Framework to enable reliability analysis of SCADA solutions

Elforsk rapport 12:56

Stefan Svensson och Deborah Spira

September 2012

ELFORSK

Framework to enable reliability analysis of SCADA solutions

Elforsk rapport 21:56

Stefan Svensson & Deborah Spira

September 2012

Förord

Denna rapport är framtagen av projektet Tillförlitlighetsstudie för styr och övervakningssystem inom FoU-programmet Riskanalys 2010-2015 som drivs av Elforsk AB.

Programmets styrgrupp består av följande ledamöter:
Horst Blüchert, Elsäkerhetsverket (ordförande)
Kjell Oberger, Fortum Distribution AB
Håkan Jarer, Svenska Kraftnät
Eva Sundin, Vattenfall Eldistribution AB
Ola Ivarsson, E.ON Elnät Sverige AB
Jenny Paulinder, Göteborg Energi AB
Pär-Erik Petrusson, Jämtkraft AB
Sven-Åke Polfjärd, Föreningen Industriell Elteknik, FIE Remy Kolessar, Energimarknadsinspektionen
Susanne Olausson, Elforsk AB (programansvarig)

Finansiärer i programmet är:
Elsäkerhetsverket (ordförande)
Fortum Distribution AB
Svenska Kraftnät
Vattenfall Eldistribution AB
E.ON Elnät Sverige AB
Göteborg Energi AB
Jämtkraft AB
Föreningen Industriell Elteknik, FIE
Skellefteå Kraft Elnät AB
Öresundskraft AB
Umeå Energi Elnät AB
Jönköping Energi Nät AB
C4 Elnät AB
Gävle Energi AB
Härjeåns Nät AB
Sundsvall Energi Elnät AB
AB Borlänge Energi

Syftet med projektet är att ta fram ett ramverk av befintliga metoder som kan användas för att utvärdera drift- och övervakningssystem utifrån ett tillförlitlighetsperspektiv. Ramverket ska både kunna användas som stöd vid upphandlingar och vid översyn av befintliga system. Ramverket innefattar metoder för utvärdering av:

- Kommunikationslösningar
- Anläggningsteknik
- Hårdvara
- Mjukvara

Susanne Olausson
Programansvarig Riskanalys
Elforsk

Sammanfattning

Inom studien har ett ramverk för att möjliggöra tillförlitlighetsstudier av SCADA-lösningar föreslagits samt testats. Utvecklat ramverk baseras i allmänhet på två befintliga teorier för tillförlitlighetsstudier och i synnerhet på undersökningar av datoriserade system. Som ett komplement till de teoretiska studierna och testningen, har det i studien ingått workshops och intervjuer med representanter från elnätetsbranschen; dels användare av SCADA-lösningar och dels leverantörer av SCADA-lösningar.

Studien visade att föreslaget ramverk medför en genomförbar metod för att utföra tillförlitlighetsanalys på SCADA-lösningar. Branschen har varit positiva till de möjligheter ramverket medför, dvs möjligheten att få ett strukturerat arbetssätt samt analytiska och objektiva mått på tillståndet för en viss SCADA-lösning. Det finns dock fortfarande kvarstående utveckling för ramverket, både när det gäller att anpassa ramarna till verkliga förhållanden samt att utveckla verktyg som underlättar användningen av ramverket på verkliga system.

Summary

A framework to enable reliability studies of SCADA solutions has been proposed and tested within the project. In general, the framework is based on two existing theories for reliability studies and in particular, studies of computerized systems. The study has, as a complement to the theoretic work and testing, included workshops and interviews with representatives from the electrical network industry; both users of SCADA solutions and vendors of SCADA solutions in this context.

The study showed that the framework was a viable method to perform reliability analysis on SCADA solutions. The industry has been positive to the possibilities that the framework bring - enabling a structured work method and analytical and objective measures of the state of a specific SCADA solution set up. However there is still remaining work, both in terms of adjusting the framework to real-life conditions, as well as developing of tools that facilitates the use of the framework on real-life systems.

Innehåll

1 Introduction

1.1 Purpose

The purpose of this project is to evaluate if a framework can be derived that will enable analysis of SCADA solutions. The analysis, that the framework shall support, describes how alterations will affect the solution from different viewpoints. The framework shall support comparisons between multiple solutions from different vendors. The key feature of the framework will be the possibility to derive performance indexes, and how different alterations from different viewpoints can be compared. This will support governance for key aspects of a SCADA solution.

This report has been written by Sweco for Elforsk within the programme Risk- och Tillförlitlighetsanalys.

1.1.1 Attributes

Dependability and security properties of a system are described by a collection of attributes. The subsumed attributes of dependability and security as defined in the taxonomy are shown in Figure 1.

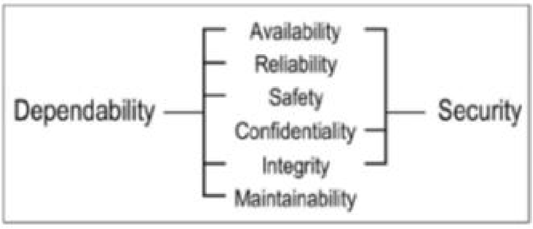


Figure 1 shows the attributes of Security and Dependability.

To further define the concepts of Dependability and Security the subsumed attributes are defined as the following attributes:

- **Availability:** readiness for correct service.
- **Reliability:** continuity of correct service.
- **Safety:** absence of catastrophic consequences on the user(s) and the environment.
- **Confidentiality:** the absence of unauthorized disclosure of information.
- **Integrity:** absence of improper system alterations.
- **Maintainability:** ability to undergo modifications and repairs.

These attributes are in focus when the aim is to analyse and improve dependability and security.

1.2 Hypothesis

This project has been undertaken with the hypothesis that it is possible to fully segment and evaluate a SCADA system as a whole by using the segmented parts. The basis for the analysis methodology has been basic probability theory, especially usage of the structure function.

2 Project methodology

The methodology used is influenced by agile software development. Agile software development is a generic term for a group of software development methods characterized by project teams collaborating and working self-organized and cross-functional. The main philosophy of these project management methods is to apply an iterative work flow that completes each task iteratively.

2.1.1 Reference group

An invitation for participation in the reference group was sent out in the initialization of the project. A total of six Elforsk member utilities accepted and have participated in the development of the framework.

The reference group consists of:

- Svenska Kraftnät
- Göteborgs Energi Elnät
- Fortum Distribution
- Borås Elnät
- Jönköping Elnät
- Vattenfall Eldistribution

2.1.2 Conceptual design

The conceptual design of the framework for system analysis has been derived through a systematic and iterative approach. An early idea of a unified approach to evaluate system designs from many different perspectives and phases.

2.1.3 Workshops

Multiple workshops have been performed as an iterative process to ensure that each extension and addition to the framework has been leading towards the defined goals.

Start-up workshop:

The purpose of the workshop was to derive the main goals for the project and framework. An outline for the project methodology was accepted and a basis for the continued collaboration was decided. Introduction of the key concept of the framework and some use cases was presented and discussed.

Participants at the workshop were representatives from the following utilities:

- Svenska Kraftnät
- Göteborgs Energi Elnät
- Borås Elnät
- Jönköping Elnät

- Vattenfall Eldistribution

Final presentation and workshop:

The purpose of the workshop is to present the final draft report and discuss the findings. The workshop will be performed on August 30th 2012.

Participants at the workshop were:

- Jönköping Elnät
- Vattenfall Eldistribution

During the workshop Sweco presented the project and the framework. In order elevate understanding of the framework and how it could be used an example of who an evaluation of a SCADA solution was presented. In this example the workshop participants performed each step in the evaluation process with assistance from Sweco. In short, the process contains these steps[1]:

1. Describe the SCADA solution with words
2. Select part of SCADA solution to be analyzed
3. Set up the model that describes the SCADA solution
4. Set up the structure function of the model
5. Defining reliability characteristics using the taxonomy for components in the model
6. Perform availability calculations for SCADA solution
7. Analyze the model to find possible improvement
8. Evaluate the proposed improvement, iterate from step 3

Discussions of the framework's pros and cons followed as well as discussions of the project execution.

2.1.4 Study of literature

Study of appropriate literature within the context of the proposed framework analyses will be conducted. List of references can be found in section 9.

2.1.5 The industry perspective – interviews and field studies

To complement theory with practice, vendors and utilities have been interviewed during the study. Main purpose with the interviews was to examine if and how the framework can be used in daily operation. The following vendors and utilities have been interviewed:

[1] The algorithm for the framework is describe in more detailed in section 4 and 5, in Figure 14 and Figure 15 this also outlined.

Utility / Vendor	Interviewee	Date
Svenska Kraftnät	Najib Mirkhani	April 17th 2012
Borås Elnät	Jesper Andersson David Håkansson	May 9th 2012
Fortum Distribution	Jan Olsson Hans Johansson Dan Kurtsson	May 10th 2012
Netcontrol	Jerker Forsblom Kim Malmberg	June 3rd 2012

3 Theory

The suggested framework is based on two theories. One theory is basic probability theory, see section 4.1 for more details, and the other is the taxonomy of security and dependability, see section 4.2. The suggested framework consists of combining the two theories in different ways. The following will describe the two theories in more detail.

3.1 Basic Probability Theory

System modelling is a basis for understanding dynamics of a chain. The graphical representation most suitable is the block diagram.

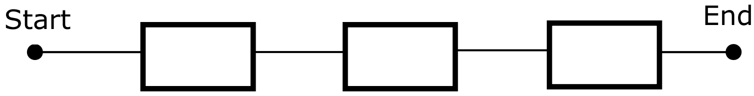


Figure 2 Serial structure

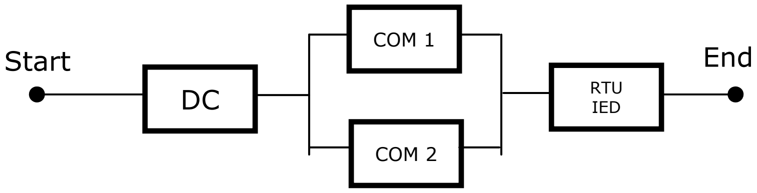


Figure 3 Parallel structure

The system model is described by the rate function:

Series structure: $\emptyset(\bar{\lambda}) = \prod_{i=1}^{N} \lambda_i$

Parallel structure: $\emptyset(\bar{\lambda}) = 1 - \prod_{i=1}^{N}(1-\lambda_1)\cdot(1-\lambda_2)\cdots(1-\lambda_N)$

For system models with n of N parallel coupled objects use:

$$\emptyset(\bar{\lambda}) = \prod_{i=1}^{n}(1 - \prod_{i=1}^{N}(1-\lambda_1)\cdot(1-\lambda_2)\cdots(1-\lambda_N))$$

To determine the state of the system function use:

Parallel couplings of min paths

$$\emptyset(\bar{\lambda}) = \left(1 - \left(\prod_{i=1} 1 - \prod_{i=1}^{n} \lambda_i\right)\cdot\left(1 - \prod_{j=2}^{n} \lambda_i\right)\cdots\left(1 - \prod_{k=n}^{N} \lambda_k\right)\right)$$

Availability is also a central concept of reliability and the functions for calculating availability are similar to those of reliability:

Availability of a series structure: $\bar{A} = \prod_{i=1}^{N} \mathrm{A}_i$

Availability of a parallel structure: $\bar{A} = 1 - \prod_{i=1}^{N}(1-\mathrm{A}_1)\cdot(1-\mathrm{A}_2)\cdots(1-\mathrm{A}_N)$

The opposite of availability is unavailability, which is described by the fault frequency multiplied with the mean down time, ψ;

Unavailability $=\lambda \psi$ Availability: $A = 1- \lambda \psi$

Note: The line connecting objects in the system model is indifferent, has no properties and only indicates that objects are connected to each other in the structure.

3.2 Taxonomy of Security and Dependability

The taxonomy, also known as classification, of Security and Dependability is a proposed support tool for system managers in the pursuit of a structured governance approach to system maintenance and development. Also, the taxonomy can be used by system architects, whom design or acts as an expert in requirement engineering.

The idea behind the taxonomy is that the analyst will be able to narrow the field of interest. Furthermore, a clearer view of the boundaries is attained with this approach.

The taxonomy and definitions used are based upon research and publication by IEEE in 2004. The article as a whole is listed in the reference list and is recommended for further information.

3.2.1 Faults

To be able to improve a system, correct measures need to be taken. However, defining the correct measures firstly requires that the problems needed correction, i.e. the fault causes, are defined.

There are multiple sets of faults that depend upon different viewpoints and leads to errors, which is exposed as a failure. IEEE discusses a set of viewpoints as shown in Figure 4. These viewpoints can be combined to stipulate different fault cause scenarios and be described as the tree structure, see Figure 5.

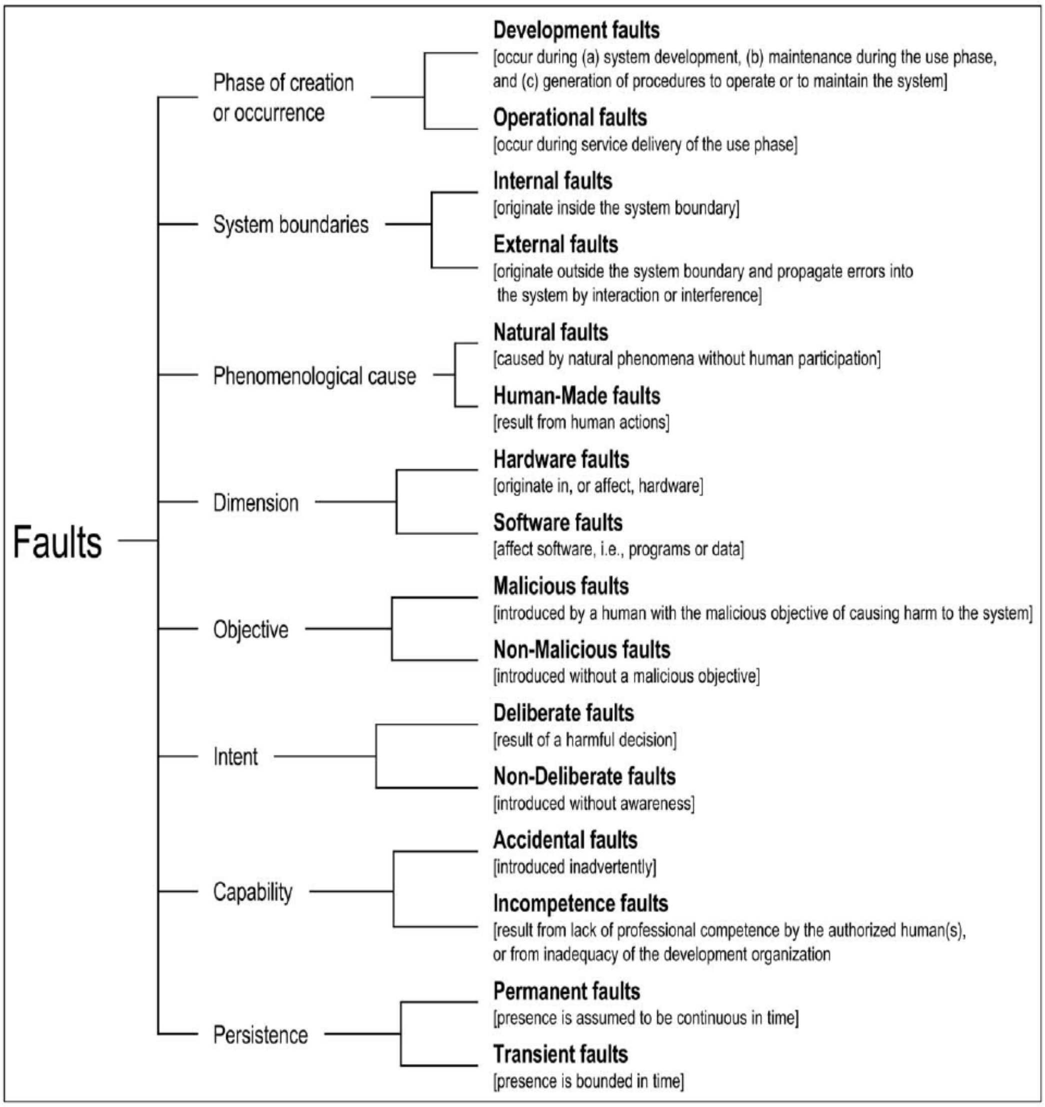

Faults
Phase of creation or occurrence
Development faults
[occur during (a) system development, (b) maintenance during the use phase, and (c) generation of procedures to operate or to maintain the system]
Operational faults
[occur during service delivery of the use phase]
System boundaries
Internal faults
[originate inside the system boundary]
External faults
[originate outside the system boundary and propagate errors into the system by interaction or interference]
Phenomenological cause
Natural faults
[caused by natural phenomena without human participation]
Human-Made faults
[result from human actions]
Dimension
Hardware faults
[originate in, or affect, hardware]
Software faults
[affect software, i.e., programs or data]
Objective
Malicious faults
[introduced by a human with the malicious objective of causing harm to the system]
Non-Malicious faults
[introduced without a malicious objective]
Intent
Deliberate faults
[result of a harmful decision]
Non-Deliberate faults
[introduced without awareness]
Capability
Accidental faults
[introduced inadvertently]
Incompetence faults
[result from lack of professional competence by the authorized human(s), or from inadequacy of the development organization
Persistence
Permanent faults
[presence is assumed to be continuous in time]
Transient faults
[presence is bounded in time]

Figure 4 shows the fault tree and the subsumed categories.

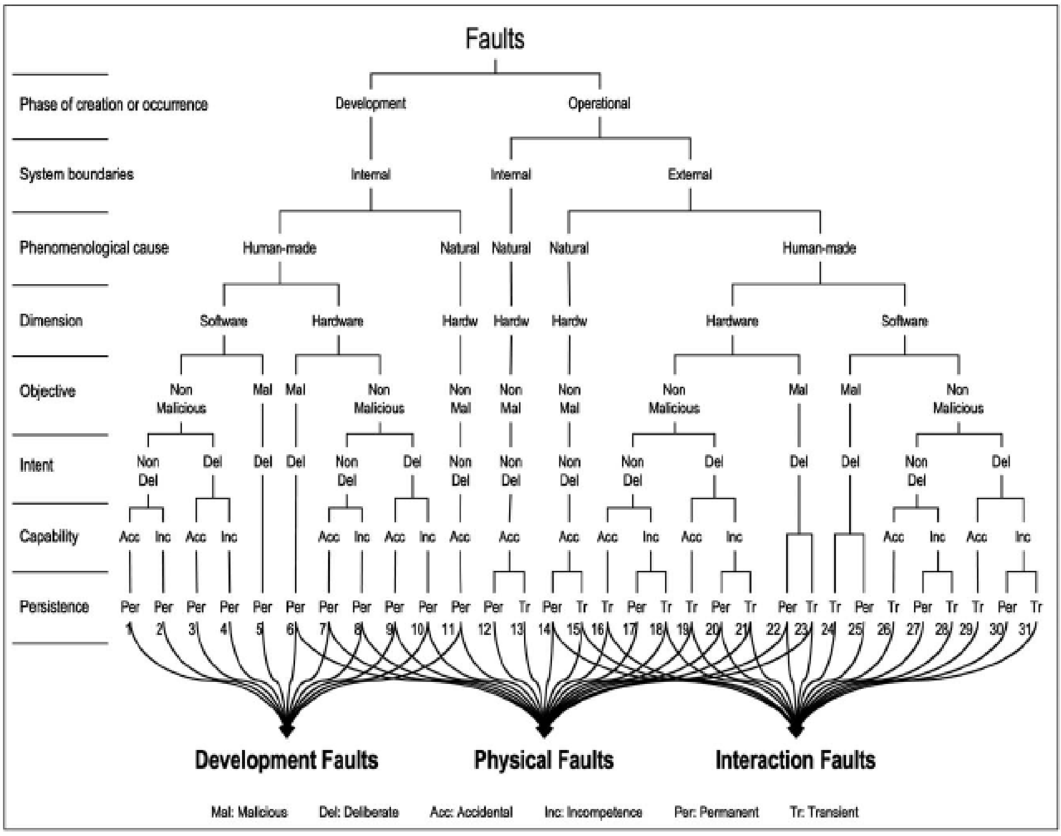


Figure 5 shows the combination of viewpoints for fault causes in tree form.

The tree structure in Figure 5 is comprised of eight levels and the end-points are grouped into three fault subsets; Development faults, Physical faults and Interaction faults. The tree structure is a good tool to determine the different faults that might occur.

For example the scenario *Operational – Internal – Natural – Hardware – Non malicious - Non deliberate – Accidental – Persistent* might hold the fault *fire due to overload* while the scenario *Operational – Internal – Natural – Hardware – Non malicious - Non deliberate – Accidental – Transient* might hold the fault *outage due to loss of power.*

Fault prevention

Preventing fault from occurring is either done in the system development phase or in the system operational phase. Figure 5 describes the source of the fault, as well as the phase when the fault could be prevented. Generally, development fault prevention should be a part of the development methodology used for the system. Operational faults preventions are handled in the system management methodology.

Fault tolerance

Systems, where measures are taken to prevent every imaginary fault from occuring, can be expensive and complicated to both develop and manage. Fault tolerance analysis can be used to find the suitable level for the system by identify mitigating actions, which enables high availability due to effective system management. Figure 6 shows different means to achieve strong fault tolerance.

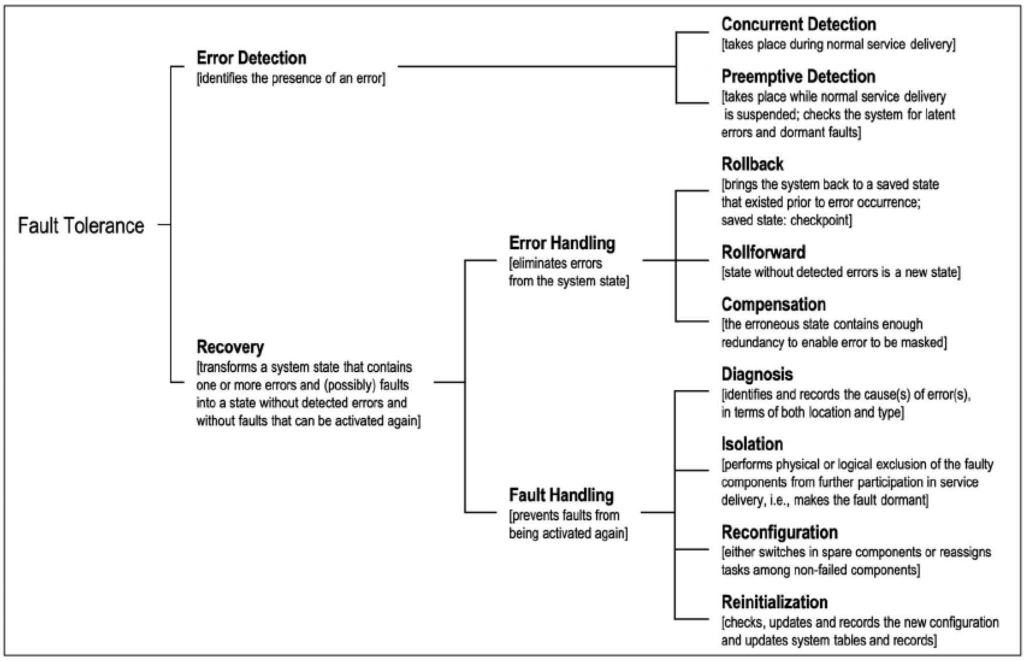

Fault Tolerance
Error Detection
[identifies the presence of an error]
Concurrent Detection
[takes place during normal service delivery]
Preemptive Detection
[takes place while normal service delivery is suspended; checks the system for latent errors and dormant faults]
Recovery
[transforms a system state that contains one or more errors and (possibly) faults into a state without detected errors and without faults that can be activated again]
Error Handling
[eliminates errors from the system state]
Rollback
[brings the system back to a saved state that existed prior to error occurrence; saved state: checkpoint]
Rollforward
[state without detected errors is a new state]
Compensation
[the erroneous state contains enough redundancy to enable error to be masked]
Fault Handling
[prevents faults from being activated again]
Diagnosis
[identifies and records the cause(s) of error(s), in terms of both location and type]
Isolation
[performs physical or logical exclusion of the faulty components from further participation in service delivery, i.e., makes the fault dormant]
Reconfiguration
[either switches in spare components or reassigns tasks among non-failed components]
Reinitialization
[checks, updates and records the new configuration and updates system tables and records]

Figure 6 shows the different means, in a tree structure, to achieve strong fault tolerance.

Fault removal

Fault removal during the use phase is either corrective or preventive maintenance. The corrective maintenance is followed by one or more reported errors to remove the faults that initiated the errors. There are usually two stages in corrective maintenance; Firstly isolation and then removal. Preventive maintenance aims to uncover dormant or non-initiated faults that lead to errors.

Fault forecasting

In order to perform preventive maintenance fault forecasting is necessary. Forecasting is performed by evaluating system behaviour with respect to fault occurrence, which can either be done qualitatively or quantitatively. In a qualitative approach, faults are identified by defining failure modes of the system. Quantitative approach utilizes a model of the system to base the analysis on; in section 4.1 this is described in further detail.

3.2.2 Failures

When considering service failure modes there are four viewpoints to consider:

- The failure domain
- The detectability of failures
- The consistency of failures
- The consequence of failures

These viewpoints, which will be described further, are combined in a tree structure in Figure 7.

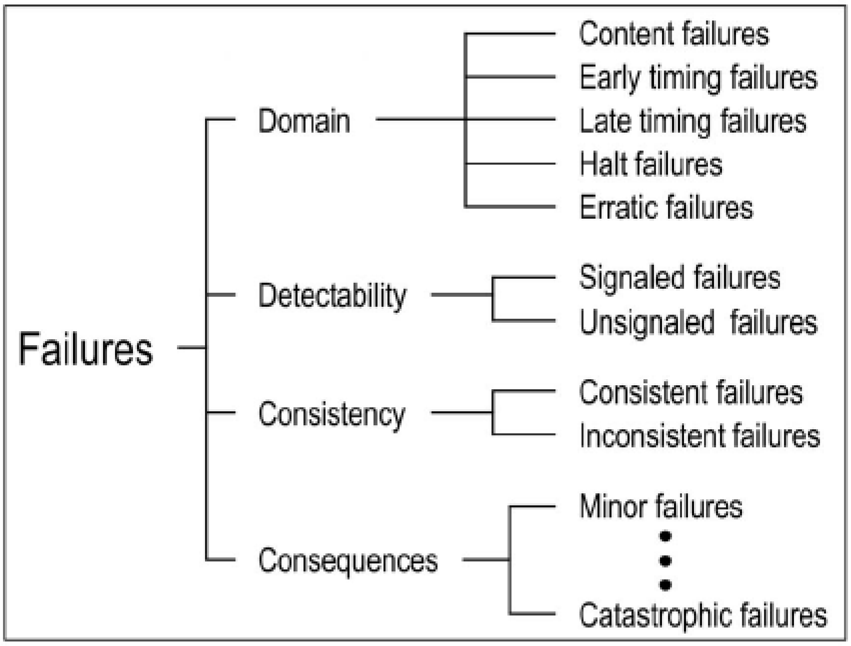


Figure 7 shows the failure tree and the subsumed viewpoints.

Domain

When considering the failure domain, see Figure 8, there are three different viewpoints:

- **Content failures with correct timing**. This failure occurs when the content delivered at the service interface does not implement the system function.
- **Timing failure with correct content**. This failure occurs when the timing or duration of the information delivered does not implement the system function.
- **Content and timing failures**. In occurrence of content and timing failures the consequence is halted service or erratic service.
  - **Halt failures**, when service halts; a special case of halt is silent failure, when no service at all is delivered.

- **Erratic failures**, when service is erratic; the service is not halted but is erratic (e.g., babbling)

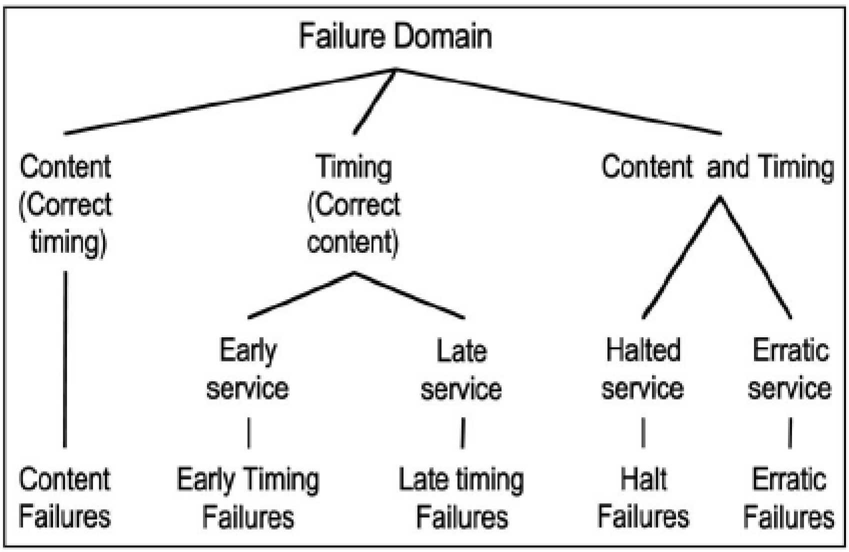


Figure 8 shows the failure domain.

Detectability

The second viewpoint is detectability of failures. Here, there are two subsets of failures; signalled and un-signalled failures. When a failure occurs and is picked up by a warning system and exposed, the failure is signalled. If the failure is not picked up and exposed, it is un-signalled.

For example, the use of try {function code} catch {error handling code} in source code will produce signalled failures.

Consistency of failures

When determining the consistency of failures, the study must be performed as a comparison between different system users. At consistent failures, all system users perceive the incorrect service identically; whilst at inconsistent failures some or all system users perceive incorrect service differently.

Consequences of failures

Grading of the consequences enables failure severities to be defined. Here, two limiting severities can be defined:

- **Minor failures** – where the harmful consequences are of similar cost to the benefits by correct service delivery.

- **Catastrophic failures** – where the cost of the consequence is orders of magnitude higher than the benefit provided by correct service.

There are different classes of severity which depends on different criteria, e.g.:

- **Availability**, the outage duration
- **Safety**, the possibility of human lives being endangered
- **Confidentiality**, the type of information that might be unduly disclosed
- **Integrity**, the extent of the corruption of data and the ability to recover from these corruptions

There are several more possible criteria and they should be determined with respect to the intended system function.

3.2.3 Means

IEEE has also composed tree models for Maintenance, see Figure 9.

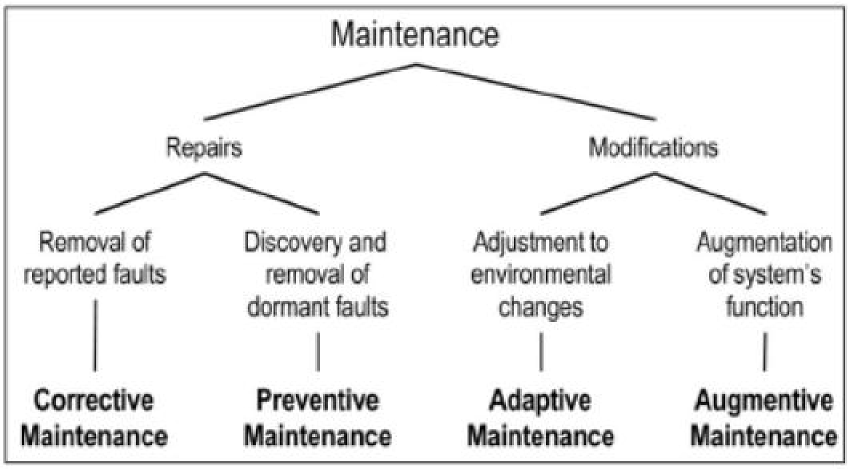

Maintenance
Repairs
Modifications
Removal of reported faults
Discovery and removal of dormant faults
Adjustment to environmental changes
Augmentation of system's function
Corrective Maintenance
Preventive Maintenance
Adaptive Maintenance
Augmentive Maintenance

Figure 9 shows the maintenance tree and the subsumed categories.

The means that are available to improve the subsumed attributes of Dependability and Security is a set of Fault handling categories. A specific action in the fault handling categories will affect one or more of the attributes, e.g. implementation of routines for IT-support actions will improve the Integrity, Reliability and maybe even the Availability.

The set of Fault handling categories is listed below:

- **Fault prevention:** means to prevent the occurrence or introduction of faults.
- **Fault tolerance**: means to avoid service failures in the presence of faults.
- **Fault removal:** means to reduce the number and severity of faults.
- **Fault forecasting:** means to estimate the present number, the future incidence, and the likely consequences of faults.

3.2.4 Summary

Combining all of the topics covered in section 4.2 the Security and Dependability tree is derived as seen in Figure 10.

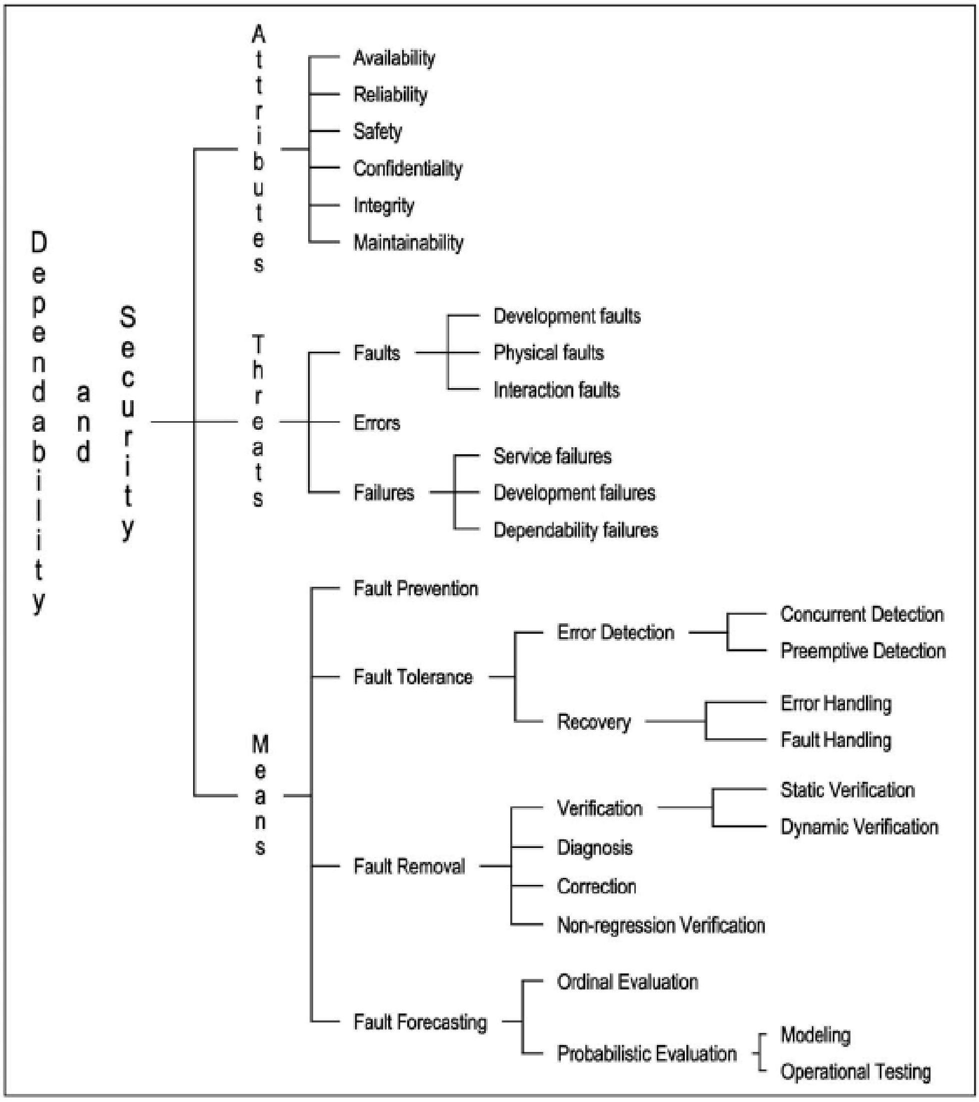


Figure 10 shows the security and dependability tree.

4 Results of suggested framework

The following section will describe the results of the suggested framework, in terms of what possible ways the framework can be utilized. In order to validate the usage of the framework, several tests have been performed. The tests are described in Appendix A: Test of framework for reliability analysis of SCADA solutions, from here on referred to as Appendix A.

There are four vital aspects to the frame work which are briefly explained in the two paragraphs below and described in more detail further on in this section.

Abstraction and aggregation level
To determine the level of abstraction and aggregation which the framework shall be complaint with is an important task. This will set the boundaries for the level of detail the analysis will cope with.

Studies of perspectives and phases
The conceptual design of the framework utilizes multiple dimensions where perspectives and phases are of special interest. The perspectives and phases have been discussed in the start-up workshop and further analysed after the interviews. A description how to introduce new perspectives and what to consider has been a request. The theory of perspectives and phases can be found in section 5.

4.1 System modelling

In order to appropriately analyse a system, models of the system must be defined. In this section, the main aspects required for an adequate model are presented. Also, different analysis technics to analyse the model are presented.

4.1.1 Definitions

The following presents definitions of the parts that the framework consists of. All of the definitions stated shall be regarded when implementing the framework analysis. The definitions enable a fully segmented structure of the framework's parts. This is a necessary to leverage the performed analysis and avoid adjunction.

Objects

Definition 1: An object has a function in the system.

The object is represented as a block in the system model. Each object x is assigned an index *i*. Hence, x_i represents the i:th object x. The indexes will be used to describe what objects are included in a specific calculation.

Object Attributes

Each object can be assigned the following attributes:

- A λ-value representing the probability of failure where $\lambda \in [0,1]$
- A γ-value representing the cost of purchase for the object where $0 \leq \gamma \leq \infty$ [SEK]
- A ξ-value representing the time to successful intrusion where $0 \leq \xi \leq \infty$ [hours]
- A φ-value representing the time to failure (TTF) where $0 \leq \varphi \leq \infty$ [hours]
- A ψ-value representing the mean down time (MDT) $0 \leq \psi \leq \infty$ [hours]
- A ρ-value representing the cost of competence $0 \leq \rho \leq \infty$ [SEK]
- A ω-value representing the cost of restoring service $0 \leq \omega \leq \infty$ [SEK/h]

To be able to analyse a specific discipline of a system all objects of the system must have relevant attributes defined.

In this study, the first attribute, the λ-value, has been studied to the greatest extent. The λ-value is the measure which is most correlated with the reliability of a system. For reliability measures other attributes are also relevant, such as time to failure (the λ-value is to great extent another way of describing this) and mean down time, ψ. However, this attribute describe availability rather than reliability.

Other attributes values, such as cost for restoring service, can be used as basis for governance decisions.

Defining the value of an attribute can be done by using several different technics, i.e. through historic empiric data or utilizing own experience. Another option to defining the value is to use the taxonomy of security and dependability. As the authors of this report see it, the main usage of the taxonomy is as a support tool for defining an objects attributes that describes its reliability properties. See section 4.2 for more details on the taxonomy.

Dimensions

The dimensions of the framework are the following:

- Discipline
- Perspective
- Phase

The framework is comprised of these dimensions, which all are fully separated from each other. The benefits of these dimensions are that the most interesting viewpoints are regarded and that the combination of them gives the analyst a tool to find overall improvements within defined and segmented dimensions.

Discipline

An object is a combination of at least one, and at most three, disciplines. The different disciplines are:

- Hardware (HW)
- Software (SW)
- Supply system (SS)

The disciplines relate to different subsets of a combined object and are defined by:

Definition 2: HW discipline is the subset of a combined object which only describes the physical function.

Definition 3: SW discipline is the subset of a combined object which only describes the logical function.

Definition 4: SS discipline is the external physical functionality provided to enable the functionality in the HW discipline.

Perspective

An object can be analysed from several perspectives dependent on what question to be answered. The scope of the perspective defines which attributes that are needed to analyse the object. Object attributes are presented in more detail in the section Object Attributes. An example of a perspective that requires a specific attribute is system reliability. System reliability analysis requires that the attribute λ, used to describe the probability of failure for an object, is defined.

Phase

Phase is defined as the time frame in which the system is studied. In general a system has two phases[2], design and governance phase. The first phase, design phase, is a relative short phase. The second phase, governance phase, refers the management phase when the system is in production to support the business process. During the management phase, the original design is often altered to facilitate changes due to wear and tear from normal operations and other outside activities that affect the system.

4.1.2 Model scope

Setting the scope is an important task, which will enable the analyst to limit the time for solution modelling; whilst achieving the desired analysis result. There are several identified approaches when setting the scope of a model, see below.

Thread Top-Down Scope

The thread top-down scope approach aims to describe the dependencies that are present from the top object all the way to a specific end object. A typical thread top-down representation models all the objects from the workstation in the dispatch centre to the RTU in a specific outstation of interest. This

[2] In the taxonomy the two phases used are development and operation; these are to a large extent interchangeable with design and governance that are used here.

approach is preferable when a limited number of endpoints are to be analysed.

Area Scope

If a wider scope is of interest, the area scope approach is suitable. The area scope approach sets the boundaries of the model by selecting a subset of the object. The objects are interlinked in respective parts of the central system, communication infrastructure and outstation. Each of these parts of the structure may be fully separated from each other which facilitate separate calculations to be performed for each endpoint. In Figure 11 the general structure of a system is described. The analysis will describe just one of these areas independently. This can be useful as the work required is focused on the relevant parts and the scope of the work is reduced. To be able to utilize the area scope, all functionality to be analysed must be defined with in the area.

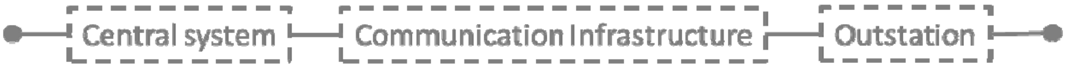


Figure 11: A top level structure model of a SCADA system

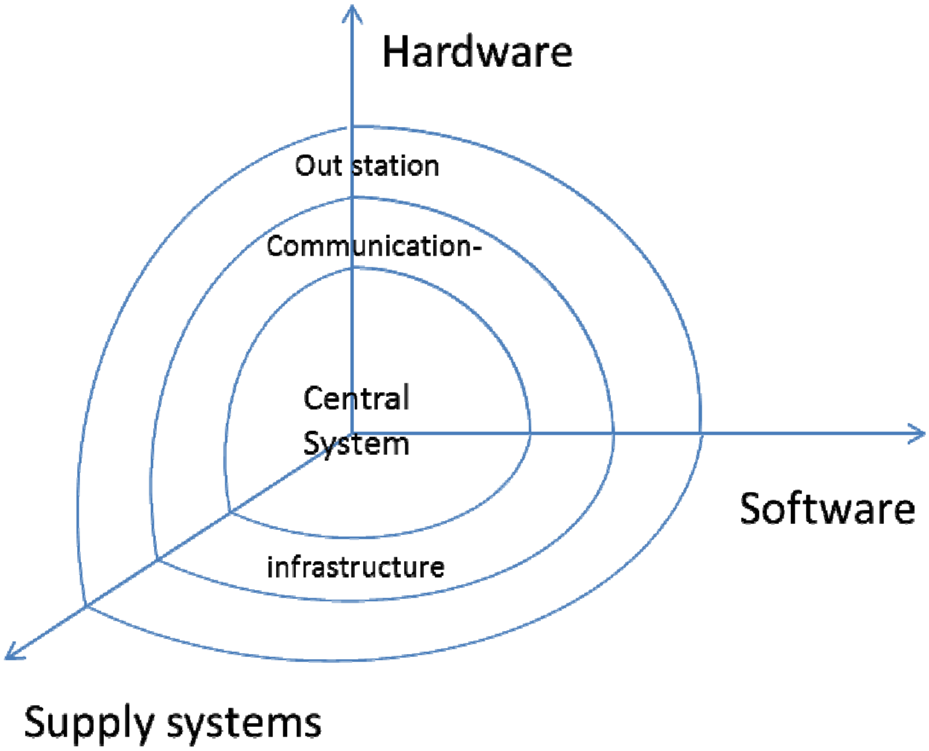


Figure 12: The three disciplines and the three model scopes of the system.

Full Top-Down Scope

The full top-down scope is the most comprehensive but also the scope that requires most work to utilize. The goal of the full top-down scope is to completely model the whole solution. When evaluating which single alteration of the system gives the greatest improvement, this model scope is required.

4.1.3 Model Abstraction Level

The model representation is bounded by the model scope and the abstraction level. The abstraction level determines to what extent objects shall be broken down. As the detail level increases, the model better reflects the dependencies that are present in the real world. However, a detailed level of information might not result in relevant analysis results. Choosing a relevant level is important in order to receive desired results whilst spending minimal work. The proposed levels of abstraction that should be used are stated below.

Physical Abstraction Levels

PAL1: The content of the highest abstraction level is the object containers alone.
A rule of thumb can be: Only model the objects main function.

PAL2: The model is extended by subparts of the objects.
Here an appropriate level of extension is decided to further model the dependencies that are present. A proposed extension is that the model considers all the network interfaces.

Logical Abstraction Levels

LAL1: This abstraction level only models the OS and subsystems, such as DB, virtualization systems and the SCADA application.

LAL2: The model is extended with third party software and the different modules of the applications.

LAL3: At this abstraction level all the system integrations are modelled.

LAL4: At this abstraction level the modules in the applications are broken down to the functions.

4.1.4 Calculations

Structure

The structure of the system can be described on several levels. To achieve flexibility a modular approach for setting up the structure is preferable. A modular structure will require that the interfaces between the different parts of the system are defined. The structure for a modular approach is based on the same structure as used in basic probability theory, see section 4.1. The system structure is also dependent on which abstraction level, see section 5.1.3, and which model scope, see section 5.1.2, that is used.

For an interface to facilitate a modular structure each module needs to start and end with a single connection. In Figure 11 the three main modules of a SCADA system are depicted, all start and end with a single connection. When performing a thread or a full top-down analysis consideration to redundant components should be taken and modular approach can’t be fully utilized.

For analysis purposes this structure may in some cases be to imprecise. Each module can be broken down into more detailed structures that can be analysed individually. The module is described in detail in the structure shown in Figure 13.

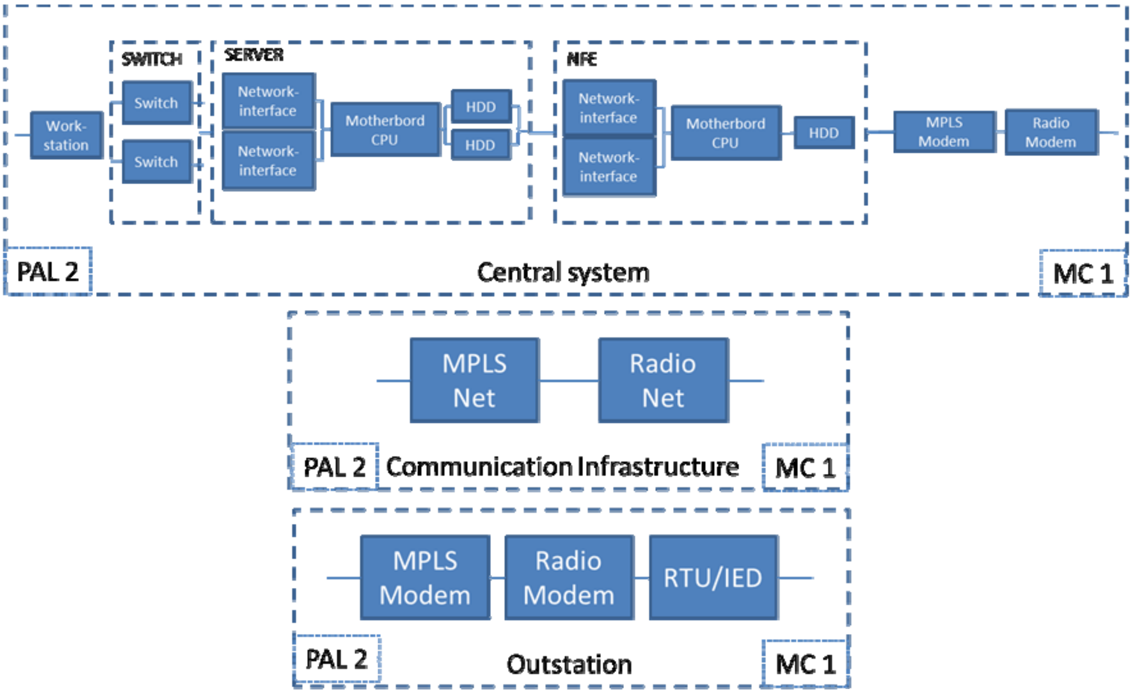


Figure 13: A more detailed structure for the system described in Figure 11. The structure is done with physical abstraction level two and model correctness level one.

By utilizing basic probability theory, systems can be described by different structures as in appendix A. To receive a comprehensive analysis of a system, it is important use both a detailed description, as well as a more concise description.

Discipline

Use the system model rate function for each discipline *i* in the combined function . The three discipline used in this scope define the rate function as:

Perspective

The functions of the different perspectives are given by:

Reliability of a series structure:

Reliability of a parallel structure:

$$f_{average\ cost\ service\ restoration} = \sum_{j}^{m} x_j(\omega) * \ x_j(\psi) \ , where\ x_{j\rightarrow m}\ are\ the\ objects\ in\ outage$$

Workflow

The suggested workflow when applying the framework analysis is given as a flowchart, see Figure 14.

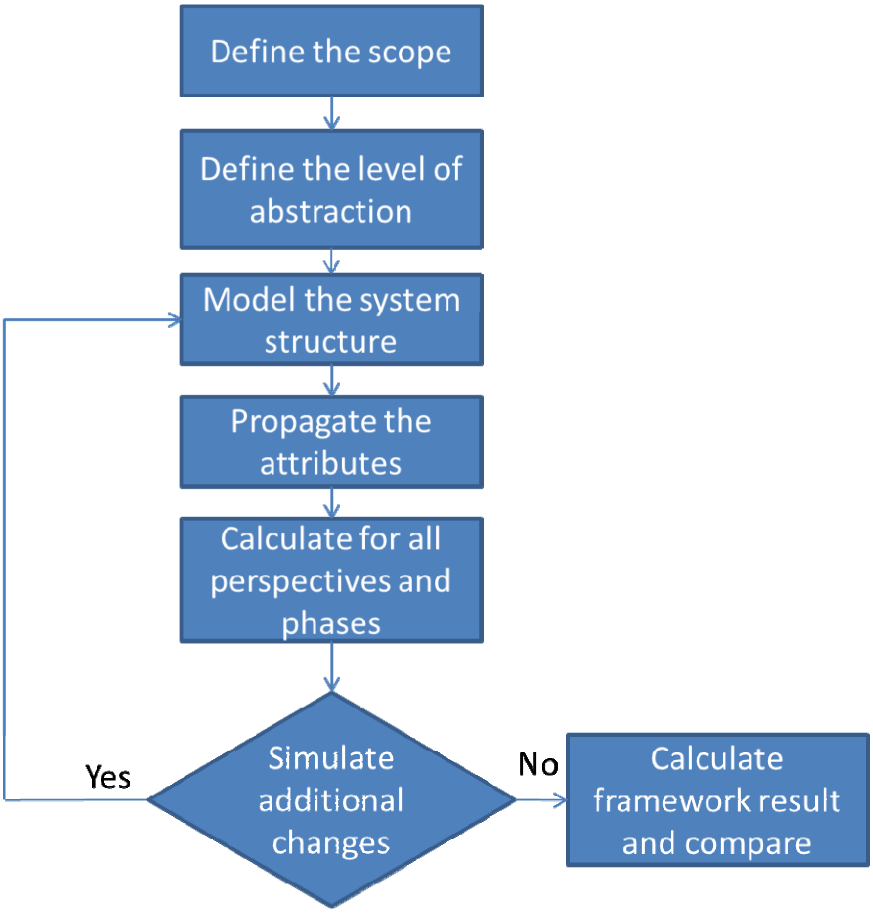

Define the scope
Define the level of abstraction
Model the system structure
Propagate the attributes
Calculate for all perspectives and phases
Simulate additional changes
Yes
No
Calculate framework result and compare

Figure 14 shows the workflow when using the framework.

4.2 Evaluation

Utilizing basic probability theory

Utilizing basic probability theory to evaluate a solution will provide a quantified description. The level of detail describing the evaluation is determined by the options selected for the system modelling. Also, the attributes defined for the object in the system model will determine the scope of the evaluation.

The possibility to evaluate a SCADA solution using probability theory has been tested, see appendix A for further information on the test performed, and is concluded to be possible.

Utilizing taxonomy of security and dependability

By using the taxonomy for security and dependability it is possible to perform qualitative evaluation of a system. The detail level, to which the evaluation can be done, is dependent on the level of detail for the information available for describing the system. If the system is modelled as suggested, the details of the system model is determined by the properties described in section 5.1.

In the tests performed, see appendix A, the taxonomy has been used to define the value of the attributes; attributes used for quantitative analysis using probability theory.

The qualitative analysis methodology is partly done in parallel with quantitative analysis. The first step in the quantitative analysis is to define the values of the attributes. This requires that qualitative analysis is performed, which implies that faults, failures and the means to mitigate these are considered. In Figure 15 the workflow for defining the value of an objects attribute is described.

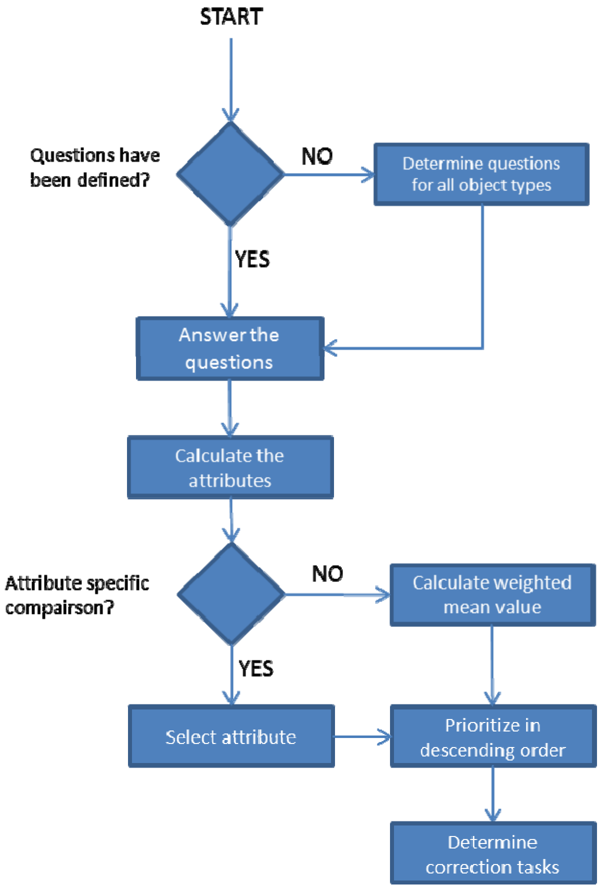

START
Questions have been defined?
NO
Determine questions for all object types
YES
Answer the questions
Calculate the attributes
Attribute specific comparison?
NO
Calculate weighted mean value
YES
Select attribute
Prioritize in descending order
Determine correction tasks

Figure 15 Workflow of Analysis to define the value of objects attributes.

4.3 Governance

Governance is a continuous process with the purpose of maintaining, or even improving, the performance of the system. The taxonomy, described in section 4.1, gives a structured way of defining the threats and means for any system. The usage of the taxonomy is not mandatory when using the framework. However, the taxonomy provides a good support to ensure coverage of possible threats when setting up maintenance tasks.

The proposed usage of the taxonomy works as a quality system for the objects in the SCADA solution. All objects have their own unique set-up and therefore the threat will be unique for all objects. It is possible to in advance determine questions according to the threat taxonomy and through these question rate the object and derive its attributes.

5 Results of input from stakeholders

In order to acquire the industry perspective on how the framework can be used in practice, interviews have been performed with a number of stakeholders. Two types of stakeholders have been identified to represent the industry perspective:

- Utilities
- Vendors

Interviews with stakeholders have mainly focused on three areas of interest:

- Need – Identifying the stakeholders need of a framework to study reliability of SCADA solutions
- Obstacle – If there is a need identified, what are the obstacles in the use of the suggested framework
- Suggestion – How can the framework be used and what are the possible applications of it

The following will briefly outline the main features from the interviews.

5.1 Documentation and modelling

Overall, the interviewed utilities today work in limited extent with documentation and modeling of reliability of SCADA solutions. While some have a lot of information and documentation regarding their network, others have less. A common denominator for the utilities is the lack of a consistent gathering place, where information and documentation can be stored. A mutual expressed need is to work in a more structured way when it comes to documentation. The reason why the interviewed utilities are in need of a more structured way of working differs. While some want to get a better picture of existing documentation, others want to move away from being dependent on one or two individuals and spread the knowledge to others.

The utilities are seeking a framework that is easy to manage; otherwise there is an imminent risk that the framework will not be utilized. At the same time, some of the utilities express a risk that a model, built on an aggregated level, will not enough clarify the systems properties in detail to be useful.

In order to build a model, the utilities want support from vendors regarding producing and providing documentation. It is suggested that the documentation is provided at the time of procurement. Also, utilities ask for the vendors support regarding training and integration of other vendors' systems. Interviewed vendors express that they are positively inclined to provide needed documentation for the utilities.

None of the interviewed utilities use modeling today to analyze reliability of SCADA solutions. Obstacles mentioned are lack of resources and time.

5.2 Evaluation

None of the interviewed utilities have previously worked with quantitative analysis to evaluate SCADA solutions. Previous methods of analysis have been based on experience and assumptions. It differs widely between the utilities if they are interested and/or in need of quantitative or qualitative evaluation.

Some of the interviewees wish to have a support system in order to be able to store documentation and calculate reliability for different SCADA solutions. Today, there is no such support system available on the market that fit for SCADA solutions.

All of the interviewees expressed difficulties knowing on what level to perform analysis for their system.

5.3 Governance

An expressed way of using the framework to govern is to use the results of the evaluation as a basis for decision making regarding investments, as well as information sharing and/or long term planning of SCADA development.

6 Analysis

6.1 Analysis of input from stakeholders

The ability to model, evaluate and govern based on a framework for reliability studies of SCADA solutions varies and depends mainly on the following parameters:

- Size/complexity of SCADA solution
- Organizational structure and knowledge
- Existing documentation

While the study has focused less on the latter two, the analysis has been based on the size/complexity of SCADA solution.

To summarize the analysis, the ability to model, evaluate and govern SCADA solutions have been visualized in a graph below, see Figure 16 . The picture shows a diagram with two axes; the x-axis representing the size/complexity of the SCADA solution, the y-axis undefined depending on which arrow is studied. Both axes have the same scale – from low to high.

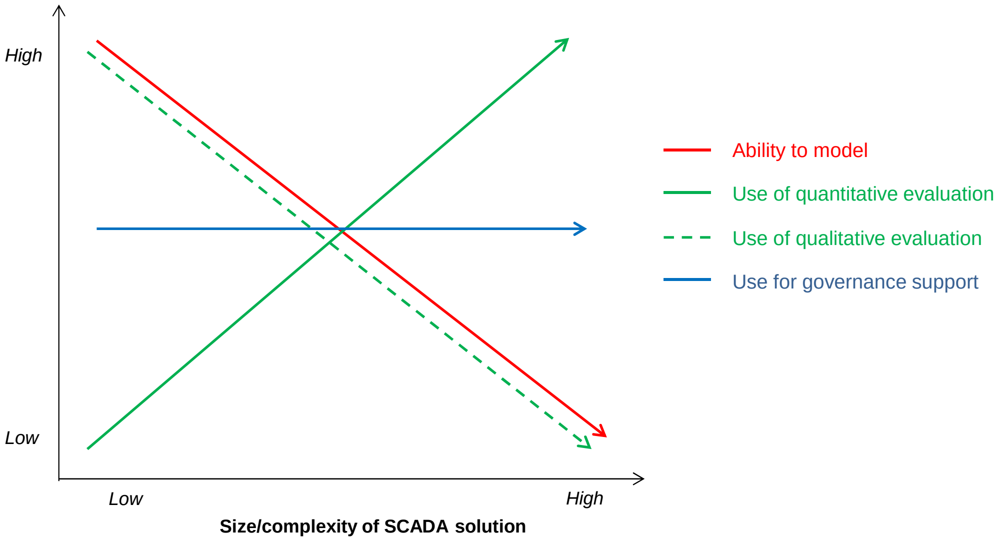


Figure 16 Visual graph of analysis.

The following will further describe the picture.

6.1.1 Ability to model

The study shows that utilities/organizations with less complex SCADA solution have high ability to model. The reason for this is that these actors have fewer components in their system. Because of this, they require less input to create a functioning model and are assumed to have more information of higher quality.

Also, we see a correlation between the complexity of the system and obstacles the utility sees in using the framework.

The greatest obstacles to modelling are:

- Collecting information
- Storing information
- Maintaining information

While collecting proper information is the easiest - although time consuming - of the above mentioned obstacles, at the moment there is no appropriate repository available to store and maintain the information. These obstacles are considered viable to overcome and over time it is possible to develop ways of working that enables modelling.

6.1.2 Use of evaluation

The study shows that the less complex SCADA solution the utility/organization has, the less interesting it is to perform quantitative analysis to evaluate different reliability solutions. To perform calculations the utilities require resources. The smaller utilities are not able to or interested in calculating SCADA solutions, although information management systems are of interest. However, larger utilities/organizations with more complex SCADA solutions generally find a quantitative evaluation more useful though it gives a more structured way, in comparison with assumptions and experience, to motivate the need for development and improvement.

While utilities with smaller SCADA solutions are also interested in using the framework to get a more structured way of managing SCADA, they are more interested in qualitative evaluation rather than mathematical reliability studies. The methodology of the framework is seen as a useful tool and way of working, whereas the main applications are seen as strategic decision making or information sharing.

A problematic issue with the framework is that the ability to model and the use of quantitative evaluation are inverted. For example, the utilities/organizations with less complex SCADA solutions and high ability to model are less interested in quantitative evaluation. While those interested in quantitative evaluation, are less able to model. Here, the framework needs to improve in order to be useful for all.

6.1.3 Use of governance support

The study shows that the use of a framework as governance support is similar for all stakeholders, independent of size or complexity of SCADA solution. The larger the size of the utility/organization, the higher the interest in using the framework as support for decisions on a more detailed level. The interest is also high for strategic support as well, which is where the smaller organizations main interest is.

Regardless of interest in and ability to model and evaluate, there is a need to get a more structured and systematic way of managing SCADA solutions. Overall, the general view is that the framework can serve as a good development and support to governance SCADA system.

6.1.4 Overall

Although the framework is not fully developed and some improvements are still needed, the framework is proven to provide support for decisions regarding improvements of reliability of SCADA solution.

The study clearly shows that for a framework to work there must be some sort of computerized support to handle information and documentation, as well as calculations. If not, the framework will have limited functions and will more show a way for a work process rather than calculating different solutions. If the utility/organization has a complex system, a support system is of even greater importance.

6.2 Analysis of framework test

All observations made during the tests is that only the project group that have taken part in developing the framework have performed the tests. To get a more accurate user point of view the framework should be tested by external users on real life systems. Conclusions drawn from the test should be relevant from a user point of view despite the lack of outside testing.

The test shows that the framework can be used as intended. Also, the test shows that certain prerequisites need to be fulfilled in order to use the framework, as well as boundaries of model and data handling.

With the framework it is possible to set up models of large, and relatively complex, SCADA solutions. However, performing analysis requires comparison between different structures and setting up several structures may be time consuming work. This work can be avoided if there was an tool that effectively could manage models.

Also, the test showed that relative analysis, comparing two different solutions, will mitigated the need for property data of high quality, at least when it comes to reliability analysis.

7 Conclusions

This report has presented a study of framework for reliability analysis of SCADA solutions. Within the work a framework has been developed, and tested. Also, the report has presented the industry perspective on the developed framework from two categories of stakeholders.

The developed framework for analysing reliability of a SCADA solution works, although tests so far only have been performed on abstract test systems. There is no reason why the framework shouldn't work on real life systems. Further studies should be made on real life systems to improve and verify the framework.

It is clear that some sort of tool is required to support modelling of the SCADA solution, as well as defining structure functions, managing data and comparing different set ups. For the tests performed in this study a very basic tool for this has been developed; this tool could be used on real life systems.

To summarize the conclusions of the study, a list of pros and cons of the framework are presented below.

Pros

- The framework provides a sought after structured and systematic way of working and manage reliability of SCADA solution
- The framework is a useful tool to describe the SCADA solution for those who are not familiar with it, i.e. sharing of information and knowledge
- The framework gives a great basis for investment and strategic decisions
- If the framework is followed , it gives a continuous development of the SCADA solution
- The framework provides an opportunity to analyze both quantitative and qualitative
- The framework is useful regardless of size/complexity of SCADA solution

Cons

- Great need for computerized support to store and maintain information, as well as to perform calculated analysis for evaluation
- One of the obstacles to build a model is the lack of information. Collecting the required information can be both costly and time consuming.
- Difficulties to interpret the output of the model. Stakeholders need to build up knowledge to be able to evaluate the output.
- If the framework is too complicated and/or difficult there is an imminent risk of it not being used

An example of one type of analysis which could be utilized is as basis for strategic decisions for either developing passive redundancy or increased resources on call for failure correction to communication channels. This can be done through the framework as two set ups of the SCADA solution, one with a new fully redundant communication channel and one where the mean repair time is decreased could be compared from several viewpoints. Firstly the mean unavailability between the setups could be compared but also the cost aspects for both the solutions can be evaluated.

Another example of utilization of the framework to support the organization that handles SCADA solutions is fault detection and minimization of restoration time. Via the system model potential fault causes on specific part of the SCADA solution are identified. This will enable the organization to create a list faults connected to respective component in the SCADA solution. This will also enable mitigation actions to either prevent the fault or to minimize the down time caused by a fault by having better information about the possible causes of the fault and which equipment that would be beneficial to have in stock.

7.1 Future work

Conclusions of the study showed that the framework was a viable and interesting approach to evaluate and govern SCADA solutions from a reliability and risk point of view. The framework in the version requires additional development to get it better adjusted to the industries requirements. In order to adjust the framework to the industries requirements study where participants from the industry have a more active role in testing is one possibility. The industry also are interesting in finding some form of tool to assist in information management and system modeling to facilitate the use of the framework. A study into this is also a possibility in order to further develop the framework.

8 References

Bertling, Lina (2004), Tillförlitlighetsanalys av elkraftsystem 2C4030 – kursmaterial 2005

IEEE Transactions on dependable and secure computing vol. 1 no. 1 January-March 2004, Basic Concepts and Taxonomy of Dependable and Secure Computing

9 Appendices

Appendix A: Test of framework for reliability analysis of SCADA solutions

ELFORSK

SVENSKA ELFÖRETAGENS FORSKNINGS- OCH UTVECKLINGS - ELFORSK - AB

Elforsk AB, 101 53 Stockholm. Besöksadress: Olof Palmes Gata 31
Telefon: 08-677 25 30, Telefax: 08-677 25 35
www.elforsk.se

Appendix A

Test of framework to enable reliability analysis of SCADA solutions

Elforsk rapport 21:56

Stefan Svensson

September 2012

ELFORSK

Appendix A

Test of framework to enable reliability analysis of SCADA solutions

Elforsk rapport 21:56

Stefan Svensson

September 2012

Innehåll

1 Test of framework to enable reliability analysis of SCADA solutions

In order to verify the framework a number of tests will be performed. These tests will not completely verify the functionality of the framework but will rather result in showing if it can work as a tool or not in a practical sense.

To test the framework a generic SCADA system is set up, only on paper, which encompasses most of the functionality that is within the scope of the analysis, which the framework is to facilitate.

2 Description of the test system

Normally, a SCADA solution comprises of three main areas:

- Central System (the SCADA system and it's peripherals)
- Communication Infrastructure
- Outstation measurements and control signal equipment

If an analysis according to the framework is to be productive, the content in the areas above must be described in more detail.

To describe the solution as a whole, the approach made is to model each of the areas by themself and then define interfaces between each of the areas.

In the model, there are a number of properties for each object that describes the object. Which of the properties relevant for describing the object depends on from which perspective that the analyze of the SCADA Solution is to be from.

A more detailed look at how properties (attributes) can be utilized for different types of analyses will be described more elaborately in the following sections.

2.1 Central System

The central system consists of servers and workstations, which run all the relevant software, and a number of switches and network cables to interconnect the objects using the network. These objects are connected to the Network Front Ends and the modems for each respective communication medium. The user, operator, interaction with the SCADA solution is also a part of the Central System area; this is modeled as a workstation.

2.1.1 Physical

In this test system the servers have single processors, multiple hard drives and RAM memories. There is one server; running the SCADA applications and the database application. The SCADA application server is connected to four workstations, which are used by the operators to control the system via the IP-network. The workstations, servers and the network front end (NFE) are all connected via two switches; both of them having the same configuration, model and age. The SCADA application is also connected to the NFE device. For communication with the outstation devices both an MPLS service and a private Radio network is used. See Figure 1 for an overview picture of the central system.

2.1.2 Logical

Main system

The SCADA system consists of a SCADA application which performs all SCADA operations and a database that stores all events that are handled in the system. The database runs on the same server as the application. The SCADA client interfaces runs on the operating system (OS) of the workstations and uses the SCADA server interface, which runs on the server OS. No virtualization of servers is in use in this example.

Switches

There are two Level 3 switches in the system that run a routing and switching table application on the switches embedded OS. The switches use standard configuration and are set up by the vendor.

2.1.3 Supply system

The supply systems that are required for the objects in the Central System are electricity, climate control and premises security. The climate control can utilize water as a cooling/heating medium. See Figure 4 for an overview picture of the supply system where the central systems supply system is one part.

2.2 Communications Infrastructure

Communication infrastructure is composed of a private Radio network and a MPLS network rented from an internet provider.

2.2.1 Physical

The cabling and the components in the MPLS network of the internet service provider are the physical blocks; these will be modeled as one block since its properties are unknown. The private radio network is modeled as one block representing the air through which the radio waves flow as well as any repeater stations or other equipment. See Figure 2 for an overview picture of the communication network.

2.2.2 Logical

The logical description of the communication infrastructure is the protocol for the Radio network and the MPLS protocols.

2.2.3 Supply system

The supply system for the MPLS network is mainly electricity. The supply system for the Radio network is also electricity; this powers the repeater stations for the radio. The modems, receiver and transceiver are a part of the Central System area and the Outstation area. See Figure 4 for an overview picture of the supply system where the communication network is one part.

2.3 Outstation

There are a number of outstations in the systems which have different set ups with regards to RTU/IED and modems for communications. Each outstation also has different properties with regards to importance for the system; i.e. the consequence of losing a specific outstation is different.

There are 12 outstations in the system. The outstations have the properties described in Table 1.

Table 1 shows the test systems outstation set up. RTU/IED are denoted as standard 1,2 and 3 which describe which model and version that are used.

Outstation	Communication 1	Communication 2	RTU/IED	Consequence [1-5]
1	MPLS		Standard 1	3
2	MPLS		Standard 2	3
3		Radio	Standard 3	2
4		Radio	Standard 3	1
5	MPLS	Radio	Standard 1	2
6		Radio	Standard 1	3
7	MPLS	Radio	Standard 2	5
8	MPLS		Standard 1	3
9	MPLS	Radio	Standard 2	4
10		Radio	Standard 1	5
11	MPLS		Standard 3	4
12	MPLS	Radio	Standard 1	4

2.3.1 Physical

The Physical set up for the outstations is different for all twelve stations; there are three general setups that are used. In Table 1 the different configurations of each outstation is presented. See Figure 3 for an overview picture of the outstations configuration.

2.3.2 Logical

The logical structure is comprised of the logical parts of the system such as receiver/transceiver applications, the RTU applications and the operating systems on the RTUs. As the different RTU's are from different vendors and from different points in time there are different software, versions, and configurations of software. In this test all outstation equipment is defined to have the same properties despite that this.

2.3.3 Supply system

For the objects in the outstation to be able to perform its tasks electricity and climate control is required. See Figure 4 for an overview picture of the supply system where the outstations supply system is one part.

2.4 Modeling of the test system

Multiple models will be derived to shed some light on differences and what end results are plausible to derive. The differences in the models will be the level of abstraction.

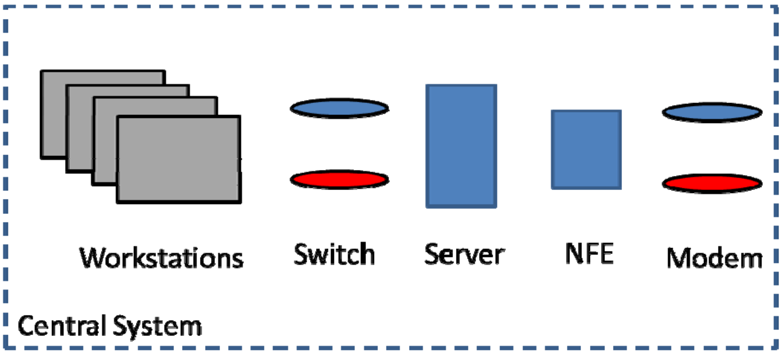


Figure 1: a comprehensive high level view of the central system to be modeled.

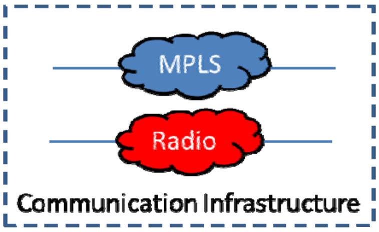


Figure 2: a comprehensive high level view of the communication infrastructure to be modeled.

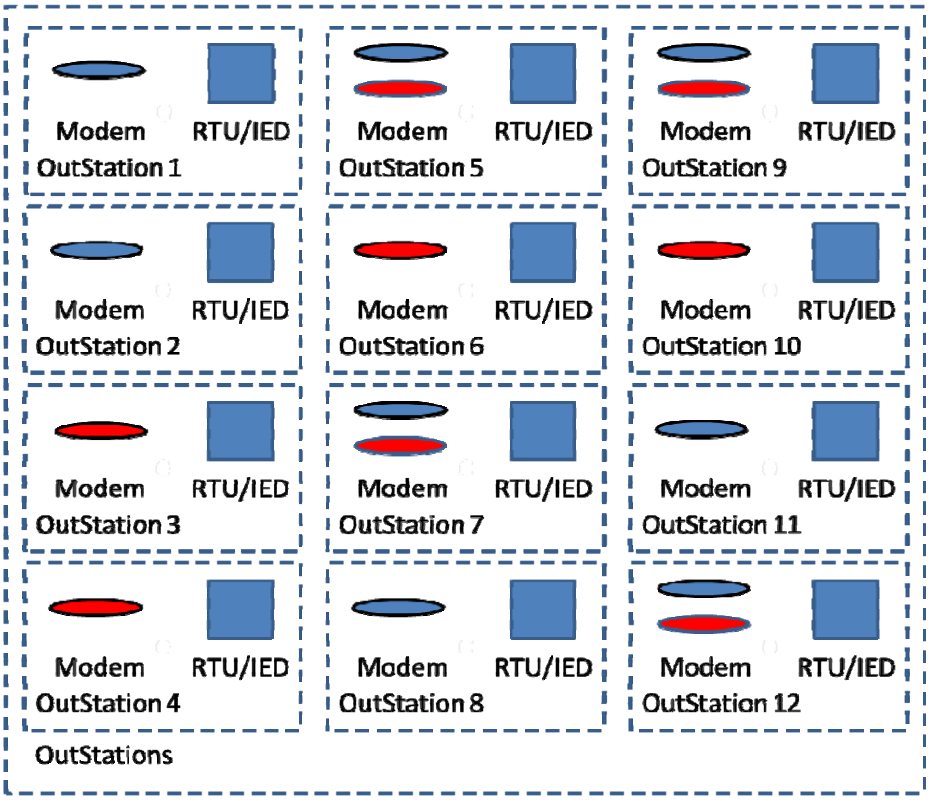


Figure 3: a comprehensive high level view of the different outstations setups to be modeled.

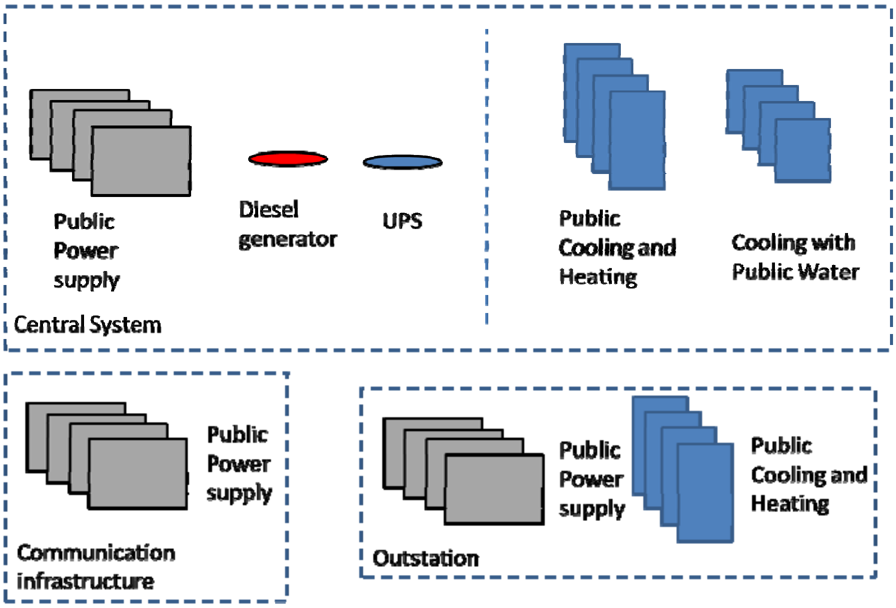


Figure 4: a comprehensive high level view of the supply systems for the three areas of the systems to be derived into a model.

2.4.1 Model scope

The model scope for all test systems is the central system, communication and the outstation. The scope includes all three layers; Logical, physical and support system.

2.4.2 Model abstraction level

The system will be modeled with separate models of the physical layer; this includes the physical representation and the supply system, and a logical layer. Each layer will also have different levels of abstraction. The abstraction level corresponds to the level of detailed used to represent the system; high abstraction level corresponds to low level of detail in the models.

Physical abstraction level one
The first (highest) abstraction, PAL 1, level is used for the first test model; see Figure 5 and Figure 6. An example of the abstraction is that a server is only modeled as a server object with no regard for the components that it holds.

Physical abstraction level two
The second level of physical abstraction is called PAL 2. This will affect the SCADA application server and the Front-End server models by introducing the components network interface, motherboard/CPU and HDDs as objects in the model.

The reason for choosing these two system components to be modeled with greater detail is that decision on the set up for these has significant impact on the system functionality which corresponds to abstraction level 2. Introducing more details to the model would give abstraction level 3 or higher.

Logical abstraction level one
The first level of logical abstraction, LAL 1, is utilized in the test of the full top-down thread analysis, see Figure 9. The software, which is modeled, is the operating system (OS), subsystems (such as DB or similar) and the application services.

Logical abstraction level two
If the model in Figure 9 is extended by modeling some of the implemented software components in the server, workstation and Front-End server this gives abstraction level two, LAL 2. Abstraction level two can be seen in Figure 10.

For the same reason as mentioned in Test model 2, these parts of the system are chosen to be described in more detail at this level.

Supply system abstraction level
To describe the supply system that enables the functionality of the system, a model that illustrates this is derived. The model uses the first abstraction level, PAL1, for this exercise.

The Supply system area is modeled in parallel with the other areas of the system. It is equally important to consider this area when analyzing the entire system.

2.5 Calculations

To verify the framework calculations for analysis are performed. Calculations are based on probability theory but also utilize other methods. Models will be specified for each calculation based on the description of the system in this section.

2.5.1 Object Attributes

To be able to perform the calculations on the test system all objects need to have certain attributes defined. Dependent on which attribute is defined it will be possible to perform different types of analysis.

The taxonomy for security and reliability is a tool that can be utilized to determine these attributes. The following attributes are used in the test and therefore defined:

- A λ-value representing the probability of failure where $\lambda \in [0,1]$
- A ψ-value representing the mean down time (MDT) $0 \leq \psi \leq \infty$ [hours]
- The ω_P- and ω_T-values in [EUR] resenting the cost of restoration to service for permanent as well as transient errors, the ω_P-value is equal to the cost of new equipment

The test attributes are set to the values in appendix A. In this test system the attributes values are defined as relative to an ideal functioning system. The values also have no meaning in an absolute sense since they are determine for use in a relative analysis.

For each of the system objects defined for the tests the failure rate is defined as faults per years of use. In this test the definition is arbitrarily performed in accordance with experience without being based on data.

In the case were an object is comprised of several sub-objects, in serial connection, the failure rate for the higher level object is based on the failure rate of the sub-objects. The rule of additivity is used in these cases:

Under certain engineering assumptions (e.g. besides the above assumptions for a constant failure rate, the assumption that the considered system has no relevant redundancies), the failure rate for a complex system is simply the sum of the individual failure rates of its components, as long as the units are consistent, e.g. failures per million hours.[1]

2.5.2 Structure functions

To be able to perform analysis of a model of a SCADA solution; its structure needs to be described by a function. The reliability and availability functions calculations are interchangeable.

[1] From http://en.wikipedia.org/wiki/Failure_rate

Reliability of a series structure: $\emptyset(\bar{\lambda}) = \prod_{i=1}^{N} \lambda_i$

Reliability of a parallel structure: $\emptyset(\bar{\lambda}) = 1 - \prod_{i=1}^{N}(1-\lambda_1)\cdot(1-\lambda_2)\cdots(1-\lambda_n)$

Availability: A = 1- λ ψ

Availability of a series structure: $\bar{A} = \prod_{i=1}^{N} \mathrm{A}_i$

Availability of a series structure: $\bar{A} = 1 - \prod_{i=1}^{N}(1-\mathrm{A}_1)\cdot(1-\mathrm{A}_2)\cdots(1-\mathrm{A}_n)$

2.5.3 Discipline functions

To be able to analyze the impact of the different disciplines on the total solution the discipline function can be utilized. When considering disciplines functions the dependability between the disciplines needs to be modeled correctly. Therefore the discipline function below is only truly valid for pure series structures.

$$f_D\big(\emptyset_{HW}(\lambda), \emptyset_{SW}(\bar{\lambda}), \emptyset_{SS}(\bar{\lambda})\,\big) = \prod_{I=1}^{3} \emptyset_i(\lambda) = \emptyset_{HW}(\bar{\lambda})\cdot\ \emptyset_{SW}(\bar{\lambda})\cdot \emptyset_{SS}(\bar{\lambda})$$

Parallel structures will require integration between the disciplines and the structure function must be divided in terms of series functions, as above, in parallel.

$$f_D\big(\emptyset_{HW}(\lambda), \emptyset_{SW}(\bar{\lambda}), \emptyset_{SS}(\bar{\lambda})\,\big) = 1 - \prod_{I=1}^{N}\left(1 - \prod_{I=1}^{3} \emptyset_i(\lambda)\right)_i$$

As with the structure functions for reliability the availability function for discipline is interchangeable.

$$\mathbf{A}_D\big(\bar{A}_{HW}(\lambda), \bar{A}_{SW}(\bar{\lambda}), \bar{A}_{SS}(\bar{\lambda})\,\big) = 1 - \prod_{I=1}^{N}\left(1 - \prod_{I=1}^{3} \bar{A}_i(\lambda, \psi)\right)_i$$

3 Test on modeling

The test system will be modeled so that the structure functions can be used to perform the desired analysis. There are three different scopes that facilitate different approaches to the analysis and different results of the analysis:

- Thread Top-Down Scope
- Area Scope
- Full Top-Down Scope

3.1 Thread Top-Down Scope

The thread top-down scope will be tested with two different threads. The structure of the two threads will be defined to show de difference in reliability between the two different threads. For this example the threads from the control room to outstation one and five in the test system are chosen, see Figure 5 and Figure 6,

The difference between the two outstations in this case is the use of communication infrastructure. The RTU/IED units are assumed to have the same reliability properties. For this test, the calculations will be performed on physical level 1. These two threads are referred to as test 1 and test 2 in the calculations.

The analysis focuses on the reliability improvement that adding the redundancy provides. For these calculations the properties defined for the SCADA solution objects defined in I are used.

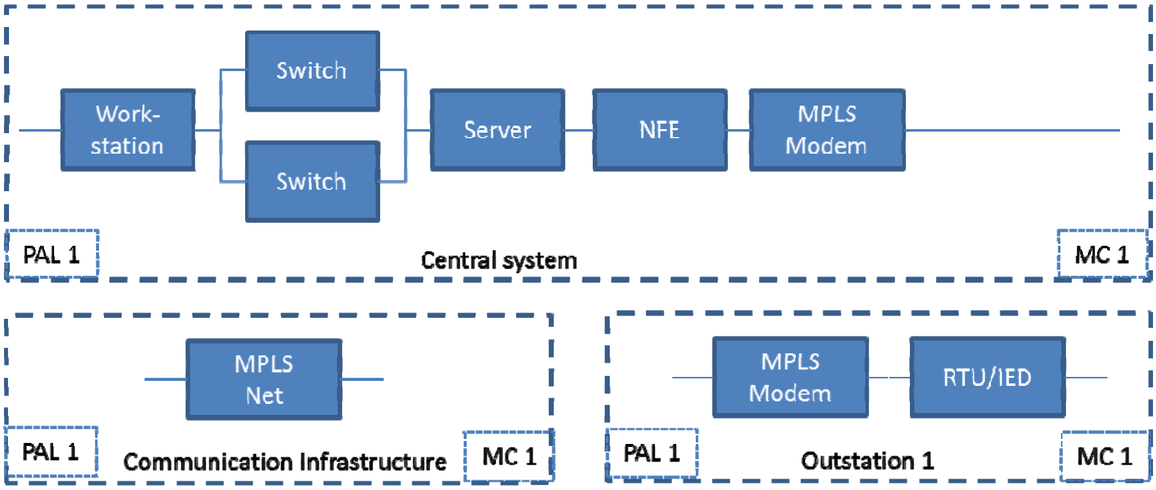


Figure 5: The structure, test 1, of the SCADA solution from control room to outstation 1, almost a total series structure through the whole thread.

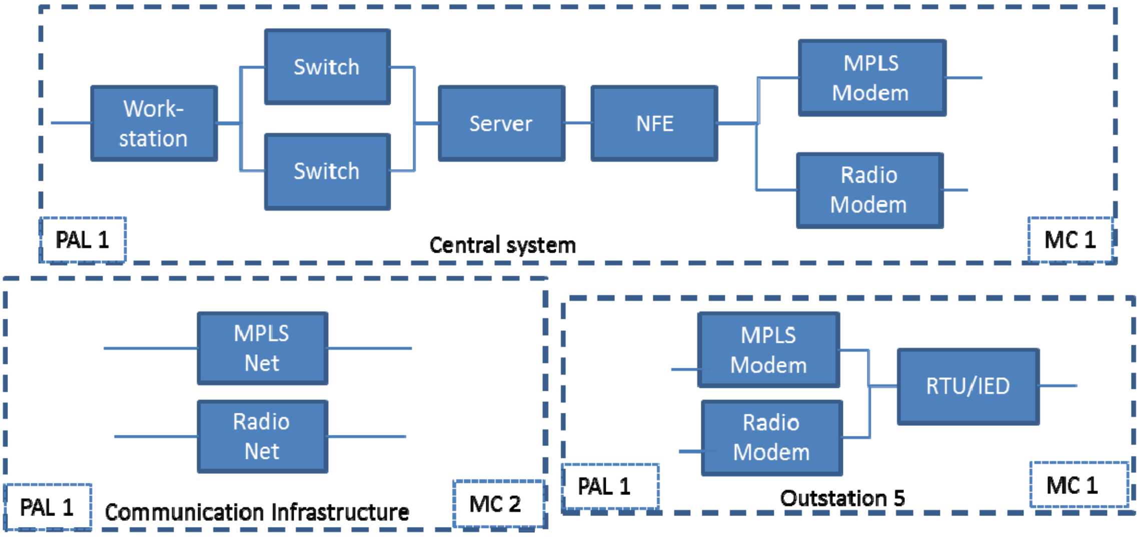


Figure 6: The structure, test 2, for the SCADA solution from the control room to the outstation. The structure is fully redundant in the communication segments of the solution.

Availability for the SCADA solution in test 1:

Availability for the SCADA solution in test 2:

Table 2: The difference between two structures in terms of availability

SCADA	Test 1	Test 12	Improvement	Improvement [min/year]
Total solution	**90,9172%**	**93,705%**	**3,066%**	**14 653**

The difference between two structures in terms of availability is described in Table 2. The improvement columns indicate the relative and absolute increase of availability between the SCADA solutions in Figure 5 and Figure 6.

3.2 Area Scope

The area scope focuses on a specific part of the system. In the performed test the focus area is the communication infrastructure and all three disciplines are considered. In this test the same set up as in the previous case is to be considered; the SCADA solution for Outstation 1 and 5. These structure models are presented in Figure 7 and Figure 8 respectively.

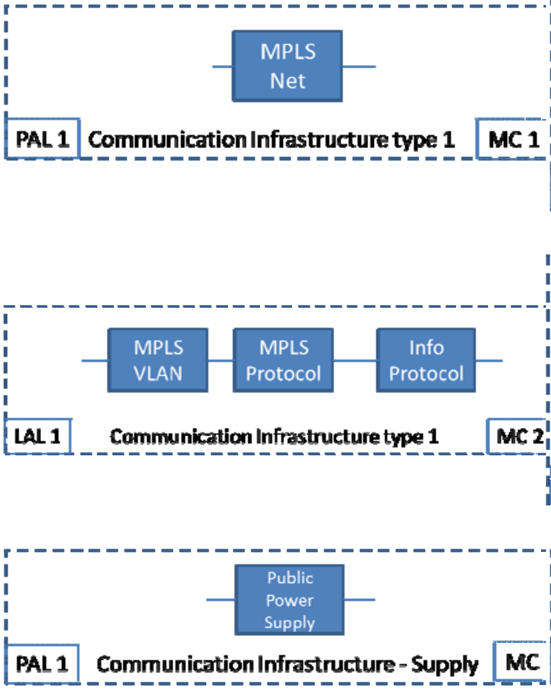

MPLS Net
PAL 1
Communication Infrastructure type 1
MC 1
MPLS VLAN
MPLS Protocol
Info Protocol
LAL 1
Communication Infrastructure type 1
MC 2
Public Power Supply
PAL 1
Communication Infrastructure - Supply
MC

Figure 7: Communication infrastructure for SCADA solution to outstation 1, test 3, for all three disciplines.

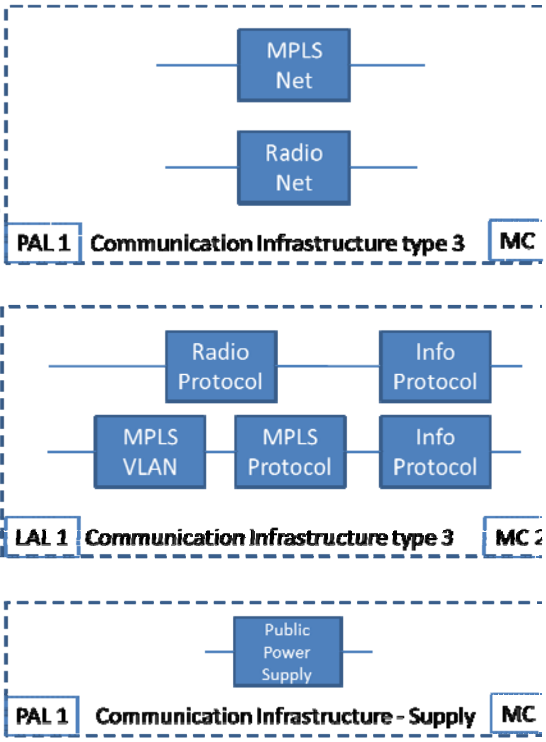


Figure 8: Communication infrastructure for SCADA solution to outstation 5, test 4, for all three disciplines.

These solutions have the structure functions for availability referred to as discipline function for test 3 and 4 respectively.

The discipline -function for test 3:

The discipline-function for test 4:

Table 3: The difference between the structures in test 3 and 4 in terms of availability

SCADA	Test 3	Test 4	Improvement	Improvement [min/year]

All three disciplines	**97,903%**	**99,895%**	**2,0352%**	**10 472**

In Table 3 the difference between the structures in test 3 and 4 in terms of availability is described. The improvement columns in the table indicate the relative and absolute increase of availability between the SCADA solutions in Figure 7 and Figure 8.

3.3 Full Top-Down Scope

The full top-down scope models the whole SCADA solution from the workstation in the control room to the RTU/IED of every outstation. This is the most comprehensive model scope and in this test one discipline will be considered; the physical. The highest level of abstraction will be used since there are such a large number of endpoints to consider. To have full functionality all endpoints must have connection all the way to the operator interface and this is the definition that will be used for this test.

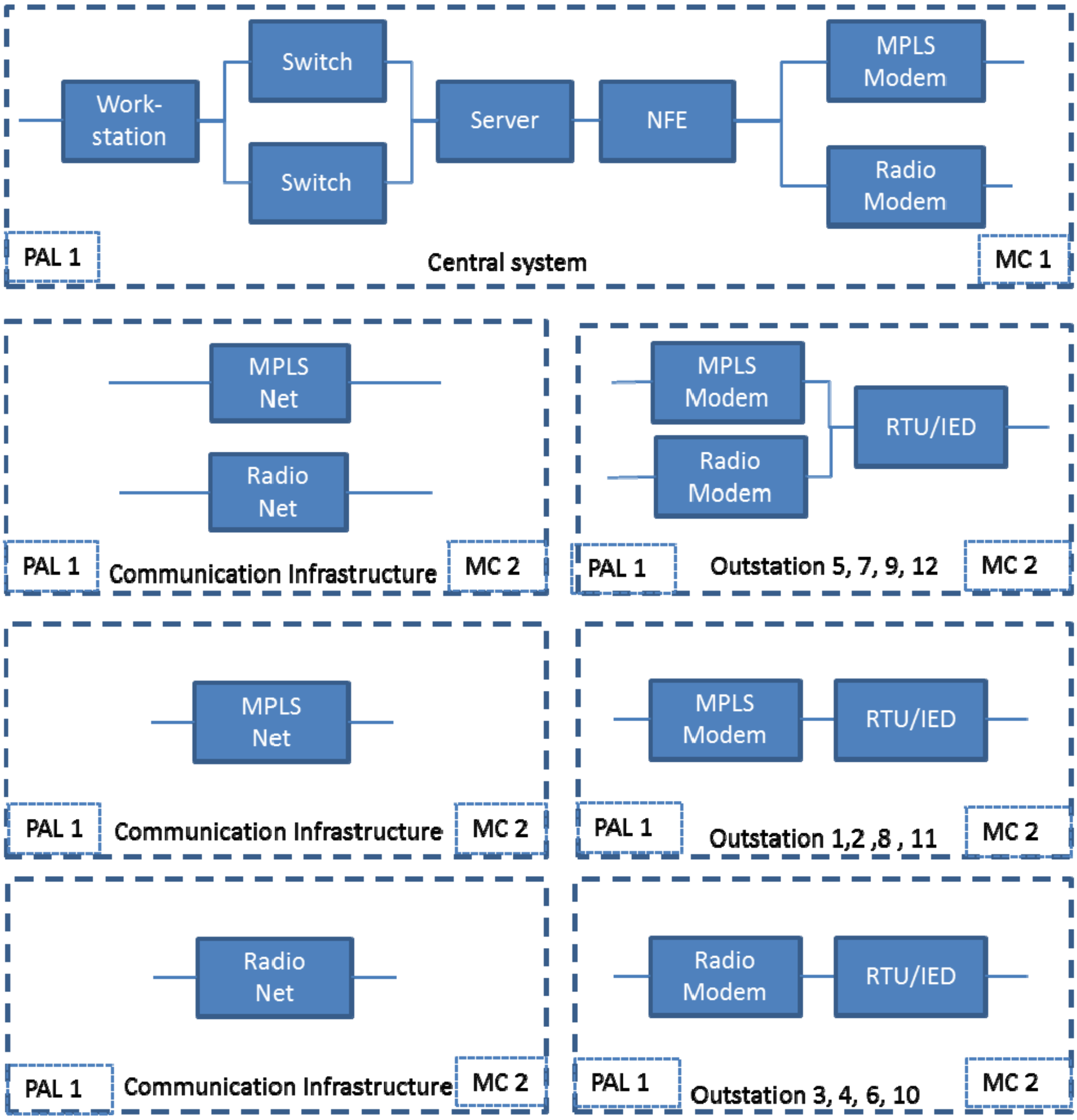


Figure 9: The SCADA solution model in a full-thread approach, in this case is tis simplified by using three general models for the outstations and the communication infrastructure.

In Figure 9, the model of the solution is displayed in the three different configurations. To be able to analyse a full-thread model simplifications are not allowed. In this case the reliability properties defined in Appendices I are the same and therefore studying these three configurations will provided the same result as studying each thread.

The availability functions for the solution have two different forms; one for the redundant solutions and one for the serial solutions.
Availability function, redundant solution:

$$\bar{A}_{Paralell} = \mathrm{A}_{Workstation}(1-(1-\mathrm{A}_{Switch})\cdot(1-\mathrm{A}_{Switch}))\cdot\mathrm{A}_{Server}\cdot\mathrm{A}_{NFE}\cdot (1-(1-\mathrm{A}_{MPLS\ Modem}\cdot\mathrm{A}_{MPLS\ Net}\cdot\mathrm{A}_{MPLS\ Modem})\cdot(1-\mathrm{A}_{Radio\ moddem}\cdot \mathrm{A}_{Radio\ Net}\cdot\mathrm{A}_{Radio\ Modem}))\ \cdot\mathrm{A}_{RTU}$$

Availability function, serial solutions:

$$\bar{A}_{Serial\ MPLS} = \mathrm{A}_{Workstation}\big(1-(1-\mathrm{A}_{Switch})\cdot(1-\mathrm{A}_{Switch})\big)\cdot\mathrm{A}_{Server}\cdot\mathrm{A}_{NFE}\cdot \mathrm{A}_{MPLS\ Modem}\cdot\mathrm{A}_{MPLS\ Net}\cdot\mathrm{A}_{MPLS\ Modem}\cdot\mathrm{A}_{RTU}$$

$$\bar{A}_{Serial\ Radio} = \mathrm{A}_{Workstation}\big(1-(1-\mathrm{A}_{Switch})\cdot(1-\mathrm{A}_{Switch})\big)\cdot\mathrm{A}_{Server}\cdot\mathrm{A}_{NFE}\cdot \mathrm{A}_{Radio\ Modem}\cdot\mathrm{A}_{Radio\ Net}\cdot\mathrm{A}_{Radio\ Modem}\cdot\mathrm{A}_{RTU}$$

To define the availability of each thread every communication path and piece of equipment in the outstations must have their reliability properties defined. All communication channels and outstation equipment have the same reliability properties, which give these three types of threads the same reliability.

This type of analysis displays the impact of structural decision for a solution. It is also possible to analyze the impact of equipment by altering the reliability properties of one or several object in the solution.

Table 4: The availability of the outstations

Test	Thread, outstation	Communication structure	MPLS channel	Radio channel	Availability Original	Improvement training
Test 5	1	Serial MPLS	1		90,6631%	0,00%
Test 5	2	Serial MPLS	2		90,6631%	0,00%
Test 5	3	Serial Radio		1	88,8699%	1,11%
Test 5	4	Serial Radio		2	88,8735%	1,11%
Test 5	5	Parallel	3	3	94,8867%	0,04%
Test 5	6	Serial Radio		3	88,8699%	1,11%
Test 5	7	Parallel	4	5	94,8867%	0,04%
Test 5	8	Serial MPLS	5		90,6631%	0,00%
Test 5	9	Parallel	6	6	94,8867%	0,04%
Test 5	10	Serial Radio		7	88,8699%	1,11%
Test 5	11	Serial MPLS	7		90,6631%	0,00%
Test 5	12	Parallel	8	8	94,8867%	0,04%

If the full- top-down result is available, the overall availability of the system could be calculated as an average of the availability of a thread. There is also the possibility of using the consequence measure defined for each outstation (thread) to analyze which actions are the most appropriate to take to improve the system.

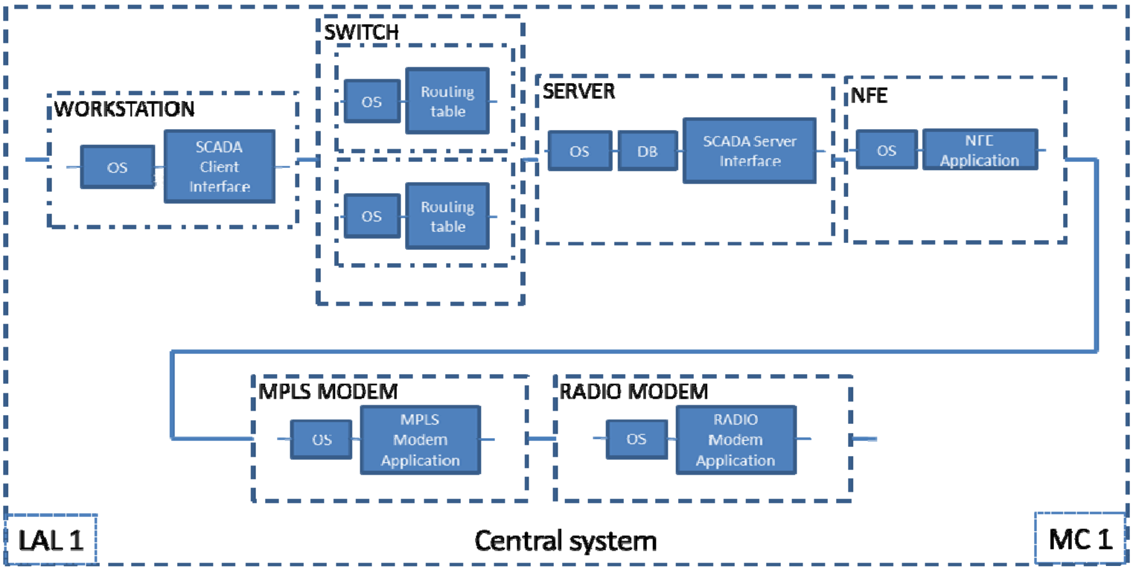
By combining these results with the taxonomy, it is also possible to locate soft areas for improvement; as education to minimize the impact of the lack of competence as it is one of the factors in reliability.

It is assumed that training of service personnel on administrating and configuring radio modems at the outstations will increase the reliability of these double folded; time to failure due to incompetence is doubled. The result of this is shown in Table 4. An improvement of 1% in annual availability is equal to 5 256 minutes a year of additional time in operations, which is the case for all the outstation that just utilize radio modems to communicate with the central system.

4 Test on model abstraction level

Dependent on which depth in the analysis is to have the abstraction level of the solutions model is to be considered. Utilizing a mode with higher abstraction level of the solution, or even part of the solution, enables a more detailed and more accurate description of the solution. However, it is only more accurate if the information on the system properties is reliable for that level of detail.

In this test the Logical discipline of the central system area is modeled and analyzed. In abstraction level two of the models, see Figure 10, the modems are not included. These are in this case modeled as for abstraction level one.



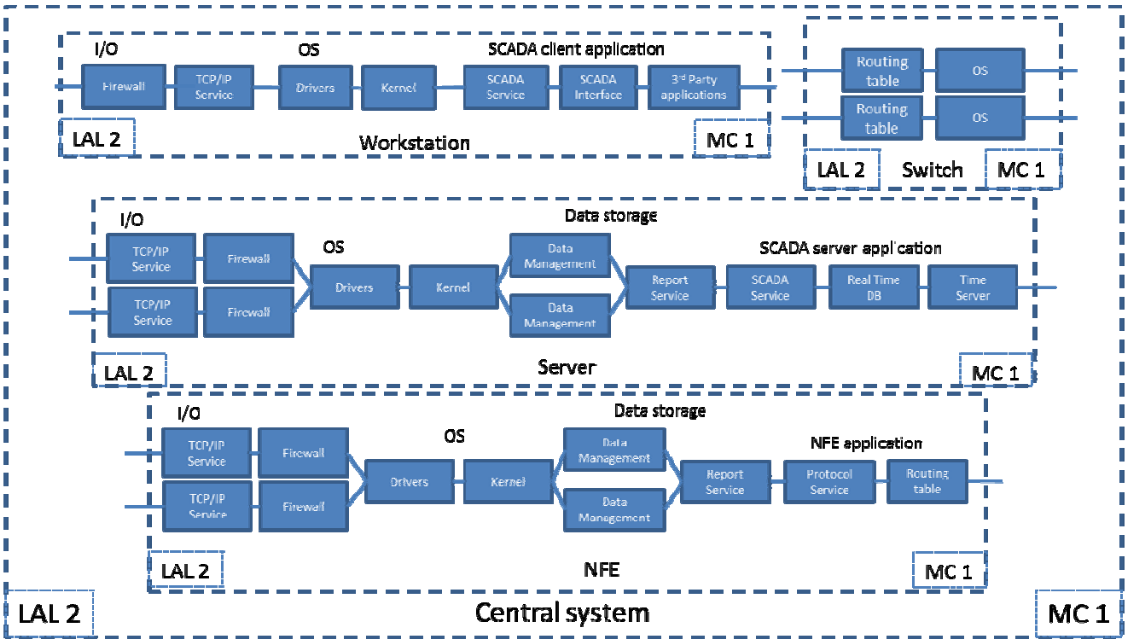


Figure 10: The logical discipline of the Central systems modelled with abstraction level one and two.

Availability function, Logical abstraction level one:

Availability function, Logical abstraction level two:

As the modems are modelled the same for both levels in this case, these are not considered in the availability functions.

The relevance of a more detailed abstraction level is highly dependent on quality of the objects data. In this test case the reliability data for the SCADA solution is set without any basis in actual empiric data. A comparison will not provide any additional information. More detailed abstraction levels requires more work to gather, store and manage object data as well as performing calculations of availability.

5 Further tests

5.1 The framework for reliability is tested with regards to analyzing systems from a reliability perspective. The framework also offers the possibility to compliment the analysis with financial data, competence data or consequence data that can introduce additional aspects to the analysis. Dependent on which type of analysis that is to be performed the functions differ.

5.2 Perspective functions

Here are examples of functions that can utilize the structure model to describe different perspectives of the SCADA solution:

$$f_{safety} = min x_i(\xi) \text{ ,} of\ all\ objects$$

$$f_{cost\ of\ purchase} = \sum_{i=1}^{n} x_i(\gamma) \text{ ,} where\ x_{i \to n}\ are\ all\ objects$$

$$f_{cost\ of\ compentence} = \sum_{i=1}^{n} \exists!\, x_i(\rho) \text{ ,} where\ x_{i \to n}\ are\ all\ objects$$

$$f_{average\ yearly\ cost\ of\ failure} = \sum_{j}^{m} x_j(\omega) * x_j(\psi) \text{ ,} where\ x_{j \to m}\ are\ the\ objects\ in\ outage$$

$$f_{consequence\ of\ failure} = \sum_{j}^{m} x_j(\lambda) * x_j(\mathrm{C}) \text{ ,} where\ x_{j \to m}\ are\ the\ objects\ in\ outage$$

If relevant attributes are defined it is possible to analyze the SCADA solution based on these values. In this test we will look at the average yearly restoration cost of failure. This is average yearly cost for mitigating errors for the components in a thread. The cost for loss of service is not included in this calculation; this is calculated by using the consequence of failure function.

This test will be performed by using the results from the Full top-down Scope test, see Table 4. To define the average cost, ω_P and ω_T, the taxonomy is utilized. The costs are not based on any real life data but assumed cost. The average cost, ω_P and ω_T, is defined in Appendices I.

Table 5: The table below displays the result of test 6 where the average yearly cost for maintain the SCADA solution to a specific outstation is calculated.

Test	Thread, outstation	Communication structure	MPLS channel	Radio channel	Average yearly restoration cost
Test 6	1	Serial MPLS	1		35,71 €
Test 6	2	Serial MPLS	2		35,71 €
Test 6	3	Serial Radio		1	35,96 €
Test 6	4	Serial Radio		2	35,96 €

Test 6	5	Parallel	3	3	41,89 €
Test 6	6	Serial Radio		3	35,96 €
Test 6	7	Parallel	4	5	41,89 €
Test 6	8	Serial MPLS	5		35,71 €
Test 6	9	Parallel	6	6	41,89 €
Test 6	10	Serial Radio		7	35,71 €
Test 6	11	Serial MPLS	7		35,71 €
Test 6	12	Parallel	8	8	41,89 €

In Table 5, the cost represents the total cost for maintaining the solution chosen for each respective SCADA solution for each specific outstation. All threads include the required set up for the central system to be fully functional. Therefore the redundant equipment for the central system is included in all the threads. This is somewhat misleading since this cost only occurs once and the cost for each thread cannot be added together to describe the total cost. The cost for the central system is 29.78 € a year and the total cost for the SCADA solution is 126.41 € a year. Keeping redundant threads full operational will require that all equipment is restored to operational status if they fail. Therefore the stations with redundant equipment have higher cost then the single equipment stations.

6 Conclusion of test

The test showed that it was possible to perform different types of analysis based on the suggested framework for a SCADA solution. Other observations are that to be able to perform such analysis certain prerequisites need to be fulfilled. Mainly the availability of information describing the SCADA solution is the defining factor for which analysis that can be performed. The work of defining a structure model for the SCADA solution can be performed on small as well as large systems; the amount of work however increases more than linearly as the systems size increases.

One key observation is that the model, with its structure function, of the current system can be created with reasonable effort but when analysis are performed the model needs to be modified. Depending on the modification setting up the alternative SCADA solution model can imply redoing most of the work done to set up the original model. Analyzing several possible setups will therefore be time consuming, especially if there is no available tool to support this. Support for storing the different objects in the SCADA solutions properties; i.e. attributes, is also something that would be helpful.

The tests also show that the quality of the data used in the analysis will have a large impact on the results of the analysis. As suggested in the framework and as performed in the tests, a relative analysis between two SCADA solutions is a mitigation that provides quantifiable results.

7 Appendices

Appendix I: Object attributes for test of framework for reliability analysis

ELFORSK

SVENSKA ELFÖRETAGENS FORSKNINGS- OCH UTVECKLINGS - ELFORSK - AB

Elforsk AB, 101 53 Stockholm. Besöksadress: Olof Palmes Gata 31
Telefon: 08-677 25 30, Telefax: 08-677 25 35
www.elforsk.se

Object	Attribute								
	λ	ψ	A	ωP	ωT	ωA			
Workstation	0,003352	2	0,993296	3000	500	2,8	PAL 1	TEST 1	MC 1
OS	0,0055176	1,75	0,990344			0	LAL 1	TEST 3	MC 1
Fiewall	0,002508	1	0,997492			0	LAL 2	TEST 4	MC 1
TCP/IP Service	0,002508	1	0,997492			0	LAL 2	TEST 4	MC 1
Drives	0,0002508	1	0,999749			0	LAL 2	TEST 4	MC 1
Kernel	0,0002508	4	0,998997			0	LAL 2	TEST 4	MC 1
SCADA Client inerface	0,038874	2	0,922252			0	LAL 1	TEST 3	MC 1
SCADA Service	0,001254	2	0,997492			0	LAL 2	TEST 4	MC 1
SCADA inerface	0,02508	2	0,94984			0	LAL 2	TEST 4	MC 1
3rd Par applications	0,01254	4	0,94984			0	LAL 2	TEST 4	MC 1
Switch	0,004561	2	0,990878	1000	300	2	PAL 1	TEST 1	MC 1
OS	0,0013794	4	0,994482				LAL 1	TEST 3	MC 1
Routing table	0,000627	4	0,997492				LAL 1	TEST 3	MC 1
Server	0,0023065	4	0,990774	7500	500	2,3	PAL 1	TEST 1	MC 1
OS	0,0055176	1,75	0,990344				LAL 1	TEST 3	MC 1
Fiewall	0,002508	1	0,997492				LAL 2	TEST 4	MC 1
TCP/IP Service	0,002508	1	0,997492				LAL 2	TEST 4	MC 1
Drives	0,0002508	1	0,999749				LAL 2	TEST 4	MC 1
Kernel	0,0002508	4	0,998997				LAL 2	TEST 4	MC 1
DB	0,011286	4	0,954856				LAL 1	TEST 3	MC 1
Data Managment	0,001254	4	0,994984				LAL 2	TEST 4	MC 1
Report Service	0,002508	4	0,989968				LAL 2	TEST 4	MC 1
SCADA Server interface	0,00627	4	0,97492				LAL 1	TEST 3	MC 1
SCADA Service	0,001254	4	0,994984				LAL 2	TEST 4	MC 1
Real Time DB	0,002508	4	0,989968				LAL 2	TEST 4	MC 1
Time Server	0,002508	4	0,989968				LAL 2	TEST 4	MC 1
Nework interface	0,00086	2	0,99828	250	100	0,1	PAL 2	TEST 2	MC 1
Motherboard CPU	0,0003415	8	0,997268	400	100	0	PAL 2	TEST 2	MC 1
HDD	0,0011425	2	0,997715	1000	150	0,2	PAL 2	TEST 2	MC 1
NFE	0,002344	4	0,990624	3500	200	1	PAL 1	TEST 1	MC 1
OS	0,0013794	1,75	0,997586				LAL 1	TEST 3	MC 1
Fiewall	0,002508	1	0,997492				LAL 2	TEST 4	MC 1
TCP/IP Service	0,002508	1	0,997492				LAL 2	TEST 4	MC 1
Drives	0,0002508	1	0,999749				LAL 2	TEST 4	MC 1
Kernel	0,0002508	4	0,998997				LAL 2	TEST 4	MC 1
NFE Application	0,002508	4	0,989968				LAL 1	TEST 3	MC 1
Protocol Service	0,002508	4	0,989968				LAL 2	TEST 4	MC 1
Routin Table	0,00627	4	0,97492				LAL 2	TEST 4	MC 1
3rd Par applications	0,01254	4	0,94984				LAL 2	TEST 4	MC 1
Nework Interface	0,00086	2	0,99828	250	100	0,1	PAL 2	TEST 2	MC 1
Motherboard CPU	0,0003415	8	0,997268	400	100	0	PAL 2	TEST 2	MC 1
HDD	0,0011425	2	0,997715	1000	150	0,2	PAL 2	TEST 2	MC 1
Radio modem	0,003115	2	0,99377	2000	200	0,8	PAL 1	TEST 1	MC 1
OS	0,0013794	1	0,998621				LAL 1	TEST 3	MC 1
Radio Modem application	0,01254	4	0,94984				LAL 1	TEST 3	MC 1
MPLS modem	0,0015575	2	0,996885	2000	200	0,4	PAL 1	TEST 1	MC 1
OS	0,0013794	1	0,998621				LAL 1	TEST 3	MC 1
MPLS Modem application	0,01254	4	0,94984				LAL 1	TEST 3	MC 1

MPLS Net 1	0,00205	8	0,98360				PAL 1	TEST 1	MC 1
MPLS Protocol	0,002508	1	0,99749				LAL 1	TEST 3	MC 1
MPLS VLAN	0,002508	1	0,99749				LAL 1	TEST 3	MC 1
MPLS Net 2	0,00205	8	0,98360				PAL 2	TEST 1	MC 1
MPLS Protocol	0,002508	1	0,99749				LAL 1	TEST 3	MC 1
MPLS VLAN	0,002508	1	0,99749				LAL 1	TEST 3	MC 1
MPLS Net 3	0,00205	8	0,98360				PAL 3	TEST 1	MC 1
MPLS Protocol	0,002508	1	0,99749				LAL 1	TEST 3	MC 1
MPLS VLAN	0,002508	1	0,99749				LAL 1	TEST 3	MC 1
MPLS Net 4	0,00205	8	0,98360				PAL 4	TEST 1	MC 1
MPLS Protocol	0,002508	1	0,99749				LAL 1	TEST 3	MC 1
MPLS VLAN	0,002508	1	0,99749				LAL 1	TEST 3	MC 1
MPLS Net 5	0,00205	8	0,98360				PAL 5	TEST 1	MC 1
MPLS Protocol	0,002508	1	0,99749				LAL 1	TEST 3	MC 1
MPLS VLAN	0,002508	1	0,99749				LAL 1	TEST 3	MC 1
MPLS Net 6	0,00205	8	0,98360				PAL 6	TEST 1	MC 1
MPLS Protocol	0,002508	1	0,99749				LAL 1	TEST 3	MC 1
MPLS VLAN	0,002508	1	0,99749				LAL 1	TEST 3	MC 1
MPLS Net 7	0,00205	8	0,98360				PAL 7	TEST 1	MC 1
MPLS Protocol	0,002508	1	0,99749				LAL 1	TEST 3	MC 1
MPLS VLAN	0,002508	1	0,99749				LAL 1	TEST 3	MC 1
MPLS Net 8	0,00205	8	0,98360				PAL 8	TEST 1	MC 1
MPLS Protocol	0,002508	1	0,997492				LAL 1	TEST 3	MC 1
MPLS VLAN	0,002508	1	0,997492				LAL 1	TEST 3	MC 1
Radio Net 1	0,00400	8	0,96804				PAL 1	TEST 1	MC 1
Info Protocol	0,00400	1	0,99601				LAL 1	TEST 3	MC 1
Radio proocol	0,00400	1	0,99601				LAL 1	TEST 3	MC 1
Radio Net 2	0,00399	8	0,96808				PAL 2	TEST 1	MC 1
Info Protocol	0,00400	1	0,99601				LAL 1	TEST 3	MC 1
Radio proocol	0,00400	1	0,99601				LAL 1	TEST 3	MC 1
Radio Net 3	0,00400	8	0,96804				PAL 3	TEST 1	MC 1
Info Protocol	0,00400	1	0,99601				LAL 1	TEST 3	MC 1
Radio proocol	0,00400	1	0,99601				LAL 1	TEST 3	MC 1
Radio Net 4	0,00400	8	0,96804				PAL 4	TEST 1	MC 1
Info Protocol	0,00400	1	0,99601				LAL 1	TEST 3	MC 1
Radio proocol	0,00400	1	0,99601				LAL 1	TEST 3	MC 1
Radio Net 5	0,00400	8	0,96804				PAL 5	TEST 1	MC 1
Info Protocol	0,00400	1	0,99601				LAL 1	TEST 3	MC 1
Radio proocol	0,00400	1	0,99601				LAL 1	TEST 3	MC 1
Radio Net 6	0,00400	8	0,96804				PAL 6	TEST 1	MC 1
Info Protocol	0,00400	1	0,99601				LAL 1	TEST 3	MC 1
Radio proocol	0,00400	1	0,99601				LAL 1	TEST 3	MC 1
Radio Net 7	0,00400	8	0,96804				PAL 7	TEST 1	MC 1
Info Protocol	0,00400	1	0,99601				LAL 1	TEST 3	MC 1
Radio proocol	0,00400	1	0,99601				LAL 1	TEST 3	MC 1
Radio Net 8	0,00400	8	0,96804				PAL 8	TEST 1	MC 1
Info Protocol	0,002508	1	0,997492				LAL 1	TEST 3	MC 1
Radio proocol	0,002508	1	0,997492				LAL 1	TEST 3	MC 1
Radio modem 1	0,003115	8	0,97508	1500	200	0,8	PAL 1	TEST 1	MC 1
OS	0,0013794	1	0,998621				LAL 1	TEST 3	MC 1

Radio Modem application	0,00627	4	0,97492				LAL 1	TEST 3	MC 1
Radio modem 2	0,003115	8	0,97508	1500	200	0,8	PAL 2	TEST 1	MC 1
OS	0,0013794	1	0,998621				LAL 1	TEST 3	MC 1
Radio Modem application	0,002508	4	0,989968				LAL 1	TEST 3	MC 1
Radio modem 3	0,003115	8	0,97508	1500	200	0,8	PAL 3	TEST 1	MC 1
OS	0,002508	1	0,997492				LAL 1	TEST 3	MC 1
Radio Modem application	0	4	1				LAL 1	TEST 3	MC 1
Radio modem 4	0,003115	8	0,97508	1500	200	0,8	PAL 4	TEST 1	MC 1
OS	0,002508	1	0,997492				LAL 1	TEST 3	MC 1
Radio Modem application	0,002508	4	0,989968				LAL 1	TEST 3	MC 1
Radio modem 5	0,003115	8	0,97508	1500	200	0,8	PAL 5	TEST 1	MC 1
OS	0,002508	1	0,997492				LAL 1	TEST 3	MC 1
Radio Modem application	0,0002508	4	0,998997				LAL 1	TEST 3	MC 1
Radio modem 6	0,003115	8	0,97508	1500	200	0,8	PAL 6	TEST 1	MC 1
OS	0,002508	1	0,997492				LAL 1	TEST 3	MC 1
Radio Modem application	0,00627	4	0,97492				LAL 1	TEST 3	MC 1
Radio modem 7	0,003115	8	0,97508	1500	200	0,8	PAL 7	TEST 1	MC 1
OS	0,002508	1	0,997492				LAL 1	TEST 3	MC 1
Radio Modem application	0	4	1				LAL 1	TEST 3	MC 1
Radio modem 8	0,003115	8	0,97508	1500	200	0,8	PAL 8	TEST 1	MC 1
OS	0,002508	1	0,997492				LAL 1	TEST 3	MC 1
Radio Modem application	0,0013794	4	0,994482				LAL 1	TEST 3	MC 1
MPLS modem 1	0,003005	8	0,97596	1500	200	0,7	PAL 6	TEST 1	MC 1
OS	0,000105	1	0,999895				LAL 1	TEST 3	MC 1
MPLS Modem application	0,000105	4	0,99958				LAL 1	TEST 3	MC 1
MPLS modem 2	0,003005	8	0,97596	1500	200	0,7	PAL 7	TEST 1	MC 1
OS	0,000105	1	0,999895				LAL 1	TEST 3	MC 1
MPLS Modem application	0,000105	4	0,99958				LAL 1	TEST 3	MC 1
MPLS modem 3	0,003005	8	0,97596	1500	200	0,7	PAL 8	TEST 1	MC 1
OS	0,000105	1	0,999895				LAL 1	TEST 3	MC 1
MPLS Modem application	0,000105	4	0,99958				LAL 1	TEST 3	MC 1
MPLS modem 4	0,003005	8	0,97596	1500	200	0,7	PAL 9	TEST 1	MC 1
OS	0,000105	1	0,999895				LAL 1	TEST 3	MC 1
MPLS Modem application	0,000105	4	0,99958				LAL 1	TEST 3	MC 1
MPLS modem 5	0,003005	8	0,97596	1500	200	0,7	PAL 10	TEST 1	MC 1
OS	0,000105	1	0,999895				LAL 1	TEST 3	MC 1
MPLS Modem application	0,000105	4	0,99958				LAL 1	TEST 3	MC 1
MPLS modem 6	0,003005	8	0,97596	1500	200	0,7	PAL 11	TEST 1	MC 1
OS	0,000105	1	0,999895				LAL 1	TEST 3	MC 1
MPLS Modem application	0,000105	4	0,99958				LAL 1	TEST 3	MC 1
MPLS modem 7	0,003005	8	0,97596	1500	200	0,7	PAL 12	TEST 1	MC 1
OS	0,002508	1	0,997492				LAL 1	TEST 3	MC 1
MPLS Modem application	0,002508	4	0,989968				LAL 1	TEST 3	MC 1
MPLS modem 8	0,003005	8	0,97596	1500	200	0,7	PAL 13	TEST 1	MC 1
OS	0,0013794	1	0,998621				LAL 1	TEST 3	MC 1
MPLS Modem application	0,00627	4	0,97492				LAL 1	TEST 3	MC 1
RTU/IED 1	0,003516	8	0,971872	2400	200	1,2	PAL 1	TEST 1	MC 1
OS	0,003516	1	0,996484				LAL 1	TEST 3	MC 1
RTU/IED application	0,003516	4	0,985936				LAL 1	TEST 3	MC 1
RTU/IED 2	0,003516	8	0,971872	2400	200	1,2	PAL 2	TEST 1	MC 1

OS	0,003516	1	0,996484				LAL 1	TEST 3	MC 1
RTU/IED application	0,003516	4	0,985936				LAL 1	TEST 3	MC 1
RTU/IED 3	0,003516	8	0,971872	2400	200	1,2	PAL 3	TEST 1	MC 1
OS	0,003516	1	0,996484				LAL 1	TEST 3	MC 1
RTU/IED application	0,003516	4	0,985936				LAL 1	TEST 3	MC 1
RTU/IED 4	0,003516	8	0,971872	2400	200	1,2	PAL 4	TEST 1	MC 1
OS	0,003516	1	0,996484				LAL 1	TEST 3	MC 1
RTU/IED application	0,003516	4	0,985936				LAL 1	TEST 3	MC 1
RTU/IED 5	0,003516	8	0,971872	2400	200	1,2	PAL 5	TEST 1	MC 1
OS	0,003516	1	0,996484				LAL 1	TEST 3	MC 1
RTU/IED application	0,003516	4	0,985936				LAL 1	TEST 3	MC 1
RTU/IED 6	0,003516	8	0,971872	2400	200	1,2	PAL 6	TEST 1	MC 1
OS	0,003516	1	0,996484				LAL 1	TEST 3	MC 1
RTU/IED application	0,003516	4	0,985936				LAL 1	TEST 3	MC 1
RTU/IED 7	0,003516	8	0,971872	2400	200	1,2	PAL 7	TEST 1	MC 1
OS	0,003516	1	0,996484				LAL 1	TEST 3	MC 1
RTU/IED application	0,003516	4	0,985936				LAL 1	TEST 3	MC 1
RTU/IED 8	0,003516	8	0,971872	2400	200	1,2	PAL 8	TEST 1	MC 1
OS	0,003516	1	0,996484				LAL 1	TEST 3	MC 1
RTU/IED application	0,003516	4	0,985936				LAL 1	TEST 3	MC 1
RTU/IED 9	0,003516	8	0,971872	2400	200	1,2	PAL 9	TEST 1	MC 1
OS	0,003516	1	0,996484				LAL 1	TEST 3	MC 1
RTU/IED application	0,003516	4	0,985936				LAL 1	TEST 3	MC 1
RTU/IED 10	0,003516	8	0,971872	2400	200	1,2	PAL 10	TEST 1	MC 1
OS	0,003516	1	0,996484				LAL 1	TEST 3	MC 1
RTU/IED application	0,003516	4	0,985936				LAL 1	TEST 3	MC 1
RTU/IED 11	0,003516	8	0,971872	2400	200	1,2	PAL 11	TEST 1	MC 1
OS	0,003516	1	0,996484				LAL 1	TEST 3	MC 1
RTU/IED application	0,003516	4	0,985936				LAL 1	TEST 3	MC 1
RTU/IED 12	0,003516	8	0,971872	2400	200	1,2	PAL 12	TEST 1	MC 1
OS	0,002508	1	0,997492				LAL 1	TEST 3	MC 1
RTU/IED application	0,002508	4	0,989968				LAL 1	TEST 3	MC 1
Public Power supply	0,001	4	0,996				PAL 1	Test 7	MC 1
Disel generator	0,01	2	0,98				PAL 1	Test 7	MC 1
UPS Public colling and heating	0,0001	2	0,9998				PAL 1	Test 7	MC 1
Cooling with Public Water	0,0001	2	0,9998				PAL 1	Test 7	MC 1