed, what is not allowed, and what is required within the telecom business. Together with the national laws of Norway, they form a significant constraint on how systems should be implemented (e.g. security measures etc.) and how they can be used (not distributing customer information etc.).

2. Theoretical ground

2.1 Defining an Information Infrastructure

Information Infrastructures are characterized as enabling, shared, open, by being heterogeneous and having an installed base as defined by Hanseth and Monteiro (1998a). We will now describe how we understand and define these terms in our report.

2.1.1 Enabling, shared and open

According to Ciborra et al. (2000), the *enabling* function of an information infrastructure comes from the fact that it is designed with many different tasks and activities in mind, instead of being designed to support only one, or a small set of tasks. They continue to write that an II is shared because it is used by a collection of users and user groups. Different users are not seeing different systems; they are using different parts of the same system. The reason we can view it as an infrastructure of systems rather than a collection of separate systems, is because the different systems used by different users are dependent on each other, and thus cannot be used separately, or independent of the others. Lastly, Ciborra et al. (2000) describes that IIs are open because there are potentially an unlimited number of stakeholders, users, systems and components (and so on) in the II.

2.1.2 Heterogeneity

IIs are not merely hardware, software, and technicians, "they are rather socio-technical networks, concerning the qualities of their constituencies" (Ciborra, 2000, ch. 4). This means that one has to account for the people interacting with the system in various ways. These stakeholders have different interests and expectations when designing, using and managing the systems (Hanseth and Monteiro, 1998a). Their competency also varies, and this reflects the way the II is designed, used and learned. Infrastructures are intersected and overlapped. The different separate components interact with each other by logical relevance.

Heterogeneity of Infrastructures also embodies that diverse solutions might be used to fulfil the same functionality. Therefore, in one II there might exist different versions of protocols or standards.

That IIs are heterogeneous also means that they evolve upon existing small parts of a system and then the parts originally irrelevant become interlinked and alter with any changes to one of them.

2.1.3 Installed base

"Infrastructures develops through extending and improving the installed base" (Ciborra, 2000, ch. 4). Innovations of IIs are based on an existing one. They evolve over time to satisfy newly emerged requirements or changes.

In conclusion, an infrastructure is an evolving shared, open, and heterogeneous installed base (Ciborra, 2000, ch. 4)

2.1.4 DS as an infrastructure

Based on our data gathering, and the definitions and terms we find in literature about IIs, we can define the systems at DS as an information infrastructure. The collection of approximately 40 different systems, consisting of numerous sub-modules and services, used by a wide range of users with different needs and use cases, makes this a complex system. It is a heterogeneous system because there are many different users, in different parts of the organization, and many different systems based on differing technologies that work together to form the II. It is enabling in the sense that the II and different software systems are used in new and different ways by different organizational bodies, and some of the systems, such as NEO, was developed specifically to enable development of new functionality that may be needed in the future. It is shared because of the multitude of users, stakeholders, and departments that use the II for different things in different ways. It is open because there is essentially an unlimited amount of users, systems, software, and other stakeholders in this II. But, it is open to a user as long as one has the right competency to use it. New staff in the DS must have two weeks of intensive training. Finally, the II consist of a large amount of installed base, that has to be taken into account when changing anything.

2.2 Theoretical framework

2.2.1 Actor-network theory

When trying to make sense of the information infrastructure that is DS, we wanted to look at it through the perspective of the Actor Network Theory. As we have seen in previous sections, information infrastructures are composed of more than machines. Hence, the users of an II are just as vital as its technical components. However, there are more than just the users and their interests that influence the makeups of an infrastructure. Our society consists of "a highly heterogeneous network of actors, institutional arrangements, textual descriptions, work practices and technical artefacts" (Hanseth and Monteiro, 1997). In an infrastructure, every entity influences other entities, to the extent of determining, or constrains, their purpose and potential.

This is where Actor Network Theory (ANT) is useful. It provides an increased level of detail in analysing how the concrete mechanisms of socio-technical elements interconnect and grows into a network. Thus, it grounds the description and analysis of problems regarding flexibility and standardization (Hanseth and Monteiro, 1998a). By large we build upon Hanseth and Monteiro's minimalistic approach to ANT when looking at our case, as they bring about a comprehensible view of the theory.

2.2.2 Actors

To act means to behave, or *do something* (Cambridge University Press, n.d). And in an II everything, both technical and non-technical, does *something*. And this something often influences other's behaviour. The use of a software program on a computer is influenced by prior experience with its functionality, as well as the functionality itself. (Hanseth and Monteiro, 1998a) mention the act of driving a car to illustrate, and it serves rather well: traffic regulation, other driver's behaviour, your driving experience, the car's manoeuvring abilities, and so forth, are all factors that relate and connect to how you act in traffic. "An actor network, then, is the act linked together with all of its influencing factors (which again are linked), producing a "network" (Hanseth and Monteiro, 1997, p. 97). This is the heterogeneous nature of ANT: both technical and non-technical factors that influence behaviour should be considered together, and everything can be an actor, as long as it defines in some extent a pattern of action on other actors.

2.2.3 Translation and inscriptions

There are two main concepts within ANT: inscription and translation (Hanseth and Monteiro, 1997). Inscription describe how entities in an infrastructure constraint others by defining, or 'inscribing' a certain way of use. Inscriptions can be technical and non-technical, and at different strengths of succeeding in enforcing a desired behaviour (Hanseth and Monteiro, 1997, p. 184). The inscription comes from translating actors interest in the entity itself. Callon (1990, p. 143) illustrates the notion of translation by stating, "'A translates B'. To say that is to say A defines B". By this, he means that A is the *materialization* of B. He further explains, "A acts depend on past translations. These may influence what follows to the point of determining them". According to Actor-network theory, the translation takes place when the negotiated interests and expectations of various actors are aligned by designers, ant turned into a scenario (or 'standardized'). The standard is then inscribed, or embodied, into materials, such as texts, machines, or bodily skills (Callon, 1990). This inscription includes a "program of action"; a predefined role for the users and system to carry out. This also means that the designers presume a certain level of competency of the users, as well as connected systems.

The strength of the inscription depends on how easy it is to work around them. By adding inscriptions, they accumulate strength. Once inscribed, the material becomes an actor on its own with its program of action, thus having interests of its own that need to align with other (technical and non-technical) actors. In this way, an information infrastructure is therefore composed of a "collection of standards and protocols, user expectations and experience, bureaucratic procedures for passing standards" (Hanseth and Monteiro, 1998a, p. 100). ANT is therefore a bottom-up concept, where “things” need to align in order to ensure workflow and stability within the information infrastructure (Ciborra, 2000, ch. 5).

2.2.4 Discussion/DS as Actor Network Theory

In this section, we will compare the key concepts of ANT to some of the characteristics we have unfolded in our data gathering. The flexibility we gain in terms of scope help us analyse the complexity later on in our project.

First, we justify the use of ANT by confirming that it is in fact material actors, technical and non-technical, influencing each other, creating a network, as according to the definition (Ciborra, 2000, ch. 5). As mentioned in the introduction, we found that DS has approximately 40 different software systems in close relation to the human actor. And with that amount of software we want to use ANT´s ability to zoom in and out of a complex context of technical and non-technical actors to describe it coherently (Ciborra, 2000, ch. 5). Another key aspect that relates to the amount of software is that the DS infrastructure is typically not a result of top-down management, but rather a bottom-up approach, which is also the tendency in ANT (Ciborra, 2000, ch. 5).

ANT describes change, or the ability of change in the information infrastructure as translation and inscription. An example from one of the systems described in the next section of the text is how the change of one system, Salsa, can create a new inscription, a new way of acting, which in turn change how the actors act with the network.

As we dig deeper in DS´s information infrastructure, we will most likely unveil many examples of translation and inscription, and how they will affect the future change of DS´s II.

[This concludes the end of delivery 1.]

3. Designing changes in Information infrastructures

3.1 Overview of previous changes in TeliaSonera Norway's II

In this chapter, we will discuss a potential change in the information infrastructure, the strategies used to change it, and identify challenges related to the process of change.. First, we outline the recent changes and the reasons behind them from the developer's perspective. Then we will present theoretical strategies of change as concepts before we apply them to our case of potential future change, which is the change of Fokus. In the last part of this chapter we will evaluate and discuss the strategies and discuss alternatives to these.

3.1.1 Outline

An outline of recent changes and the reasons behind them might help us understand the II´s potential future change. The infrastructure, as described in chapter 1, give us an overview of how it looks like and how the different systems are interconnected, but does not explain how it came to be the way it is, and what thoughts are behind the development of newly added or redesigned systems. We will describe three different, but important systems that have been recently changed or have the potential to change.

POS was, as mentioned in chapter 1, implemented in May 2015, and is the most recent change in the II. Before POS was developed the physical NetCom shops were using two systems for sales - Salsa and Navision. Salsa handled registering the order in TeliaSonera´s system while Navision handled the payment. The motivation for developing POS was to get the functionality of Navision and Salsa in one system. POS has a lot of attributes that was able to replace Salsa internally, with the ability to take payment and updating the storage, as well as meeting the users demand of only having to input order data once.

As one of the middleware clients, NEO was uniquely developed with change in mind, making it easy to add, adjust or remove components from its list of features. Since it is right below the sales clients, it has to be able to adjust to changes in the order handling, prices and payments. A possible, near future change in NEO is the redesign of its largest component, OrderManager, to follow the idea of having small components that is specialized to small tasks, making it more flexible. The emphasis on flexibility is realized through using REST or SOAP APIs, making NEO capable of having massive impact on the way the system is interconnected. Around the time of NEO's implementation, the developers had already begun playing with the idea to phase out Fokus as the main database.

3.1.1 Shift of Fokus

The change of Fokus will alter both the technical and socio-technical aspect of the II. This makes the trajectory of the II very interesting depending on how they want to realize that goal. Therefore, the rest of this chapter will mainly describe the challenges of eventually replacing Fokus and strategies on how to do this.

3.2 Theoretical framework

As said in a previous chapter, ANT is concerned with the way the actors inscribe certain patterns of action, by implementing standards in materials. The theory is well suited for addressing how entities in a network influence each other, and to what extent change(s) in this system might influence respective actors, both technical and non-technical. However, it is not sufficient to describe *how* to design and execute the change(s). To address this we will focus on *bootstrapping* outlined by Hanseth and Lythinen (2010), and look at some of the design principles that derive from the kernel theory "complex adaptive systems" (CAS) (Holland, 1995 as cited in Hanseth and Lyytinen, 2010, Monteiro, 2000).

3.2.1 Bootstrapping

Bootstrapping is used in many contexts with slightly different meanings. In general terms, it emphasizes how you can develop something through initiative with little or no assistance. Bootstrapping is built on a bottom-up, evolutionary approach to develop an II (Hanseth and Aanestad, 2003a). Thus, bootstrapping is self-reinforcing installed base through network effects of an increased user base. In terms of inscribing a standard that defines an IT capability, when it gains users, so does its value. This will probably result in more users adopting it, and so forth (Hanseth and Lyytinen, 2010, p.6).

As more and more users adopt the system, or the design decisions shape the system, the infrastructure develops towards a certain direction. Some design decisions can have irreversible effects, making the system dependent to its current trajectory (Hanseth and Lyytinen, 2010). Two components; cumulative growth and technology traps distinguish this *path-dependency*. When an infrastructure expands, new design and functionality requirements that has to fit in with the rest of the components, both technical and social, will eventually arise (Edwards et al., 2007 in Hanseth and Lyytinen, 2010). Design decisions made early on to satisfy the initial users could then prove to become *technology traps* when later changes are necessary (Hughes, 1987). Early design decisions are often socially and technically based on early installed bases and predicted development. The result of this is that the early design may also confine further expansion down the road as new actors enter the infrastructure, changing the trajectory of the infrastructure (Hanseth and Lyytinen, 2010).

The expansion driving the path-dependency will eventually lead to a *lock-in,* as the system's expansion reaches *momentum*. Huges (1987) defines this as when the system has reached a point of no return. This creates a new lasting stability with irreversible effects.

3.3 Analysis

Through our interviews, we have identified two factors motivating the need to evolve Fokus, and eventually remove it as it is today. Firstly, the way Fokus works today it provides a risk for human errors when handling customer related actions. In addition, there are almost no restrictions to what a user *can* do. This concerns the social aspect of the infrastructure. Secondly, Fokus is not a very adaptable and easily maintained system; issues that provides a technical based motivation for change.

3.3.1 Prone to human errors

Fokus is the main tool involved in managing customer actions through a suite of applications, the most used being "Fokus Customer Service Management". NEO, the service layer directly beneath the sales clients, tells Salsa and POS what services are available for the customer. However, these "business rules" are also separately implemented in NSL, the Fokus emulator, despite being the same. If they do not match the user will get an error, and the intended action fails to complete. This would not be a problem, had it not been for the fact that there are almost no restrictions to what the user *can* do (within the limitations of functions embedded) in the suite of applications. To give an example:

If a DS employee enters a subscription in Fokus for a business customer that is not included in the corporation's call plan agreed on paper with NetCom, and the customer later tries to make a change to any other subscription on the same business plan in a retailer store, it will not work. This provides a security concern, as every staff has access to things that NetCom does not want them to have, like changing credit limit, credits and billing.

We have here identified a profoundly low strength in the inscribed patterns of action, as they are based on training and written rules, not implemented as a constraint through the system (Monteiro, 2000).

3.3.2 Need for local flexibility

Although such a flexible standard is good for correcting errors, NetCom's leadership would rather have the risk for errors and misuse reduced. The setting of commitment period on a call plan also needs to change. Currently, this can be set to just about anything by the push of a keyboard key. Per rule, it cannot be set to zero unless it is less than a month left. This is often the case when a customer wants to end the subscription or downgrade. Despite company policy, local management gives permission to not follow this rule in specific cases in order to make sales and handle bad customer cases. There are also national regulations stating that private consumers cannot have a commitment longer than 12 months, and business customers no longer than 24 months (Ekomloven §2-4). These restrictions are not in Fokus either.

However, if the intentions of NetCom's leadership results in the removal of this feature, it is still important to be able to change the value of commitment period when an error *actually* occurs. External and internal dealers cannot do this, but DS can. We do not deem it wise to move the functionality over to dealers, as this would increase the risk of misuse and possibly harder to inscribe the policy. This is a case of bottom-up need for flexibility, and top-down need for constraints, where different actor's interests need to align (Monteiro, 2000, Hanseth and Monteiro, 1998b).

3.3.3 High dependency rate

In addition to the cases described above, the *dependency* on Fokus offers an issue of social importance, but also technical ones, should it break down. If Fokus fails, the staff at DS is unable to do most of their normal tasks. There are some possibilities for ad-hoc strategies like manual workarounds (Gasser et al., 1986), but the options are limited; most things cannot be done subsequently, for example entering upgrades to the call plan. In addition, the retailers are often not interested in calling back an hour later to get an answer to their questions, because then the customer is usually gone. A crash would also affect the IT-department in NetCom, who would have to prioritize this over other tasks.

3.3.4 Low adaptability - Technology traps

On the technical side, Fokus is also a gateway between NetCom's infrastructure and the GSM network, and it estimates phone usage for billing as well. Since there is no API, it is hard to integrate new products and services. This suggests that this region of the infrastructure has been designed into a *technology trap* (Hughes, 1987).

While NSL only emulates a Fokus client, it is also hard to do maintenance as well. Changes or adjustments in Fokus need to happen in NSL and vice versa. The low adaptability and high dependency rate calls for decentralizing the functionality in Fokus.

3.4 Discussion on design strategies and principles

As with most changes to an information infrastructure, the solution to the problems would not come without cost. The issues illustrated in the previous chapter fits nicely into the two design problems mentioned by Hanseth & Luthinen (2010): the bootstrap problem and the adaptability problem.

3.4.1 Actor-Network implications when designing the II

Today, the staff has individual access restraints to the Fokus suite. This means that the manager in charge governs every restriction manually to each user. Category access instead of individual access could be a way of handling the issue of commitment period and errors in subscriptions. However, inscribing a standard like this, an access gateway of sorts, can provide problems to the actors directly and indirectly to the system if not designed correctly. This is because there is not always a manager or assistant manager at work during DS opening hours, and never in the weekends.

Therefore, DS will not be able to help as quickly as they can. This is where ANT provides an useful image of the effect when changing something so "small" as for instance how to set the commitment period; one actor (leadership) changing another actors (DS staff) inscribed pattern of action that forces a change in when actors (DS needs to be present (Monteiro, 2000). In other words, it would force the business to change the work schedule in order to accommodate the new practice of (for instance) changing commitment period to expedite the need of retailers and customers. It can be hard to get people on board with this change. The question then, remains; how can we align the interests of these heterogeneous stakeholders?

3.4.2 The bootstrap and its problem

Although the decision for change is top-down, the implementation needs to be bottom-up to reach momentum (Hughes, 1987), and within here lies the problem; the information infrastructure needs to attract early adopters in order to *initiate* expansion of users. If the users do not accept the system, it will not be adopted. To address this issue, Hanseth and Luthinen (2010) suggest generating attractors that can bootstrap the installed base.

They propose that by building upon the existing installed base, and designing initially for direct usefulness, you can attract early adopters. By slowly *cultivating* the installed base, the implemented changes gives the users time to familiarize and adjust their attitude towards their new path of actions (Aanestad and Jensen, 2011). This work actually went in progress when NetCom released Tango.

Another way to attract early adopters is by generating "superusers" as attractors. Identifying highly motivated users (Hanseth and Aanestad, 2003b, Hanseth and Lyytinen, 2010), train them and use the network effect of their prior positive attitude towards the system will possibly make the transition easier. This strategy was utilized when implementing POS the summer of 2015, with good results.

3.4.3 The adaptability problem

Another challenge that needs to be addressed is the adaptability problem. In short, this addresses the issues of technological traps and early lock-ins. As mentioned previously, one can argue that Fokus is subject to low adaptability. Hanseth and Lyytinen (2010) provides two design principles to cope with the adaptability issue.

They claim that the first one should "make the organization of IT capabilities simple" This means using simple architectural principles and making the design simple (Hanseth and Lyytinen, 2010, p. 13). It is also important to keep the socio-technical elements in mind, and by relating to Actor Network Theory it can be achieved by keeping the actors loosely connected and avoid clashing head-on with other networks (Hanseth and Lyytinen, 2010).

The second design principle proposed is to "modularize the II" – which essentially is to keep loose coupled sub-infrastructures making separate sub-infrastructures for application, service and transport. This also applies vertically – making side-by-side application infrastructures connected by gateways (Hanseth and Lyytinen, 2010).

Although an entirely different case, it is worth pointing out how well the airline company Norwegian has modularized their entire information infrastructure. Within a short amount of time, they can remove or add functionality (modules) to the network without compromising other modules, because of the loose coupling between them. This is possible by connecting the modules to a "bus". Here we see how early design decisions have gained momentum, but without falling into a technological trap. This tells in favor for a modularization of an infrastructure, but it must be done almost from the beginning. NetCom would have to discard the current installed base and start over, in some equivalence to Hammer's (1990) revolutionary reengineering.

3.4.5 Irreversibility - How to make the inscription strong enough?

However, it is not enough to inscribe a new pattern of action, unless its not strong enough to hold. This forces us to shift the focus from bootstrapping and look to ANT again. To determine the strength of the translation we can look at Callons (1990) concept of irreversibility. Callon identifies two states which can be used to determine how "irreversible" a translation is. The first being to what extent it is possible to go back to a point where the translation is just another in the masses. The second is to what extent the translation dictates subsequent translations. When talking about switching out Fokus for another system this would be a huge inscription from the technical side, but it would probably also have a huge impact on the users as they would have to learn a new system. When looking at the first state introduced by Callon, this translation would make it impossible - at least for the users - to go back to the "old ways". The management could decide a roll-back, but we will take for granted that the systems have been tested from the technical side. When considering the second state - this translation would also dictate subsequent translations. One can argue that a modular system with an API would make later inscriptions easier to cope with, at least when looking at it from a technical perspective.

4. Concluding remarks

Through our analysis and discussion above, we consider the replacement of Fokus as a major change in the information infrastructure, which will have large side effects. Changes will propagate through the information infrastructure, in both technical and social aspects.

The case of Fokus is a good example of a domain where the need for control meets the need for flexibility. TeliaSonera management wants to control what is allowed and what is not with regards to subscriptions, services, commitment periods and so on. On the other hand, there is a need for flexibility in the Customer Channels Service department, to be able to give fast and efficient customer support. Fokus is great in the sense that it enables the service staff to help with most problems, however, it places a large amount of responsibility and autonomy on individual staff members, and diffuses upper managements control possibilities to creating written and oral routines, that it is up to local managers to inscribe into their employees. This does not always work, as with the example of the commitment periods.

TeliaSonera Norway has a problem with lock-in when it comes to Fokus. Routines, software systems and staffing policies are all based on an environment enabled by Fokus.

We believe that the strategies and design implications we have discussed above will be a good starting point in solving these problems. The concept of ANT has also allowed us to view some of the consequences of changing the system.

4. Bibliography

ANTONSEN, H. A. V. 2015. Observation at NetCom Dealer and Channel support.

CALLON, M. 1990. Techno-economic networks and irreversibility. *The Sociological Review,* 38**,** 132-161.

CAMBRIDGE UNIVERSITY PRESS. n.d. *act Meaning in the Cambridge English Dictionary* [Online]. Available: http://dictionary.cambridge.org/dictionary/english/act [Accessed 14.10 2015].

CIBORRA, C. U. 2000. *From control to drift : the dynamics of corporate information infastructures,* New York, Oxford University Press.

EKOMLOVEN Lov av 01.10.2015 om elektronisk kommunikasjon.

EMPLOYEE AT DCS 2015. Interview about systems in use at Dealer and Channel support, NetCom. *In:* BERNTSEN, E. U. & ANTONSEN, H. A. V. (eds.).

GASSER, L., HEWITT, ZDONIK & ALLEN, C. S. R. 1986. The integration of computing and routine work. *ACM Transactions on Information Systems (TOIS),* 4**,** 205-225.

HAMMER, M. 1990. Reengineering Work: Don't Automate, Obliterate. *Harvard Business Review*.

HANSETH, O. & LYYTINEN, K. 2010. Design theory for dynamic complexity in information infrastructures: the case of building internet. *Journal of Information Technology,* 25**,** 1.

HANSETH, O. & MONTEIRO, E. 1997. Inscribing behaviour in information infrastructure standards. *Accounting, Management and Information Technologies,* 7**,** 183-211.

HANSETH, O. & MONTEIRO, E. 1998a. Understanding Information Infrastructure.

HANSETH, O. & MONTEIRO, E. 1998b. *Understanding Information Infrastructure*.

HANSETH, O. & AANESTAD, M. 2003a. Design as Bootstrapping. On the Evolution of ICT Networks in Health Care. *Methods of Information in Medicine,* 42**,** 385-391.

HANSETH, O. & AANESTAD, M. 2003b. Design as bootstrapping. On the evolution of ICT networks in health care. *Methods of information in medicine,* 42**,** 384-391.

HUGHES, T. P. 1987. The evolution of large technological systems. *The social construction of technological systems: New directions in the sociology and history of technology***,** 51-82.

MANAGER OF TSS CO 2015. Interview with manager of Technical System Support, Customer Operations at TeliaSonera. *In:* BERNTSEN, E. U. & ANTONSEN, H. A. V. (eds.).

MONTEIRO, E. 2000. Actor-network theory and information infrastructure. Oxford: Oxford University Press.

NETCOM. n.d.-a. *Om TeliaSonera Norge | Netcom Mobil* [Online]. Available: https://netcom.no/om-teliasonera [Accessed 22.09 2015].

NETCOM. n.d.-b. *Organisasjonen i NetCom - NetCom Mobil* [Online]. Available: https://netcom.no/om-netcom/organisasjon [Accessed 22.09 2015].

SUPPORT STAFF AT TSS CO 2015. Interview with support staff at Technical System Support, Customer Operations at NetCom. *In:* BERNTSEN, E. & ANTONSEN, H. (eds.).

TELIASONERA. 2014. *TeliaSonera in brief - TeliaSonera* [Online]. Available: http://www.teliasonera.com/en/about-us/teliasonera-in-brief/ [Accessed 22.09 2015].

TELIASONERA. n.d. *Norway - TeliaSonera* [Online]. Available: http://www.teliasonera.com/en/about-us/markets-and-brands/norway/ [Accessed 22.09 2015].

AANESTAD, M. & JENSEN, T. B. 2011. Building nation-wide information infrastructures in healthcare through modular implementation strategies. *Journal of Strategic Information Systems,* 20**,** 161-176.