UNIVERSITY OF VAASA

FACULTY OF TECHNOLOGY

COMMUNICATIONS AND SYSTEMS ENGINEERING

USER INTERFACE CONCEPTION FOR ASSET MANAGEMENT SYSTEM

Xiaoguo Xue

July 7, 2015

Master´s thesis for the degree of Master of Science in Technology submitted for inspection, Vaasa, 7th July, 2015.

Supervisor Mohammed Salem Elmusrati

Instructor Reino Virrankosiki

ACKNOWLEDGEMENTS

This thesis work is a done in a joint research project between University of Vaasa and Wärtsilä Finland Oy. Therefore, first and foremost I would like to express my deepest gratitude to my supervisor Professor Mohammed Salem Elmusrati and my instructor Reino Virrankoski, for their dedicated guidance, patience and constant support throughout the whole work.

Then, I would like to give my sincere thanks to my instructors from Wärtsilä: Jonatan Rösgren, Patrik Selin and Matias Aura. Without their support and help, this work can never be done. There are far too many people to thank from Wärtsilä, in order to not forgetting anyone, I would like to thank all the experts I have been interviewed, all the staff from the assembling line and all the personnel that have helped me.

Special thanks to Tero Frondelius and Jukka Aho for their elaborate help and cooperation; special thanks to Patrik Selin's and Markku Mäenpää's teams, for providing me a pleasant working environment; special thanks to Tobias Glocker for providing theory and material help.

At last, I would like to thank my family members for all the support to my study and life.

Xiaoguo Xue

Vaasa, Finland, 7th July 2015

Contents

List of Figures

List of Tables

Abbreviations

3GPP	3rd Generation Partnership Project
API	Application programming interface
BDC	Bottom dead center
BI	Business Intelligent
CAN	Control Area Network
CBM	Condition Based Maintenance
CIP	Common Industry Protocol
CSV	Comma-separated values
DBMS	Database Management System
DCT	Discrete Cosines Transformation
DST	Discrete Sine transform
FFT	Fast Fourier Transform
GPRS	General Packet Radio Services
GUI	Graphical user interface
HDF	Hierarchical Data Format
HDFS	Hadoop Distributed File System
HTML	HyperText Markup Language
ICD	Intelligent Communicatios Director
IDM	Integrated document management
JSON	JavaScript Object Notation
IoT	Internet of Things
MAC	Media Access Control
MFI	Multiple port fuel injection

PFI	Port fuel injection
PLOT	Piecewise Linear Online Trending
RDBMS	Relational Database Management System
REST	Representational State Transfer
RRD	Round-Robin Database
SDT	Swinging Door Trending
SOAP	Simple Object Access Protocol
TDC	Top dead center
TSDS	Time Series Database Servers
URL	Uniform resource locator
VLT	Visible Light Transmission
W.O.	Wärtsilä Optimisers
WAN	Wide area network
WT	Wavelet Transform

UNIVERSITY OF VAASA
Faculty of Technology Author: Xiaoguo Xue
Topic of the Thesis: User Interface Conception For Asset Management System
Supervisor: Mohammed Elmusrati
Instructor: Reino Virrankoski
Degree: Master of Science in Technology
Department: Department of Computer Science
Degree programme: Master Programme in Communications and Systems Engineering
Major Subject: Communications and Systems Engineering
Year of Entering University: 2012
Year of Completing the Thesis: 2015 Pages: 99

ABSTRACT

Data as a critical resource nowadays, it is growing expotentially. As a consequence, the issue of data storing, filtering, processing and analysis have attracted a lot of attention in the database industry, since they can not be handled by the traditional database systems. The new situation has been discussed as a concept of Big Data. To be able to handle and process the Big Data, one muct be capable of deal with large volume, high velocity and various types data. With this trend, Relational Database also succeeded in integrating parts of the functions which are required to handle Big Data. So it becomes that those two techniques advance side by side.

Here in this work both Big Data and Relational Database are discussed based on the current database industry development situation. Moreover, data mining techniques and communication standards are also studied and discussed to give directions for further implementation work. An interfacing work, which includes both software interfacing for third party user and user interface design and implementation, is done to provide Wärtsilä internal users access to their remote monitoring data. The desing work is done based on the needs of the different user groups of the personnel of Wärtsilä.

KEYWORDS Data Mining, Big Data, Relational Database

Chapter 1

INTRODUCTION

Data mining and analysis have attracted more and more attention in the information industry in recent years, because of the rapid development of the technology of collecting and storing huge amounts of data, and the needs to figure out the useful information and knowledge from the collected data.

Since 1970s computer system industry has gone through an evolutionary path in the development of the following areas: data collecting and database creation, data storage, data retrieval, database transaction processing and data warehousing and data mining (Ye, 2014; Kennesaw, 2010) as showed in Figure1.1

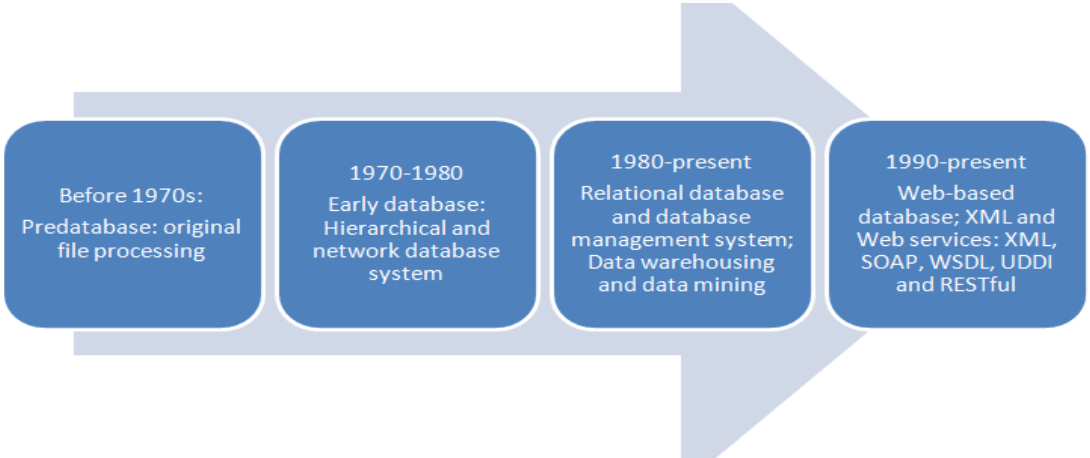


Figure 1.1: Data collecting and analysis technology development.

Data collecting techniques have been developed from expensive file based limited space storage to enormous amount of cheap database management system (DBMS) and web services, while the pace of discovering useful information from massive data falls far behind the efficiency of data collecting techniques. However because of its huge potential usage in commercial and society development, the process of the knowledge discovering has become more and more popular. Therefore, the techniques to extract useful information from the massive amount of data have become a critical bottleneck in the application development.

Data warehousing and data mining are the results of natural evolution of information technology and computer hardware technology. They both are techniques for data analysis.

Data warehousing is the process of aggregating data from multiple sources into one common repository. The data repository is usually maintained separately from operational database. Data mining is the process of finding patterns from the data set and interpreting those data patterns into useful information. In general, data warehousing is the process of compiling and combining data into one common database; data mining is the process of extracting meaningful phenommenas, features, patterns etc, from the collected data. Datawarehouing occurs before data mining. However, nowadays most of the data warehousing processes also include some kinod of preprocessing of the data. Therefore when talking about data mining, the data warehousing must also be considered.

1.1. Data Mining

With the rapid development of the digitalized measurement and data aggregation systems, industry has showed huge interest in developing data mining techniques. Especially during recent years, an increasing amount of data can be collected from industrial systems and processes. Companies are awared about the additional market value that can be created and added by the efficient utilization of the data. As a consequence, it is becoming more and more critical to know how to utilize the massive data and extract meaningful information from it. Data mining is closely associated with a number of areas such as database systems, data filtering, data integration, data transformation, pattern discovery, pattern evaluation and knowledge presentation. (Han *et al.*, 2012)

Data mining is a technique which targets to find the potential or hidden interesting infor-

mation from vast amount of data. It consists of the following processes: (Padhy *et al.*, 2012) as showed in Figure1.2 (UNIQ, 2014)

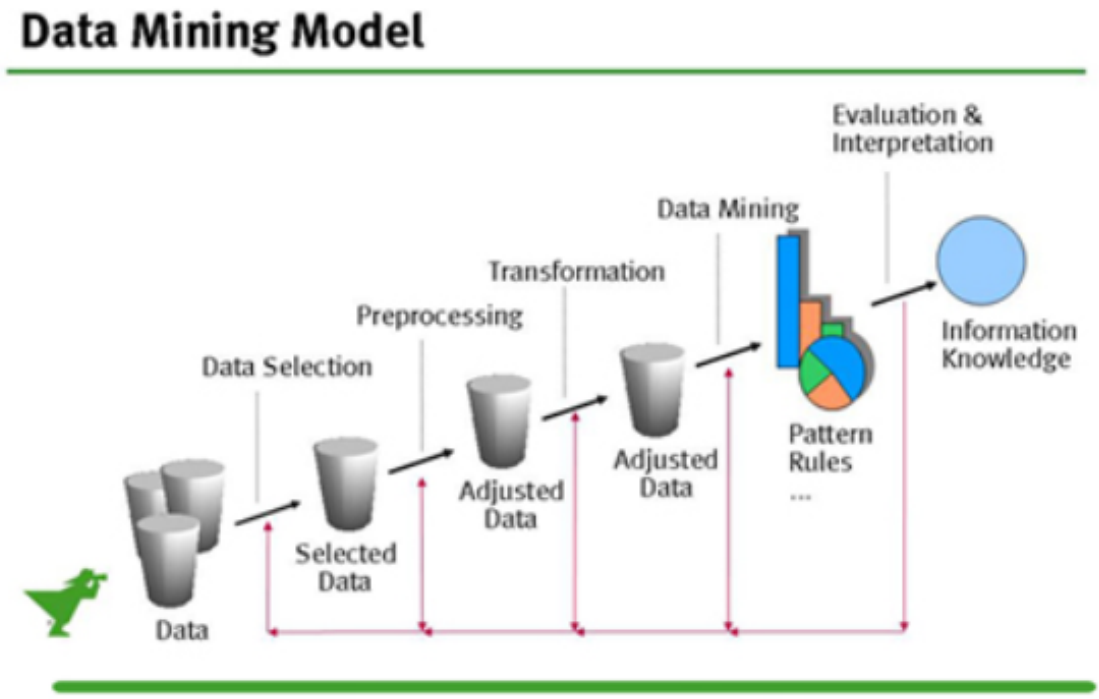


Figure 1.2: Data mining process.

- Business understanding or propose understanding; find out what are the objectives and requirements, then convert this knowledge into data mining problem definition.
- Understanding the data; process of collecting correct data for mining.
- Data preparation; usually the raw data must be preprocessed before further analysis.
- Model building; select the modelling techniques which will be applied. Selection depends strongly on the tasks and targets of data mining.
- Evaluation; evaluate how well the model satisfies the originally-stated target of the data analysis once the results are available.
- Deployment: insight and actionable information can be derived from the data mining.

Data mining in general can be divided into two categories of problems: descriptive data mining and predictive data mining. Descriptive data mining is used to describe the general properties of the data, while predictive data mining can be used to predict the future based on the available data.

1.1.1. Data Source

Data mining relies heavily on data, and for different purposes different types of data are needed. Some of the useful ways to store the collected data are: (et al Int, 2014)

- Flat file, for researchers this is one of the most usual ways to store the data. It normally has text or binary format with a known structure.
- Relational database, it is a set of tables with values of entity attributes or values of attributes from entity relationships.
- Transaction database, it is a set of records representing transactions, each with a time stamp and identifier and other items.
- Multimedia database, this contains video, image, audio and text meida, which makes data mining more challenge.
- Spatial database which contains not only the usual data, but also the geographical information.
- Time series database, this is a database which contains time related data, such as stock market data.
- World Wide Web, which needs the most dynamic repositor.

As we can see, data can be stored in various formats, and it must be taken into account in data mining.

Moreover, data mining can be divided into supervised and unsurpervised learning models. Supervised data mining tries to infer a relationship or function based on the training data and use this function to predict the output variables. Unsupervised data mining targets to find patterns from the data based on the relationship existing in the data.

An alternative way to classify data mining problems is proposed as: (Kotu & Deshpande, 2015)

- Data classification is to predict if a data point belongs to one of the predefined classes.
- Regression, which is the process of predicting the numeric target label of a data point.
- Anomaly detection is the process of predicting if a data point is an outlier compared with other data points in the data set.
- Time series is the function of predicting the value of the target variable in a future based on previous values.
- Clustering is the process of identifying natural clusters with the data set based on inherit properties within the data set.
- Association analysis, which is the process to identify the relationships within an item set based on transaction data.

Different algorithms can be used for different types of tasks in order to find the valuable information from the whole data set.

1.1.2. Extracting the Data

The first step in making the best use of any data source is to understand how the data is gathered and managed. Data extracting is the process of extracting data from a source system for further process. There are two widely used extraction methods: (Bhaskar, 2015)

- Full extraction: all the data available in the source system are extracted.
- Incremental extraction: only the data that has changed since last successful extraction are extracted. It is a key-enabling technology for providing near real-time or on-time data warehousing. This method is also called Change Data Capture, because for this method it is critical to identify the changes. Timestamps, triggers and Partitioning etc are some of the common techniques for self-developed change capture.

These two methods are both designed to be used for relational database table. Then it depends on the application type and purpose, which methods can be used.

After data is extracted from source system, it is stored in a temporary data storage. Then the data is reconstructed and uploaded into data warehouse.

1.1.3. Preprocessing the Data

There are many factors comprising data quality including accuracy, completeness, consistency, timeliness, believability and interpretability.(Han *et al.*, 2012) Realistic data includes always noisy, missing and inconsistent measurements. As a consequence, the data must be preprocessed before performing any further operations.

The major tasks in data preprocessing are:

- Data cleaning which is used to estimate and fill in the missing values, filter noisy data, detect and remove outliers and correct inconsistencies.
- Data fusion has the function of utilizing multiple data sources into a uniform data repository.
- Data compression can reduce data size by clustering or eliminating redundant features etc.
- Data transformation, is the process of data normalization and aggregation.

Those techniques are not mutually exclusive, many of them can be used depending on the application.

1.1.4. Data Cleaning

Based on the target of data cleaning, it can be divided into two operations:

- Estimating and completing the missing values
- Filtering the noisy data

It is common that in real measurement data, some values are missing from database. After detecting the missing values, they can be either estimated or just marked. In time series analysis, there are many methods, from the simple ones to the more computation intensive to estimate the missing measurement values. If the missing values are just marked, there is a certain variable which is used for that purpose in the data sorting system architecture.

Measurement data is always noisy in some level. The first and most important method to reduce the effect of noise is the filtering of the measurement data. It can be done in several places and in several levels of the measurement system architecture. For example, first on the sensor level, then on the local area network gateway level and finally on the level of the centralized database system.

In the processes of data cleaning, different methods can lead to apparent different working load for the system, and in the worst case the whole data set may become distorted because of the use of wrong data cleaning methods. Therefore, it is very important to select the right methods.

1.1.5. Data Fusion

The integration of data and knowledge from several sources is known as data fusion. It is often required in data mining, because we may have several sources of measurement data, and we have to avoid redudancies and inconsistencies in the resulting dataset. And the metadata can be used to help avoid errors. The available data fusion techniques can be clarified into three categories: data association, state estimation and decision fusion. There are many algorithms available such as Dasarathy's Classfication, JDL data fusion classification, Nearest Neighbors and K-Means, Probabilistic data association etc. Sensor fusion is also well known as data fusion and is a subset of information fusion.

In the geospatial domain, data fusion is often synonymous with data integration(Wikipedia, 2015c). In those kinds of application, diverse data sets are combined into a unified data set, which includes all the information from the input data sets. In applications outside of the geospatial domain, data fusion and data integration are different with each other, where data integration is used to describe the combining of data, whereas data fusion is data integration and reduction or replacement (Wikipedia, 2015c).

1.1.6. Data Compression

Data transmission and storage cost money, so the more information is needed to be handled with, the more it costs. Moreover nowadays the amount of data is growing exponentially. Therefore data compression can not only save budget, but also improve the efficiency in doing data mining by using a small volume data which still produces the same result as the whole data set. Data compression is the technique that can be used to obtain

a reduced version of the dataset which has smaller volume, and still sustain the integrity of the original data. Data compression can be approached by following methods:

- Dimensionlity reduction: selecting a minimum number of variables or attributes, so some unimportant attributes can be removed. Here many methods can be used for example Attribute selection, Wavelet transforms and Principal components analysis, which can detect and remove the irrelevant, weakly relevant or redundant attributes.
- Numerosity reduction: choosing alternative, smaller forms of data representation, it can be parameteric method where only model parameters are stored, or non-parametric method where histograms, clustering etc are stored.
- Data compression in such a way that the original data can be represented. However it is possible that some features of the original dara are lost when reconstructing back to original data.

Data compression is commonly used in all field from media to entertainment, from industry to social networking.

Basically data compression can be classified as lossless and lossy methods. With lossless techniques the restored data is identical to the original information, so normally this is used in the applications which have strict requirement of data quality. Lossy compression applies for applications which do not require a complete restoring of the original data. The acceptable loss in quality depends on the particular application. However when talking about time series or process data, lossy techniques are more commonly used.

And data compression for time series or process data can be divided into three methods:

- Direct method or time domain method which is done by utilizing the subset of significant samples from the original sample set, such as Swinging Door Trending(SDT), PLOT etc.
- Transformational method first needs to transform the signal, then perform spectral and enery distribution and analysis. So the compression is done in the transformed domain. Commonly used techniques are discrete cosines transformation (DCT), fast fourier transform (FFT), discrete sine transform (DST), wavelet transform (WT) etc. (Chhipa, 2013) Transform method is not real-time, it requires historical data.

Table 1.1: A comparison of the compression methods.

Methods	Pros	Conns
Direct compression method	Simple system, fast and low error	Suffer from sentitiveness to sampling rate and high frequnency interference. Fail to achieve high data rate
Transformational method	high compression ratio	Complicated, slow and high error

From the table we can see every method has its strength and weakness, so which method is better depends on the application and requirement. Some cases need highly compressed data, then transformed method could be one good option, while if high data quality is requested, direct method is better to use. With time series or process data, normally direct method is prefered because of its simplification, fast and high quality.

Nowadays, sensors can produce fixed-width segments by recording values at regular time intervals or only to record new value when it differs from the previous record by a certain minimum amount. So with fix-width segments, every data at sampling frequnecy is stored without time stamp, so there is no data compression; while for unfixed case, data is compressed.

- In current industry market SDT is widely used, however this technique depends on system configuration, sample rate and signal condition, so the space this can save is really difficult to know. Based on this, a modification of SDT solution is proposed in (Yilin & Wenhai, 2010). By using feedback controlling system to automatically adjust the parameters of SDT to compress data with compression ratio increase by 60% to 75% and the absolute difference between actual error and the expected error can limit on 10^{-3} orders of magnitude, which is much lower than 10^{-2} in orginal SDT (Yilin & Wenhai, 2010). Figure 1.3 shows the controller of the SDT system.

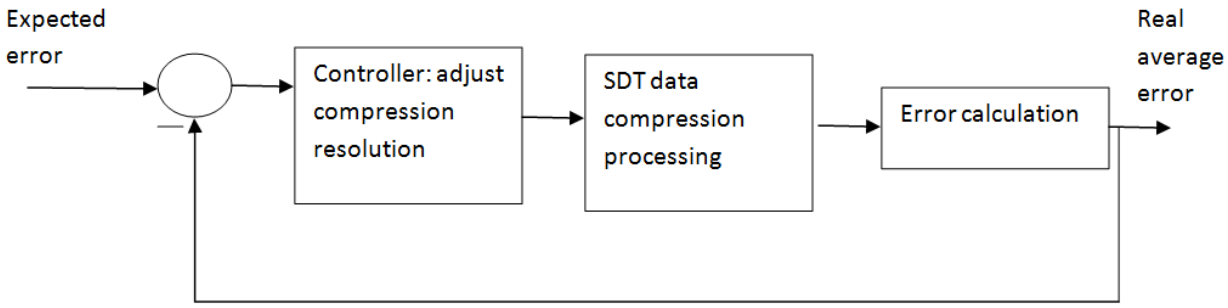


Figure 1.3: SDT system.

- Wavelet compression is also one technique that has been used in time series signal compression. In article (Oinam *et al.*, 2013) which demonstrates the comparison process and result by using Wavelet Decomposition, Wavelet Packet and Decimated Discrete Wavelet compression techniques. In this method, the original time series signal is decomposed by time-frequency transformation into groups of coefficients, each group represents signal in appropriate bandwidth. The coefficients which belongs to lower frequency have higher amplitudes and coefficients which belong to higher frequency have lower amplitudes. So if some numbers of coefficient absolute amplitudes are close to zero and neglected, then compression is obtained. So this method can be one option for Wärtsilä data compression system as well. However this method is mostly discussed and used in image and video compression because of its complexity and other factors, therefore for time series or process data compression, it needs more detail research based on the requirement of the application performance.
- Extracting the major extrama is another data compression technique, which is discussed in (Fink & Gandhi, 2011) and based on minima and maxima data. The theory for this technique is try to find the extreme values such as strict minimum, left-end right minima and flat minimum. So based on important data, compression rate, distance function and monotonicity this compression technique is applied. This method can be used for indexing and fast retrievaling of series, and for evaluation of similarity between compressed series, however this is a fast lossy compression and the compression rate is very difficult to optimised, so this method is not recommended.

- Merging the segments with different resolutions can reduce data amount and storage requirement in database depends on users requirement. However this may cause latency as segment can enter a compression buffer only after it exits the preceding one. (Goldstein *et al.*, 2011)

Moreover, with different types of data, different compression techniques can be used: for binary or integer the changing algorithm can be used which only record when value changed; for process value the modified SDT can be used.

In general, direct method is normally used in application, especially with SDT which is also used in today's Wärtsilä system. So the method which modifies the standard SDT, can improve the performance by automatically adjusting key parameters through controlling.

Furthermore, database is also one important factor in the whole process of data analysis, therefore database platforms which support time series data are also studied.

- Relation database, with the table structure, in every measurement interval, tables are populated with fresh data that increases table size, meanwhile when table indexes become large, data retrieval becomes significantly slow. However with the invention of time series database servers (TSDS) which is mostly used in industry, a better performance is reached. (Deri *et al.*, 2012)
- Round-Robin database, it relies on file system, so its performance are limited by filesystem and disk, moreover RRD database needs to update at each time step, thus all of those makes it time consuming and unsuitable.
- tsdb, which is created to handle big amount of time series data.
- MangoDB is also one popular option baseuse of its flexibility and fast speed.
- Hybrid solution: a classical time-series database for the historical data, and a relational database for analysis and reporting etc. This is used by many applications such as OSIsoft.

So after data reduction, data size is reduced siginificantly, and because the reduced data keeps the original data characters, so it can improve data mining efficiency.

The amount of data being stored in the estimated system is 7.9GB/month or 96GB/year uncompressed for one engine, let's take the average amount of 8 engines in a plant , so for 8 engines totally $96 \times 8GB = 768GB = 6Tb$ which will be slow in relational database.

So compression is definitely needed in this case. And according to (Yilin & Wenhai, 2010), the modified SDT compression ratio can increase by 60% to 75% when compares to original SDT. So the data amount will be siginificantly decreasing to $Original\_data \div RC \times 60\%$, so the bandwidth can be reduced to 60% as well if the same performance is requested.

1.1.7. Data Transformation

Data transformation is the process to transform data from one format into other formats which are more appropriate for data mining, it includes following operations:

- Filtering: remove noise from data
- Aggregation: it is any process where information is represented in a summarized format to achieve specific process for analysis.
- Scaling is used to scale the data within a specific range, standardize all the features of dataset, so that all the features are equally weighted.
- Generalization: concept hierarchy generation
- Attribute construction: new attributes are constructed and added, to improve knowledge discovery accuracy.
- Discretization can divide the continous parameter value into intervals, and all intervals have their own labels. If this is applied, raw values are replced by interval or conceptual labels.

1.1.8. Data Modeling

Data modeling is the process in which multiple sets of selected and preprocessed data are combined and analyzed to uncover relationships or patterns. By using algorithm to selected and preprocessed data to build a model, to discovery statistics, patterns, associations, correlations, and prediction. There are many methods available such as Linear Regression Models, k-Means Clustering, Markov Chain Models and Hidden Markov Models etc. This is mostly application based, so it is normally mentioned in an application level.

1.2. The Concept of Big Data

It is not a surprise that the amount of data generated on a daily basis is staggering. The world is in a data revolution phase, where the amount of data has been exploding at an exponential rate. This has led to the introduction of the concept of Big Data. For decades, companies have been making decisions based on data from relational databases, beyond that structured data, however, is a potential treasure trove of non-traditional, less structured data. Since the data storage capacity and computer processing power are developing fast, it is doable to collect those huge amounts of available data and analyze it further for useful information.

Big Data is a relative term describing a situation where the amount and cumulation speed of incoming new data, and the diversity of data exceed storage or computing capacity for using relational database (Inc, 2012b). The main characteristics of Big Data are presented in Figure: 1.4 (Grobelnik, 2012)

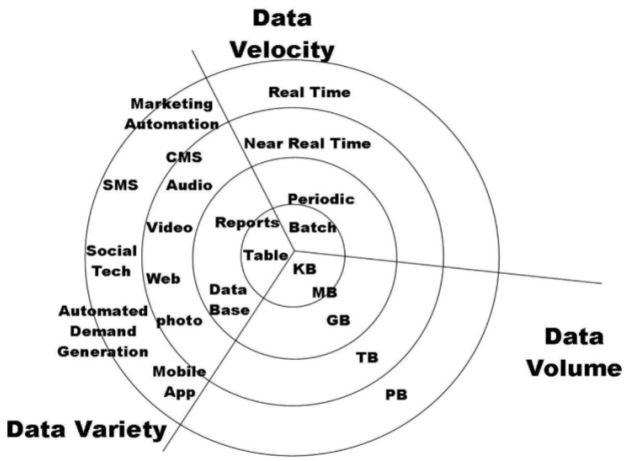


Figure 1.4: The characteristics of Big Data.

As shown in Figure 1.4, Big Data has the following characteristics: data volume is increasing exponentially, data is generated fast and need to be processed fast, data sources are in various formats, types and structures. Commonly veracity which defines the quality of the data is also considered as one important characteristics of Big Data.

A data environment can become extremely complex along any of the above characteristics

or with a combination of two or all of them. It is important to understand that with any of the characteristics above, it is difficult to handle Big Data by using traditional relational database. Therefore, completely new systems must be developed.

1.2.1. Big Data Processing Architectures

Within the last 20 years, data center infrastructure has been designed in a manner that closely aligns data, applications and end users to provide secure, high-performance access. (Inc, 2012a) The infrastructure has often been referred to as a three-tier client-server architecture in which the presentation, application and data management are physically separated.

This kind of architecture is largely optimized based on the needs of the end users and on the performance of the database systems. However, as data becomes more horizontally scaled and distributed throughout network, traffic between server and storage nodes has become siginificantly greater than traffic between servers and end users. (Inc, 2012a)

The processing of Big Data can be divided into five main activities, as presented in Figure 1.5.

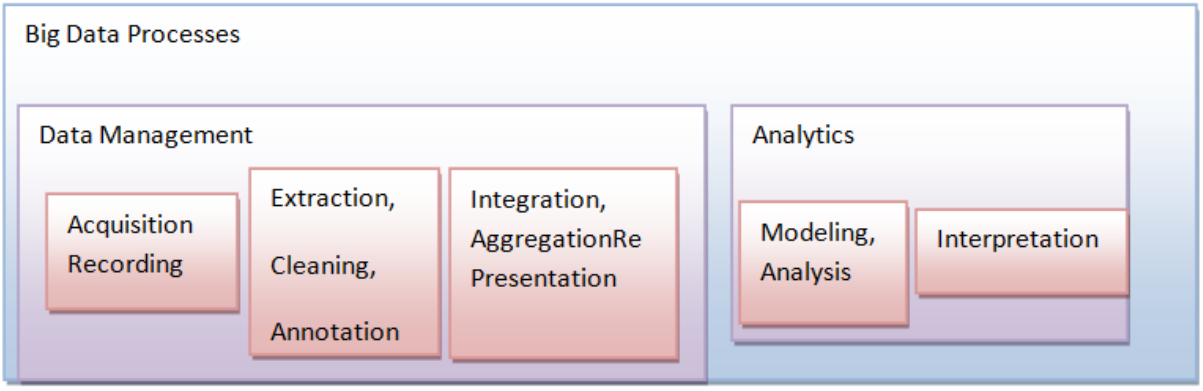


Figure 1.5: Five main activities in Big Data processing.

As presented in Figure 1.5, the Big Data management involves processes and supporting

techniques to acquire, store and prepare data for analysis. In analytics phase, techniques are used to analyze and acquire the targets of interest from the Big Data.

The Hadoop is a software framework for distributed storage and distributed processing of large sets of data on commodity hardware. It includes a distributed file system HDFS and the paradigm MapReduce which is for analysis and transformation of large sets of data. It is a critical Big Data technology that provides a scalable file storage system and allows a horizontal scale of data for quick query, access and management. The architecture of the Big Data management system is presented in Figure 1.6 (Walker, 2012)

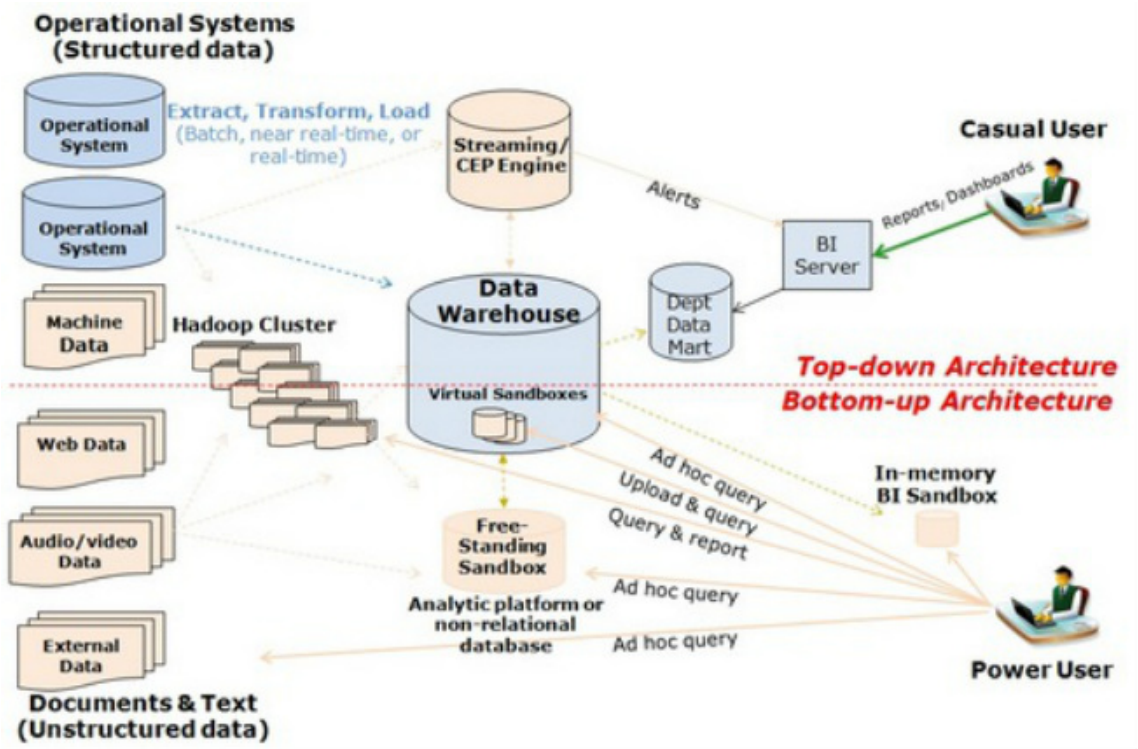


Figure 1.6: The architecture of the Big data management system.

In Figure 1.6, the objects in blue represent traditional data architecture, objects in pink represent the new data warehouse architecture. The new data warehouse architecture includes Hadoop, NoSQL database, analytical engines and interactive and visualization tools (Walker, 2012) . In the traditional business intelligent(BI) architecture, analytical process first passes through a data warehouse. In new BI architecture, both structured and unstructured data are transmitted through Hadoop which acts as a staging area and online

archive. From Hadoop, the data is fed into a data warehouse hub. It distributes data to downstream systems, where the users can perform queries by using SQL-based reporting and analysis tools. So the modern BI architecture can analyze large volumes and various types of data. It is better platform for data alignment, consistency and flexible predictive analytics.

1.2.2. Big Data In Industry

There are great expectations regarding the Industrial Internet, which is the combination of Big Data analytics and the Internet of Things.(General Electric, 2014) Internet-connected devices collect data and communicate through internet, makes it possible to collect massive amounts of data.

To be able to utilize Big Data, industry needs to have the infrastructure to support different types and massive amounts of data, and also the ability to use the collected historical data, and to perform analysis. As techniques develop, more and more bussiness is expected to come from information based techniques. According to one servey from GE and Accenture, 73 percent of companies are already investing more than 20 percent of their overall technology budget on Big Data analytics (General Electric, 2014). Industrial companies are facing mounting pressure of staying competitive with data-driven strategies, which requires increasingly more data and this in turn will accumulate larger datasets. Obviously with this volume of data from which to extract value is beyond the capability of RDBMS, moreover the data source from industry come in various formats and from scattered sources. Those create more challenges for RDBMS. The Industrial Internet enables companies to use sensors, softwares, communication and other technologies to gather and analyze data from physical components or other large data streams, then use those analyses to optimise operation, manage assets, problem disgnosis, predicting and preventing risks and other new value-added services.

1.3. Research Issues and Applied Methods

This thesis work is done for Wärtsilä Ship Power 4-stroke R&D department. As mentioned before, industrial data from sites is increasing rapidly, with the gradually improved infrastructures of sensor networks, communication systems and computer systems. Wärt-

silä, a global operator in complete lifecycle power solutions for marine and energy markets, had awared of this years ago.

In Wärtsilä a very matural data collection and communication system have been developed and applied. This thesis work is done based on the current asset management system about data mining and analysis.

Nowadays, in Wärtsilä ship power experts can get detailed field data by manually copying, and power plant data are accessable either through manually copying or the available platform WebWOIS. WebWOIS is a web-based platform where the collected data from power plants and ships can be extracted. WebWOIS has an interface for regular users where user can download the data in a CSV format, and a Python interface where Python users can access data from IPython Notebook. However, majority users in Wärtsilä R&D are using Matlab, therefore, in this work, a Matlab user interface is designed as one adding function in WebWOIS. The whole work is done in a flow of: first the platform of WebWOIS is studied, data structure is analyzed, then data is extracted and exported to Matlab. Finally this function is integrated into WebWOIS and tested and evaluated by users. This work improves the functionalities of WebWOIS, and also enables Matlab users to extract and analyze data more efficiently.

Moreover, for the future asset management system research plan, Wärtsilä is designing a new system Optimisers which aims at improving the use of information and knowledge. In this work, the user interface for Wärtsilä Optimisers is proposed for internal users . In order to fulfill this task, first a preparation of needs gathering from all different departments and experts is done during which meetings and interviews are arranged. Based on the interviews, a signal requirement for Wärtsilä Optimisers system to be monitored is standardized and user interface functions are listed. Finally a user interface is proposed for Wärtsilä Optimisers with a static offline website. This work clarifies the needs from Wärtsilä internal experts about the available and non-available signals, Optimiser system and user interface designing. Moreover, it provides a solid instruction and reqirement for later developing work.

Different methods and resources are utilized in order to reach the goal of this work:

- Interviews: interviewing and gathering the needs from experts about engine monitoring signals and functionality improvements in their current and future work.
- Assembly line study: in order to get a better understanding of engine components and working theory, a one week study in assebling line is conducted.

- Seminar: a one day seminar in Helsinki was organized by Teradata which gave practical introduction of utilizing Big Data, and sublimating with applications in various areas. This provides one option for future data warehouse system improvement.
- Software experience, since one of the tasks is user interface proposing, so different softwares with similar functions are experienced and tested.

The whole thesis follows the sequence of work explanation, summary and future work. In Chapter 2, Matlab user interface designing with WebWOIS platform is introduced and implemented. Then followed by user interface designing for Optimisers in Chapter 3, where different communication standards are listed, needs are specified from users and proposed interface is explained in detail. Finally, the whole thesis ends with conclusion and future work, where a summary about current Wärtsilä Optimisers system is given, and possible techniques and impreovements for future work are summarized.

Chapter 2

EXTRACTING THE MEASUREMENT DATA

Data extraction is the process of retrieving data out of data sources for further processing. It aims at electing the correct information from huge amount of data for further process of analysis. This process connects data source with data analysis service application. In this chapter, Wärtsilä WebWOIS Matlab interface designing is introduced and implemented, during which the whole process of data extraction is clearly explained.

2.1. The Introduction of Wärtsilä WebWOIS

As a global operator in complete lifecycle power solutions for marine and energy markets, Wärtsilä has equipped its products with many types of sensors in order to remote monitoring, support operators in maintaning and optimising equipment performance. Moreover, internal experts can utilize this information to improve product development and reliability. Therefore, a data accessing platform is needed in order to get the available data.

Wärtsilä WebWOIS is a web based platform, where all the collected data are accessable in different interfaces. It is a standard RESTful API, which has separate clients and servers, JSON as server sending data format with http uniform interface. The backbone of WebWOIS system is:

- Collected data is stored and retrieved in WonderWare LGH (InTouch Historical

Log files) files and Microsoft SQL database. Here LGH files are historized tag data captured by the InTouch data logger.

- LGHParser converts from LGH format to Hierarchical Data Format (HDF) meanwhile attach Microsoft SQL database.
- WebWOIS backend designing: using Python and Pyramid framework.
- Based on Pyramid framework, different Representational State Transfer Application Programming Interfaces (RESTful API) are designed: API for IPython notebook users, API for Matlab users and WebWOIS HTML/JavaScript backend for regular users.

The visulization of Wärtsilä WebWOIS is designed based on d3.ju, which is a JavaScript library for handling documents based data. So the data gathered in site are transmitted to WebWOIS data storage, users can fetch useful signal information in this procedure: user selects site, tagname, signal, then goes to the main data extraction page, as in Figure2.1

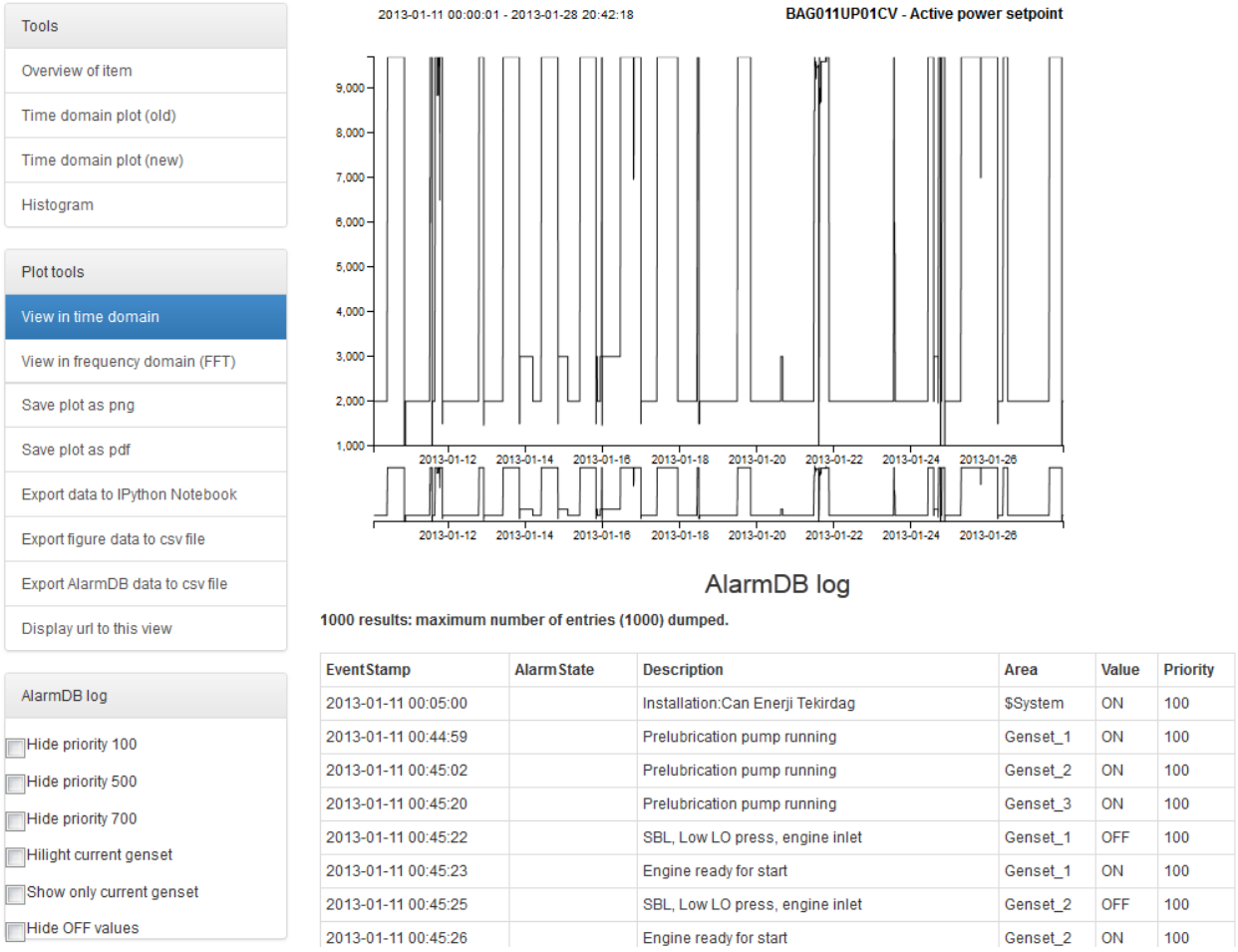


Figure 2.1: WebWOIS data extraction page.

Figure 2.1 shows all the operations users can achieve in this page. First in Tools kit, overview of the signal, view in time domain and time domain trending function for self defined parameters, and histogram are available. Then for Plot tools kit, view in time and frequency domain, save current figure to pdf and png, viewing alarm list with different priority, and export data to different format and platforms are accessable.

The main focus of this work is on the Plot tools kit with its exporting data to different platforms functions. Here in Figure 2.2 shows the already available interface for IPython and regular API:

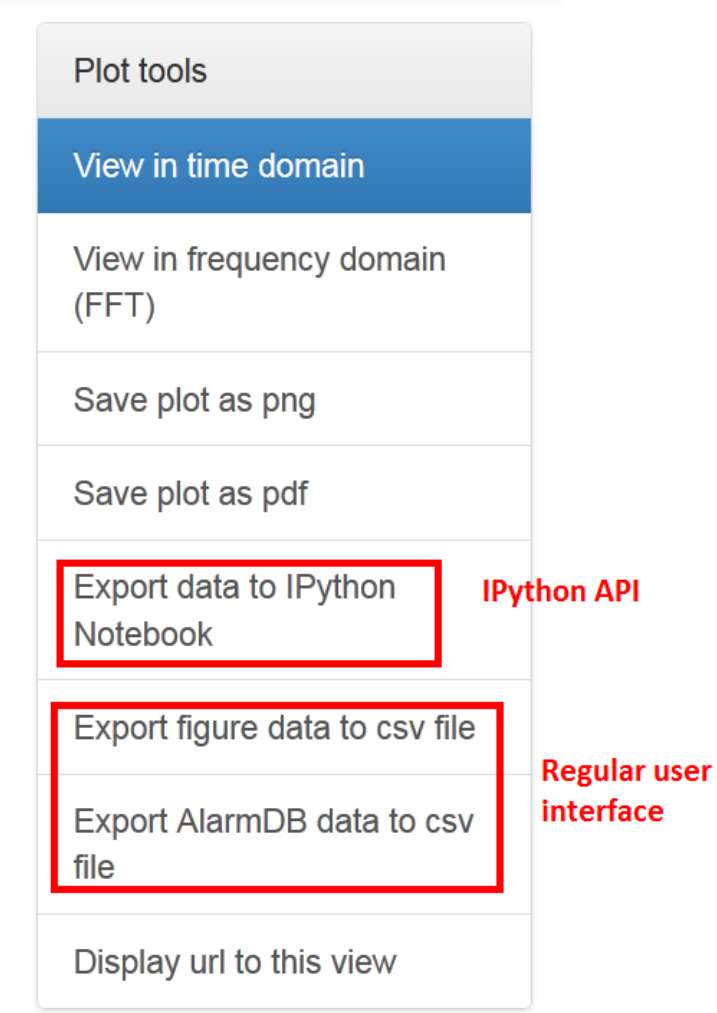


Figure 2.2: WebWOIS IPython and regular API.

As Figure 2.2 reveals that users can export signal and alarm data in two ways :

- Regular user UI: directly export and download data to CSV files
- IPython Notebook API: Running python script in IPython Notebook

Those two methods are well defined its target user groups, however because part of Wärtsilä internal experts are used to use Matlab as their daily tools, therefore it is recommended to develop Matlab API as the third method.

2.2. Data Extracting by using WebWOIS

In order to get an approach to solution, it is necessary to know the backbone structure of the current WebWOIS system. So a insight of the interactions between user, database and interface is deployed in Figure 2.3.

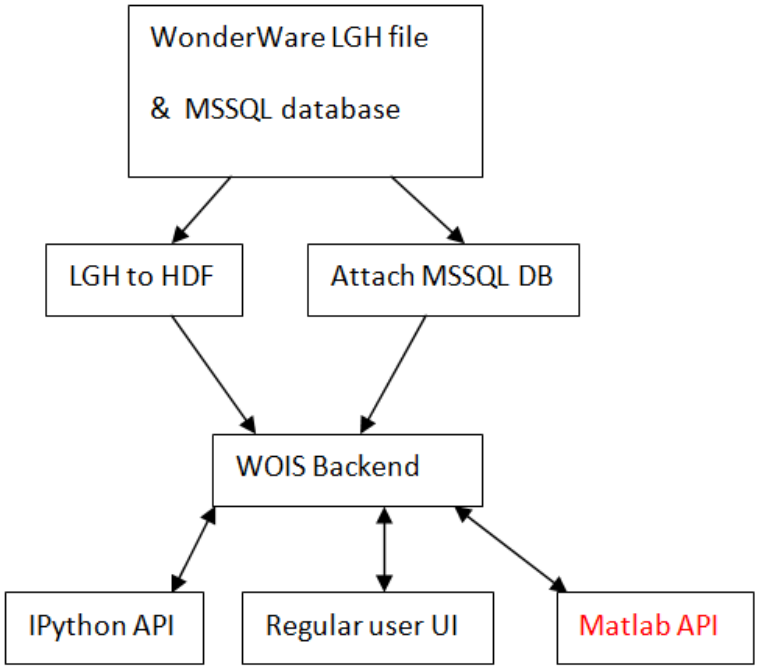


Figure 2.3: WebWOIS system backbones.

Figure 2.3 represents a clear flow of the interactions between Matlab API, WOIS back-end,and HDF or MMSQL database. In other words, the flow of interaction is : First, user initiates a user session by using web browser. Then, browser sends request to Server and at last Server returns response. So Matlab needs to fulfill the function which first forwards the request to server, then reads the result file from server and extracting data from this result files and exporting data to Matlab.

Matlab API has interaction of WebWOIS backend, the detail view of current data system is done: In WebWOIS, signal data and alarm data are stored separately in HDF and JSON

source files.

- HDF is a set of file formats designed to store and organize large amounts of numerical data. The current version HDF5 is used in WebWOIS, which simplifies extremely large and complex data collections into Datasets and Groups.

  In Matlab 2014b function h5info returns the structure information about HDF5 file, and h5read is documented for reading data from HDF5 data set.

- JSON is highly portable, human readable text format data objects which uses attribute and value pairs to represent complex and hierarchical data.

  In Matlab no available function can work with JSON files, so here JSONlab is used as a JSON encode/decode library. JSONlab is a open source implementation of JSON encoder and decoder in Matlab language. It can convert JSON file into Matlab data structure and vice versa.

In conclusion, WebWOIS signal data is stored in HDF5 file format. And every signal has its own HDF5 file with the whole life cycle data stored. So when the user has a requirement of a specific time interval data, the only way is to download the HDF5 file which contains the whole life cycle time data, then going through the whole file and fetching the required data.

Alarm list in WebWOIS is in JSON format which can be dynamically generated by server based on the time slot and other related requests from the user. So alarm data can be retrieved directly by downloading correct time interval JSON file from the server. Then encode the downloaded JSON files to Matlab workspace.

All the files are downloaded based on dynamically constructed uniform resource locator (URL): first web browser interprets users' requests to query, and send it to web server, then based on query information dynamic URLs are constructed and server returns the result in file format.

To make it more straightforward, in Matlab the whole process can be concluded as: users' query information can be inputs to Matlab function, then depends on the input information, URLs are constructed dynamically, and the data files are downloaded based on those URLs. After files are downloaded in Matlab workpath, by using different methods to encode HDF5 and JSON files and extracts the data into Matlab workspace. Based on this route, implementation is done and the following sections explain the details of the implemantion work.

2.3. Implementation of the Developed System

In this work, implementation is done by using Matlab 2014b. The way how the data is selected goes according to the following stpes: first the user selects the site and tag name, Then followed by time define of the selected signal. In the meantime, users can determine if there is a need to use filters to get the filtered alarm data only. Finally the data can be retrieved based on user inputs. In another words, the data that are retrieved is for specific signal in specific time slot with optional filter parameters. In addition, signal data and alarm data are in different files, therefore it is needed to fetch all the files one by one. The whole process can be represented as a flow chart shown in Figure 2.4 below.

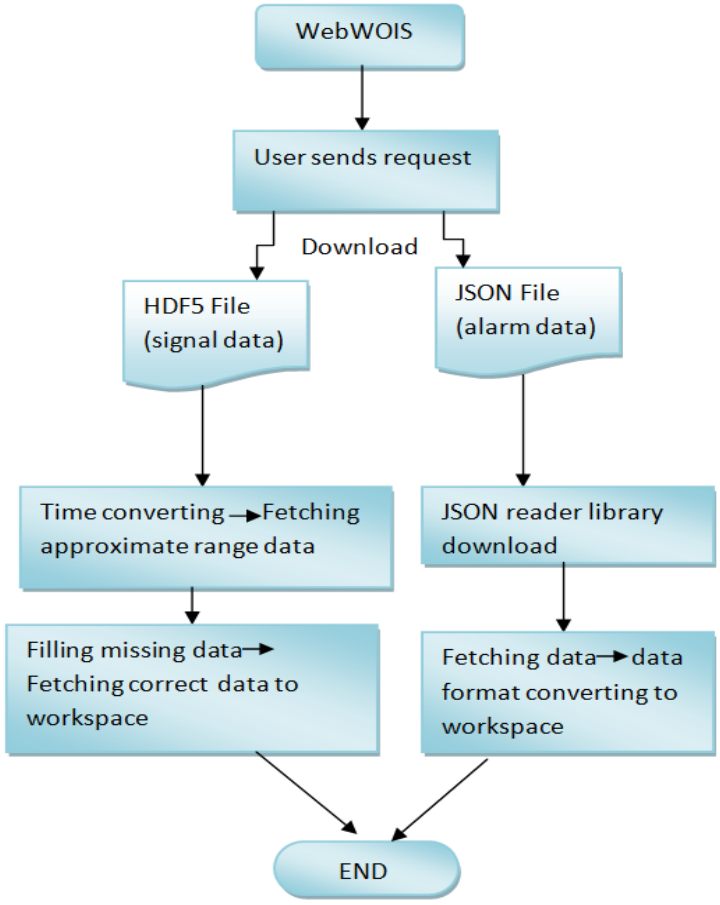


Figure 2.4: Matlab API general working process.

For data signal file, downloading based on URL is achieved by using function websave

which is available since Matlab 2014b. It is a function to save content from the web service specified by URL. For this the assembling method of the dynamic URLs are needed, here shows how the signal URL is dynamically constracted:

```
1 http://fis8038.accdom.for.int/wois-0.6/sites/ + siteName +download_h5_file?tag= +↩
      tagName +.h5
```

The code explains that, for signal data, only siteName and tagName are needed as dynamic inputs from user. With a pre-defined format, only needs to fill in the two missing parameters, then the dyanmic URL is constructed. Moreover, based on this URL, the file format can be extracted, so that the file after downloading still keeps its format. And by using siteName and tagName, the file name is constructed in the format of $siteName - tagName.h5$. Thus, the data file is downloaded in Matlab workpath with a specific file name $siteName - tagName$ and file format:$.h5$

Another issue with websave function is the authentication. Because all the data is preserved in Wärtsilä internal network with http basic request, therefore authentication process is needed before the download work. In this work, it is implemented by utilizing an already build dialog as seen in Figure 2.5.

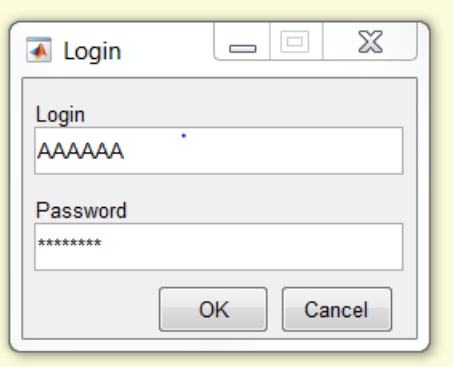


Figure 2.5: Login GUI.

This login dialog is from Mathworks File Exchange, which contains all the basic functions of login: user name and password input, meanwhile the password is visually hidden for security. With this function, user name and password are stored and passed to websave for authentication purpose.

So with correct URL and user authenticated infromation as inputs to websave, signal data is downloaded to Matlab workpath. Following steps are to extract correct time interval data and transform it into Matlab format.

The time interval is obtained by user manually zoom in or zoom out from the trend plot. When the user has selected the right time interval, the starting time and ending time of this interval are send to server. So Matlab can use those information as inputs to narrow down and fetch the target data from the whole life cycle data.

The time range parameters from user are in ISO 8601 format. However, in HDF5 data file, time is based on Unix time system. Thus a conversion from ISO8601 to Unix is premise. In this work this is done by the formular of converting time from ISO8601 format to Unix format.

With all the parameters known, approaching of HDF5 file starts with its structure analysis by using h5info. From here a clear structure of signal file is given: signal data is stored in one struct with 2 fields; one field is index, one field is value. And the data value of every time index is stored in the same row of the value field. However, by default of the data collecting system behind WebWOIS, Swing Door(SDT) compression method is applied.

Here in WebWOIS the deadband in SDT is 1%. Therefore the time interval between two recorded values are not constant, so in Unix time stamp, the index value is not continuous with fixed difference, as shown below in Figure 2.6:

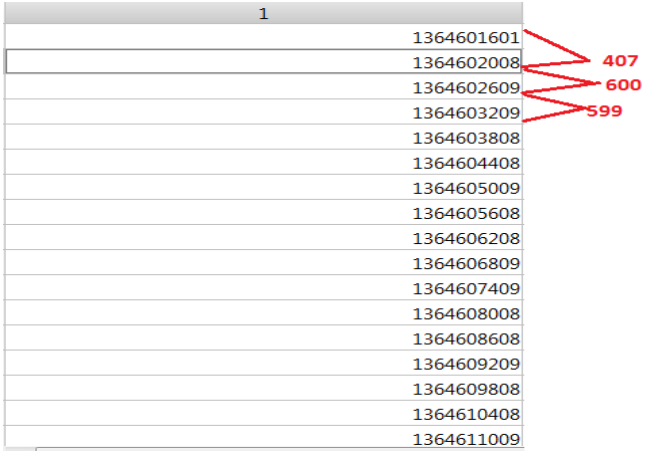


Figure 2.6: Signal data index value in unix time stamp.

Figure 2.6 points out that the index column has a random difference between any adjacent values. And due to the compression method is used here, it is demanded to reconstruct the missing value according to the available data. According to the Swing Door compression and precision of WebWOIS, it is requested to reconstruct the missing value by keeping the missing value the same as the previous archived value in trend. So the reconstructing process needs to do a computation among the available data. However, struct is not a computable format, so the struct format data is converted to mat format which is an ordinary array of the underlying data type in Matlab.

In order to fill the missing values, two methods are proposed:

- Filling all the missing points for the whole variable which contains the whole life cycle data of the signal. Then based on the signal time interval parameters to extract the requested data. This method in theory can work perfectly, however with the huge amount of data it is time consuming to fill in all the missing points, especially when the user only intends to download a small amount of data.
- Extracting the requested data from the unfilled data directly based on the time interval parameters. In this method it is possible that the time parameters after converting to Unix time are missing from the available data, because of this nonuniform value difference. For this reason, an estimated range data is extracted directly by defining starting index with smaller or equal to signal starting time and end index with bigger or equal to signal ending time. In this way the data is narrow down siginificantly before missing points are inserted. Then filling all the missing data only for this part, and going through the filled data again and find the exact range of data. In general this method has higher efficiency especially when data size is small. Therefore in this work this method is applied. Figure 2.7 explains this theory with an example.

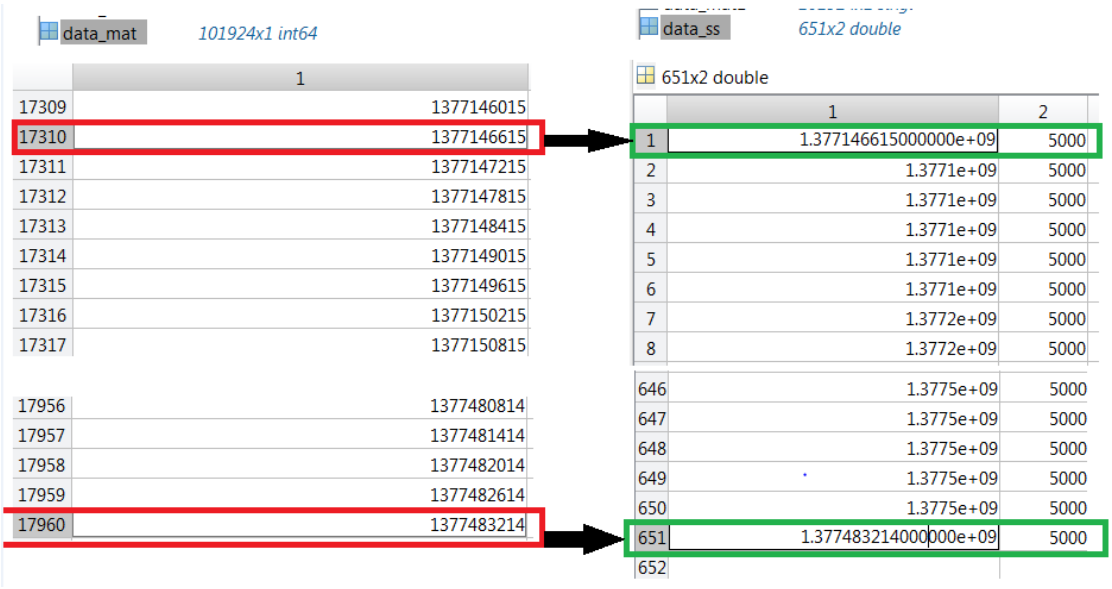


Figure 2.7: Fetching approximate time range data.

In Figure 2.7, datamat is the original data with whole life time data stored. The strarting time in Unix time stamp is 1377146615 and it is in line 17310; while the ending time in Unix time stamp is 1377483214 and it is in row 17960. So totally only 651 rows of data are fetched out from 101924 rows of data for further process.

In order to keep the unarchived data value the same as its previous archived value, in Matlab it is done in this way:

- First, filling all the missing index value with difference 1.
- Then, filling the second column with all 0s.
- Put the location of the index which is archived.
- Finally, filling up all the data with for loop.

The data after filling in all the missing points is with size: 336600 rows and 2 columns. Figure 2.8 shows the result in long format.

1.377146615000000	0.000005000000000
1.377146616000000	0.000005000000000
1.377146617000000	0.000005000000000
1.377146618000000	0.000005000000000
1.377146619000000	0.000005000000000
1.377146620000000	0.000005000000000
1.377146621000000	0.000005000000000
1.377146622000000	0.000005000000000
1.377146623000000	0.000005000000000
1.377146624000000	0.000005000000000
1.377146625000000	0.000005000000000
1.377146626000000	0.000005000000000
1.377146627000000	0.000005000000000
1.377146628000000	0.000005000000000
1.377146629000000	0.000005000000000
1.377146630000000	0.000005000000000

Figure 2.8: Filling the missing points.

At last, by going through the filled data, to extract the part with the correct starting and ending time. And the data is extracted to workspace successfully and stored in variable 'signal_data'.

Implementation: Extracting Alarm Data

Alarm data extraction, the general solution is similar with the signal data: both start with downloading the data file. But because alarm data is dynamically generated and stored with the requested time interval, so by constructing the URL dynamically, the requested alarm data is listed in one file. Therefore, reading through this whole file, data can be directly extracted to Matlab.

For alarm data URL is constructed in a way that it depends not only the siteName, signal strating and ending time, but also other optional parameters which are used to set the alarm data priorities. So a dynamic URL can be structured based on all the compulsory and optional parameters, as shown below.

```
http://fis8038.accdom.for.int/wois-0.6/sites/ + siteName + /alarmdb.json?sunix= + ↩
    signal_startTime + &eunix=+ signal_endTime + Optional Parameters
```

The optional parameters are constructed in the following format.

```
&genset=Genset_N + &hide_priority_100=1 + &hide_priority_500=1 &hide_priority_700=1 &↩
    hide_off_values=1
```

From this dynamic URL, a JSON file is created on the fly with the correct data stored. In another word, all the alarm data that user has requested is stored in one JSON file.

In Matlab, there has no functions available that man can use to read JSON file, however there is open source functions shared in Mathworks which aims at encoding and decoding JSON files. So here in this work, JSONlab is utilized as a library and by using the functions from this library JSON data can be fetched. Alram data is stored in cell format parameter, which contains 6 fields: EventStamp, AlarmState, Area, Value, Description and Priority. In order to visualize the cell data in a more user friendly way, cell is converted into table format as Figure 2.9 shows.

1 EventStamp	2 AlarmState	3 Area	4 Value	5 Description	6 Priority
1.3771e+12	''	'Genset_1'	'ON'	'Prelubrication pump ...	100
1.3771e+12	''	'Genset_2'	'ON'	'Prelubrication pump ...	100
1.3771e+12	''	'Genset_3'	'ON'	'Prelubrication pump ...	100
1.3771e+12	''	'Genset_7'	'ON'	'Prelubrication pump ...	100
1.3771e+12	''	'Genset_4'	'ON'	'Prelubrication pump ...	100
1.3771e+12	''	'Genset_1'	'OFF'	'SBL, Low LO press, e...	100
1.3771e+12	''	'Genset_6'	'ON'	'Prelubrication pump ...	100
1.3771e+12	''	'Genset_5'	'ON'	'Prelubrication pump ...	100
1.3771e+12	''	'Genset_2'	'OFF'	'SBL, Low LO press, e...	100
1.3771e+12	''	'Genset_3'	'OFF'	'SBL, Low LO press, e...	100
1.3771e+12	''	'Genset_4'	'OFF'	'SBL, Low LO press, e...	100
1.3771e+12	''	'Genset_7'	'OFF'	'SBL, Low LO press, e...	100
1.3771e+12	''	'Genset_5'	'OFF'	'SBL, Low LO press, e...	100
1.3771e+12	''	'Genset_6'	'OFF'	'SBL, Low LO press, e...	100
1.3771e+12	''	'Genset_1'	'ON'	'Prelubrication perfor...	100
1.3771e+12	''	'Genset_2'	'ON'	'Prelubrication perfor...	100

Figure 2.9: Alarm data in table format.

After all the processing work, signal and alarm data are successfully extracted into Matlab workspace, therefore all the downloaded data source files are deleted from local path. Becuase the size of data files can differ from KB to GB, so if they are not deleted after usage, in long run they may blow up the compter memory with all h5 and JSON files. Therefore a simple but critical step is needed before stepping out of the function.

At last but not the least is to encapsulate or package the whole process:

- Library files: all the functions are encapsulated as library files, so users need to download the library files into own workpath. Here the same principle is used as data source file downloading: all the library files are preserved on fly behined URL. However, those library files can be downloaded without any authentication, because of its complexity and unnecessary of confidential.
- Input parameters handling: receiving inputs from server and forwarding inputs to Matlab.
- Utilizing library files with the parameters to extract required data.

This encapsulated code is added to WebWOIS as the Matlab API. So users can simply copy this code and run it in Matlab, the required data are stored in workspace after the whole process.

Chapter 3

USER INTERFACE DESIGN FOR THE ASSET MANAGEMENT SYSTEM

Asset management system is any system that monitors and maintains property value to an entity or group. In Wärtsilä it refers to the system of monitoring and maintaining the facility systems and to the practice of managing assets to achieve the greatest return, with the objective of providing the best possible service to users. (Wikipedia, 2015b) Wärtsilä asset management system Optimisers is one platform which provides data acquisition, analysis and reporting etc, in order to enable asset monitoring, maintenenace optimizing and operation optimizing. It uses data mining as the foundemental techniques where the collected data are preprocessed and extracted to perform asset condition monitoring. Moreover, based on the mined data, one can not only tell the history operation condition, but also predict the future situation. Therefore, it is possible to improve current asset usage efficieny, mitigate possible risks and plan maintanance in advance.

In this chapter we will look through the tasks about asset management system Wärtsilä Optimisers, not only the user interface designing but also mapping the requirements from experts about the whole system.

3.1. General Architecture of Asset Management System

The asset management system is an application of Internet of Things (IoT), so here a research in IoT level is applied. It consists of the processes of data collection, data trans-

mission, data storing, processing and analysis etc.

Figure 3.1 (Rogers, 2014) shows the genral architecture of the asset management system.

Figure 3.1: General architecture of asset management system.

In physical level, various devices are used to fulfill the function of collecting data. In the case of Wärtsilä, sensors are the most commonly used device. Then by using different commnucation standards to transmit site data to remote data warehouse. Finally, the data warehouse users can access the data through visualization or presentation layer. In Figure 3.2, the data warehouse is based on Big Data, however, it works in the same way with traditional database.

In this work, we focus on the connectivity of the whole system. Figure 3.2 (Guruprasad.K.Basavaraju, 2014) shows the connectivity with different distance ranges.

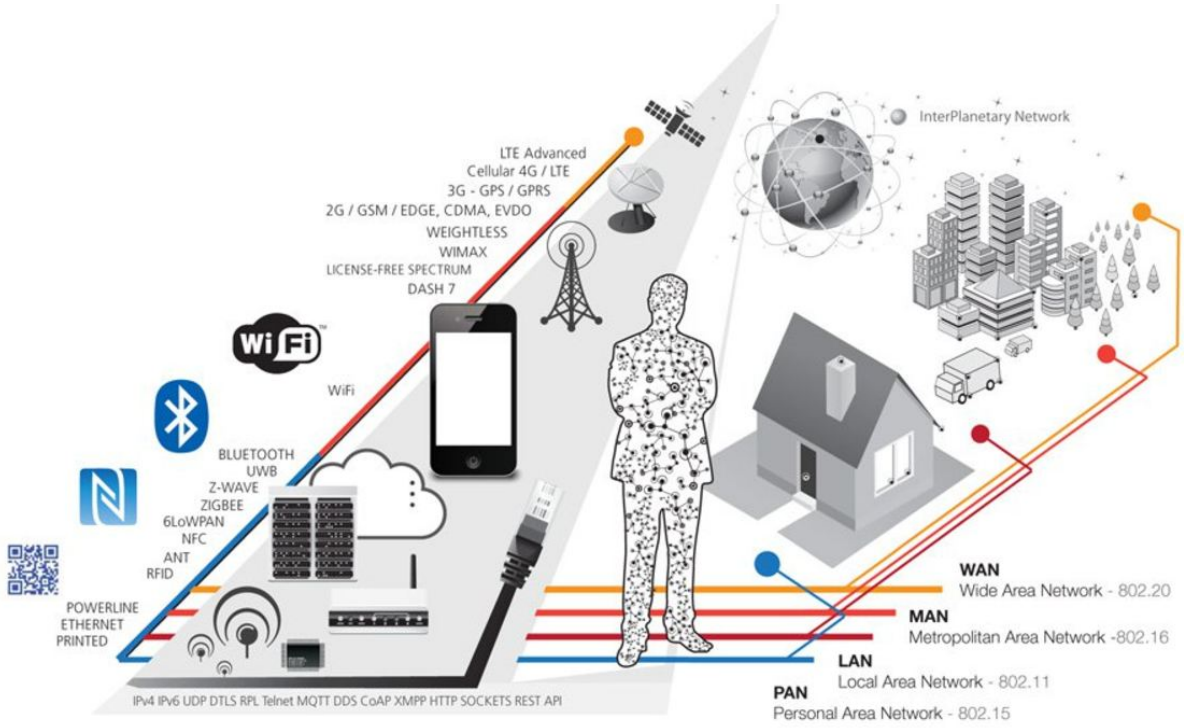


Figure 3.2: IoT connectivity protocols.

Sensor systems can be used for to collect and transmit information about their surrounding environment. Many technologies which can be applied in WSNs have been developed in recent years, such as, Bluetooth Low energy, IEEE802.15.6, IEEE802.15.4 (Zigbee), WirelessHART, ISA100, WIA-PA and 6LoWPAN etc. And some other standards which are not open standard but have been widely applied in certain field, such as the Z-Wave etc. Despite of the diversity of technologies, some common features are shared: low power consumption, short range communication, flexible networking capacity and light weight protocol stack (Pang, 2013).

Industrial networks can be divided into three categories based on functionality: field level networks, control level networks and information level network. Field level and control level are both for site based processes. Here the commonly used site based network protocols include:

- PROFIBUS can provide digital communication for process data and auxiliary data with speeds up to 12Mbps.

- Control Area Network (CAN) bus provides physical and data link layer for serial communication with speeds up to 1Mbps.
- CANopen and DeviceNet are higher level protocols on top of CAN bus to allow interoperability with devices on the same industrial network.
- Modbus can connect up to 247 nodes, with speeds up to 115kbps.
- CC-Link is based on RS-485 and can connect with up to 64 nodes with speeds up to 10Mbps.
- Ethernet: industrial Ethernet protocols uses modified Media Access Control(MAC) layer to achieve very low latency and deterministic responses.(Wikipedia, 2015f)In this protocol, the nodes number in the system can be flexible. Ethernet is becoming the trend in industry, therefore more and more industrial communication protocols are moving to Ethernet-based solutions.

In industrial applications which require critical real-time and reliability, wired Ethernet and/or field buses are often used. And due to the substantially higher performance and cost effectiveness, an upgrading from buses-based solution to Ethernet-based solution is getting more widely applied. And the commonly used Ethernet-based protocols are (Lin & Pearson, 2013):

- EtherCAT is a real-time Ethernet Master-Slave network. It is a MAC layer protocol, and it is transparent to any higher level Ethernet protocols. It can connect up to 65535 nodes, and the master can be a Ethernet controller.
- EtherNet/IP is an application layer protocol on top of TCP/IP. It combines standard Ethernet technologies with the Common Industry Protocol(CIP) (Wikipedia, 2015d). It can have unlimited nodes in a system, but it has limited real time and deterministic capabilities.
- PROFINET has three protocol levels, first level, access to PROFIBUS network through TCP/IP with cycle time 100ms. The typical application is building automation. Second level, PROFINET Real-Time with cycle time 10ms, it is normally used in factory automation and process automationa and other PLC-type application. Third level, PROFINET Isochronous Real-Time with 1ms cycle time, it is used for motion control operation application (Wikipedia, 2015f).
- Powerlink is a deterministic real-time protocol. It expands Ethernet with a mixed

polling and timeslicing mechanism for real-time data transmission. Modern implementations can reach cycle time of under 200us and jetter of less than 1us. (Wikipedia, 2015e) This kind of system can be used for all kinds of automation and motion application.

- Sercos III merges the hard real-time aspects of the Sercos interface with Ethernet. It can have 511 slave nodes and is mostly used in servo drive controls.
- CC-Link IE enables devices from numerous manufacturers to communicate. It has two versions: CC-Link IE control is designed for controller-to-controller communications and can have 120 nodes. CC-Link IE field is mainly for I/O communications and motion control, and it can have 254 nodes (Wikipedia, 2014).
- Modbus TCP is implemented on the standard Ethernet network, however it does not guarantee real-time and deterministic communications.

For information level communication or long distance communication, a connection from the site to external networks is established. For example, when a ship arrives to the harbour, it is possible to find wired connection to the Internet, such as optical network which can have extremely high data rate. However, when sailing in the sea, it is not possible to obtain wired connection. Thus, wireless technology must be applied there.

The type of the gateway which collects information from all sensors and communicates with external internet, can be divided into two groups: wired WAN, such as IEEE802.3 Ethernet and broadband power line communication, and wireless WAN such as IEEE802.11 WLAN, 3GPP wireless cellular communication(GSM, GPRS, EDGE, UMTS, LTE, LTE-A etc) and satellite. In this wide area wireless communication, signal travels from several kilometers to several thousand kilometers. (Pang, 2013) When the system is in rural environments, usually a powerful basestation is used as a gateway to access internet through wireless cellular or Ethernet.

Wi-Fi enables devices to exchange data wirelessly at a high data rate between 54Mbps to 600 Mbps. However, the transmission range of Wi-Fi is limited. Therefore, it is a feasible option when the ship is near the harbor.

2G was developed in the 1990s and used a compeletely digital system. GSM enables subscribers to use the services anywhere there the mobile station has multi-band capabilities and is able to switch between major GSM frequency bands. (Smith, 2008) This technique was improved with the launch of General Packet Radio Services (GPRS) which

uses packet switched mobile data service. Compared to dedicated, and circuit-switched channel for MS, GPR resources are only used during actual transmission.

3G is developed around year 2000. It is designed for higher data rates and it enables services for integrated high quality audio, video and data.

4G includes HSPA+, LTE and WiMAX. 4G is designed for dynamic information access and wearable devices. WiMAX and LTE are both dedicated data networks offering coverage over large areas. The main advantages LTE has over WiMAX are the greater throughput than WiMAX and compatibility with previous technologies. The latest LTE standard, LTE Advanced is regarded the only true 4G technology. So many people believe LTE is the future.

5G covers the services of dynamic information access, wearable devices with AI capabilities. The standard for 5G is not defined, and most likely it will come during 2020 to 2030 (Wikipedia, 2015a).

For onboard communications, GSM, 3G, HSPA, 4G and LTE marine network can provide the possible solution. When those are unavailable, satellite is still a great option, such as in the middle of the sea etc. Satellite does have a few more limitations than towered services. Satellites are thousands of kilometers away, and as a result, there is a ping time or lag of on average 800 milliseconds. (Inc, 2015) To an average user, this won't make much difference, but for those who want to do real-time following or trading, this could be an issue. In current marine communication industry, Inmarsat which uses VSAT type device to connect to geosynchronous satellites can provide very good communication links to ships at sea. However, satellite communication can also be very expensive.

Each generation of technology uses different communication protocols. Those include details on which specific frequencies are being used, how many channels are involved and how information is converted between digital and analog etc. Different protocols mean that with each generation, all the hardware needs to be upgraded. Different protocols are also not always compatible, so this is also one important factor when choosing the right protocol.

3.2. Specification of Requirements

The first phase of user interface designing is to gather the needs from end users and organization, so that designer can totally understand the customers' needs and provide the best solution.

For Wärtsilä Optimisers, client software W.O. Site Core and Site GUI are installed on one or more PCs at site for customers to monitor and report the asset condition, shore software W.O. Center Core and W.O.Design Studio are used by Wärtsilä experts to access data and to do the site configuration. Figure3.3 (Teräväinen, 2013) shows the functional and structural overview.

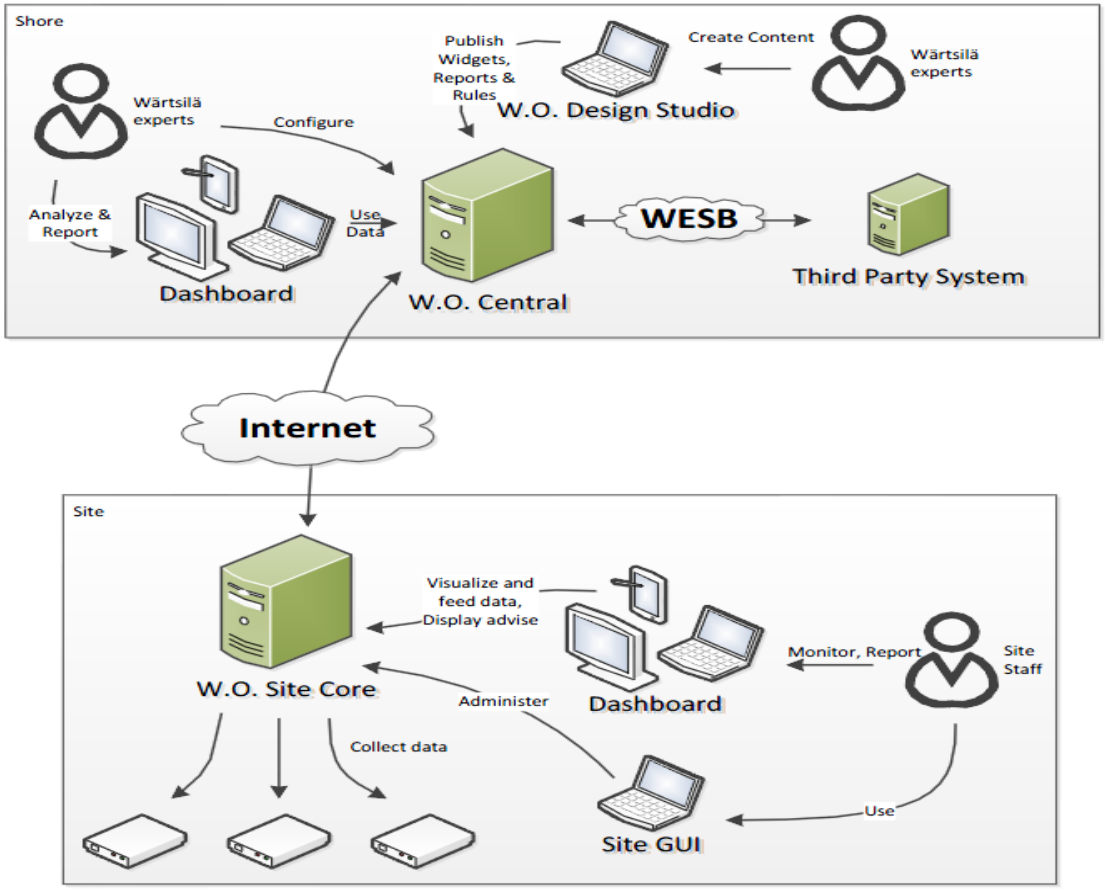


Figure 3.3: Functional overview of the system and main elements.

This work is based on a Third Part System which access data through WESB to W.O.Center

Core. The end users are the internal R&D personnel, therefore understanding their needs about Wärtsilä Optimisers and finding out the optimized interface solutions are critical foundations.

The mapping of the requirements started by interviewing experts from different fields. Totally 17 experts from 14 departments are interviewed and much more experts were contacted for revising the needs. Moreover, with the current system, Wärtsilä can monitor approaximately 3612 parameters in total. However not all the signals are required when considering the memory space, tansmission capacity etc. So a standardized parameters list is beneficial considering R&D experts needs. In this work, it is done through interviewing about the Optimisers requirements at the same time. It starts with the minimum requested signals in the list, improvements can be done later when needed. Thus the interview at the same time will provide a standardized signal list for Wärtsilä logging system.

In general, the interview covers the requirements for components, system, available and unavailable signals, functions, user interface designing and other related aspects. Meetings are organized individually, reports are documented for every meeting, finally when all the interviews are done, a summarized conclusion is gathered based on all the reports.

The requirements about Optimisers user interface is summarized in a general level:

- Flexible and intuitive filters, so that user can approach data from different ways.
- Ability to select multiple signals or options at the same time.
- Ability to save work and continue afterwards
- Not only processing data is needed, service and configuration data are also important.
- Event list, alarm list, shut down liste and other operational list.
- Sharing information can reduce duplicated work and inspire innovation effectively.
- Unavailable and useful signals should be added to Wärtsilä Optimsiers system
- Main engine signals should be saved the whole life time
- Standardized monitoring parameter list is attached in Appendix.

Moreover, the requirements about logging system parameters is summaried into excel document. There are totally 114 parameters in the list:

- 100 parameters are counters or parameters with fixed sampling frequency. Data amount is calculated based on those 100 parameters. Here the sampling frequencies are defined as 1Hz and 20 Hz. However for some parameters such as cylinder pressure which requires much higher sampling frequency, 2000Hz is defined as the nominal value.
- 2 parameters are duplicated, so they are not counted again.
- 2 parameters are out of the scope of engine, therefore they are not considered.
- 10 parameters are either no need for sampling frequency or counter such as engine software version and alram, or they are unknow currently such as EC_MRPM (filtered speed signal).

Based on the standarized parameter list for Optimisers, the data amount is summaried in Table 3.1, according to sampling frequency and system type .

Table 3.1: One engine data amount calculation.

	Slow Trend	Fast trend	Counter
Parameters	59	20	21
Channels	208	40	4122
Log frequency	1	20	—
Log points/sec	$208 \times 1 = 208$	$20 \times 40 = 800$	4122

Since the system used in Wärtsilä is 32-bit, so every symbol has 5 bits. Therefore the amount of data can be calculated by:

$$(4122 + 208 + 800) \times 5 \times 60 \times 60 \times 24 = 265MB/day \tag{1}$$

and in a month 30 days, this will be

$$265MB \times 30 = 7.8GB/month \tag{2}$$

3.3. Communication standards for Wärtsilä Optimisers System

As mentioned before, the data amount one engine generated one day is 265MB. According to Shannon theorem,

$$C = B \times log_2(1 + \frac{S}{N}) \tag{3}$$

Here C is the channel capacity in bits per second, B is the bandwidth of the channel in herts, S is the signal power and N is the noise power.

So when we know that one engine generates 265MB/day, we can define the channel capacity based on transmission time. For example if we want to send all this 265MB data in 10 minutes, then the channel capacity will be 3.6Mbps. But in practice, the channel capacity should be given, and the SNR can be measured directly by doing communication experiment. So based on this formular the requried bandwidth can be calculated.

$$B = C \div log_2(1 + \frac{S}{N}) \tag{4}$$

Moreover, by using modulation technique which means more than one bit per baud, higher data rate and better spectral efficiencies can be reached. But when modulation level increases, the bit error rate also increases. So a correct modulation technique is very important for the system performance.

During data transmission, the elements related are: data transmission rate and data amount. Here data amount is fixed by the sensor amount and frequency, but the transmission rate can be different for uplink and downlink. So here we consider this also for transmission time calculation.

By using different communication standards, the data transmission time are estimated based on both uplink and downlink data transmission rates. Table 3.2 shows the result.

Table 3.2: Data transmission time calculation for different communication standards.

Standards	Peak Download Rate	Minimum download time	Peak Upload Rate	Minimum upload time

2.5G	144kbps	4 hours	20kbps	29 hours
2.75G	384kbps	1.5 hours	60kbps	9.6 hours
3G	100Mbps	21s	5Mbps	7minutes
4G	1Gbps	2.1s	500Mbps	4.1s
WiFi	600 Mbps	3.6s	600 Mbps	3.6s
WiMAX	141Mbps	15s	138Mbps	15s
WiMAX-2	1Gbps	2.1s	–	–
LTE	300Mbps	7s	75.4Mbps	27s
LTE-A	1Gbps	2.1s	–	–
WEIGHTLESS	16Mbps	133s	80s-212min	80s-212min
HSPA+	672 Mbps	3.1s	168Mbps	12.1s
EDGE	1.6Mbps	21.1minutes	0.5Mbps	1.2hours
Satellite	1Gbps	2.1s	10 Mbps	3.4minutes

Table 3.2 shows that data transmission time varies depending on communication standards. Moreover, if we assume that one site has average N engines, so the total time of data transfer for one site can be calculated by multiplying N. However, the data transfer rates listed here are theoretical values, so the data transmission time in practice may vary significantly based on the system design.

Nowadays, data transmission via email, ftp or SMS is the easiest solution to implement in practice, because not much needs to be done at the receiver end. The data that is transmitted by the GSM-2 (e-mail/ftp/SMS) is continuously read in by the DataManager software and stored in the MySQL database (Gautschi, 2013). The structure of the system is shown in Figure 3.4.

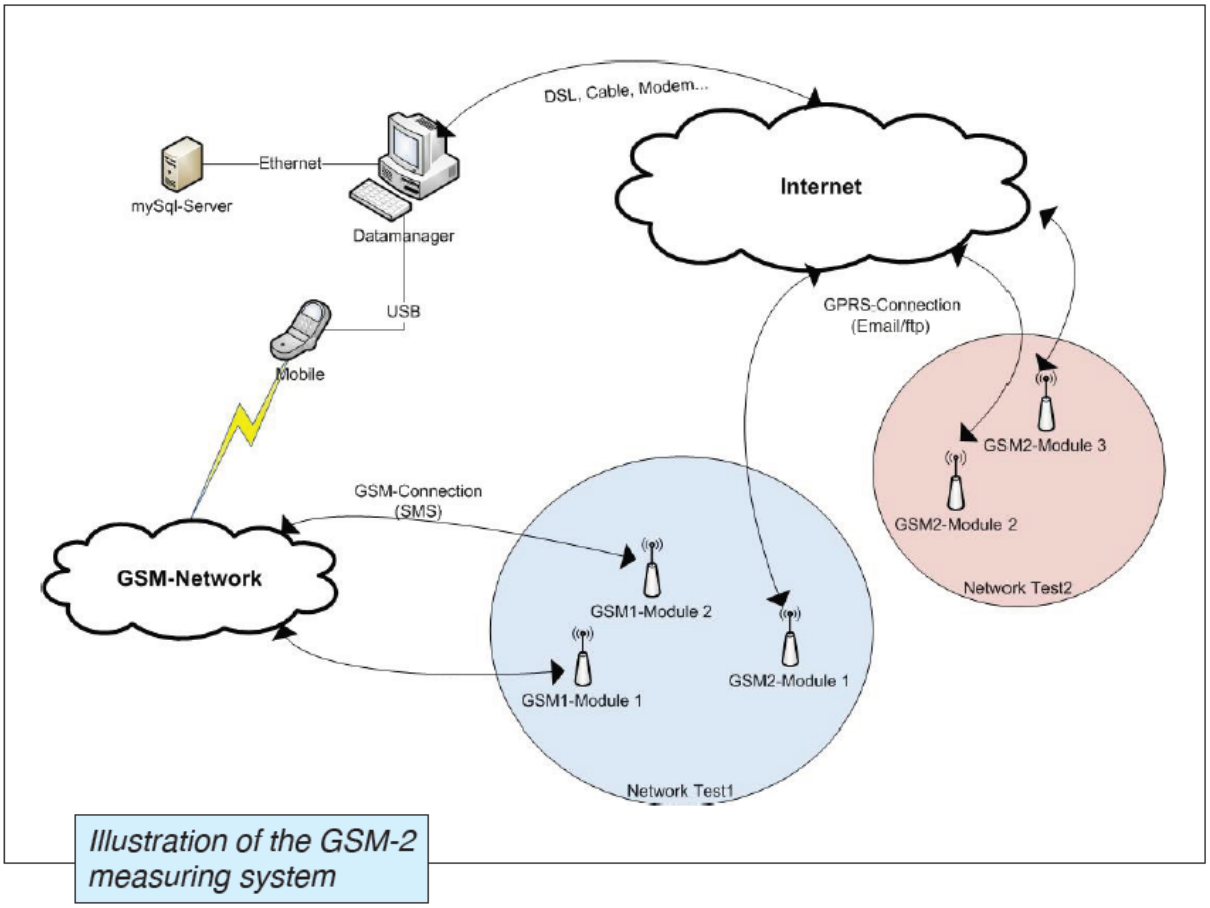


Figure 3.4: GSM-2 measuring system structure.

Conventional maritime wireless communication is based on voice communication and usually operates with radio devices of various radio frequency bands. For remote environments, satellite systems are in use for broad coverage. However, current wireless communications at sea mainly rely on satellite links which are slower than HF and VHF or on expensive Inmarsat. Article (Kim *et al.*, 2014) represents a shore-to-sea maritime communication with Visible Light Transmission(VLT), as showed in Figure 3.5:

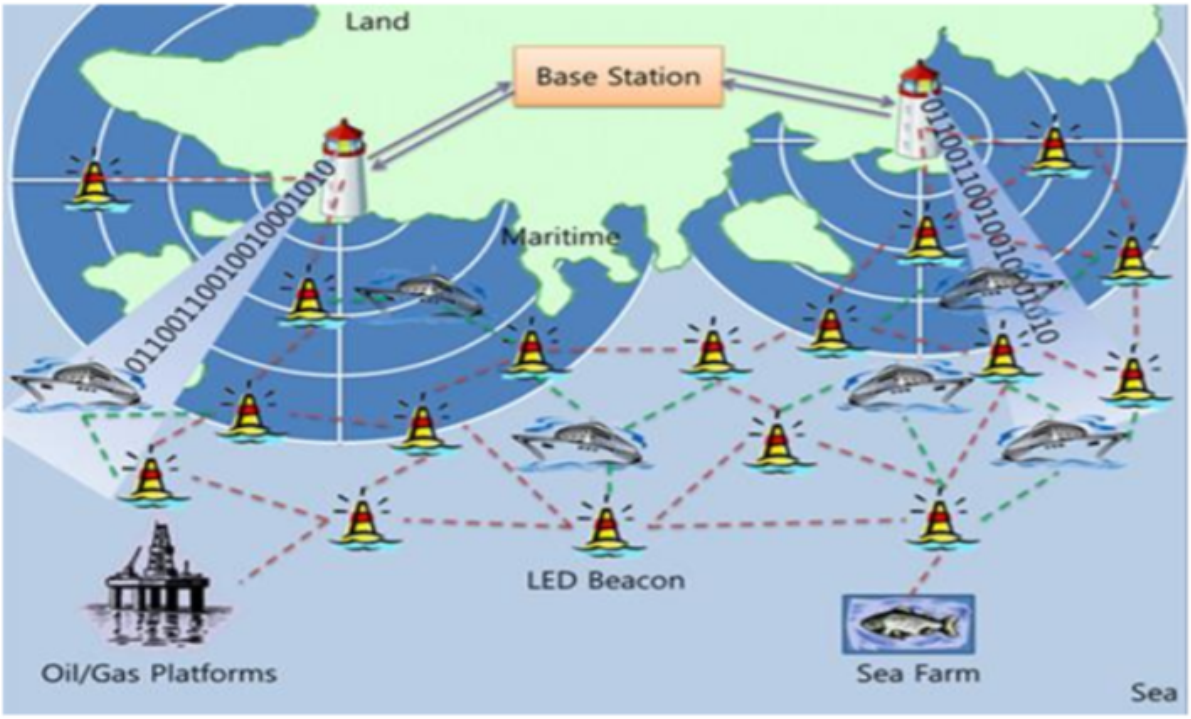


Figure 3.5: High-level architecture of maritime VLT network.

GE applies integrated system wirelessly transmits the sensor's raw data to the GE Wireless Gateway which in turn transmits the data via a cellular connection to InSight, GE's cloudbased industrial internet platform. In the cloud, analytics convert raw data into performance indicators. These performance indicators are accessible via the web. Moreover, mobile platform is also supported with InSight. (Electric, 2015) For different applications, different industrial communications can be applied. For marine applications, the WiYZ Gateway supports up to 4 wireless connectivity options for bridging communication between Mesh, Cellular, WiFi and MDS wireless. Satellite, LTE, WIMAX are all commoned used standards for long distance data transmission.

Imtech Marine and ITC Global have developed a global VSAT network that delivers reliable satellite communications to the entire maritime industry. This network, the Imtech Marine global VSATKu-Band network, has automatic beam switching along all of the major shipping lanes.(Imtech, 2013; Communications, 2015)

Harris CapRock delivered enhanced internet access for Carnival curise line fleet. It of-

fers passengers a wide range of convenient internet access options as well as social media packages by applying hybrid C- and Ku- band network solution. Moreover, Harris CapRock has recently launched the industry's first unified, fully managed satellite, wireless and terrestrial connectivity service designed to reduce customers' voice, data and equipment management costs. It is comprised of a multi-band antenna and an Intelligent Communications Director (ICD). The tri-band antenna allows for C-, Ka- and Ku-band connectivity with any satellite orbiting the Earth can be accessed with no additional equipment installments or upgrades required.(Caprock, 2015)

Inmarsat, which provides satellite communication services, the price is listed in Table 3.3 as a reference.

Table 3.3: Price list.

Pick your Plan:	Small Vessel Plan	Standard Plan
What You Pay per Month	$199	$749
What You Get per Month	5MB data	20MB data
Cost/MB (Data)	$39.8	$37.45

With 265MB data, it can cost around $9737. And this is only the data from 1 engine within 1 day's data expense. So from this point of view, satellite communication is not a feasible solution. With this amount of data, even very high data compression ratio, the data may lose its value for analyzing. However, it is possible to use satellite communication for critical data transmission or emergency situation communication.

From a pure software side, Charter Solution, Inc has proposed a Big Data solution for Time-Series Data: Cloud-based infrastructure services: Amazon Web Services, and Big Data database: Vertical Database (Yhibodeau, 2013). The key elements of the architecture are showed in Figure3.6

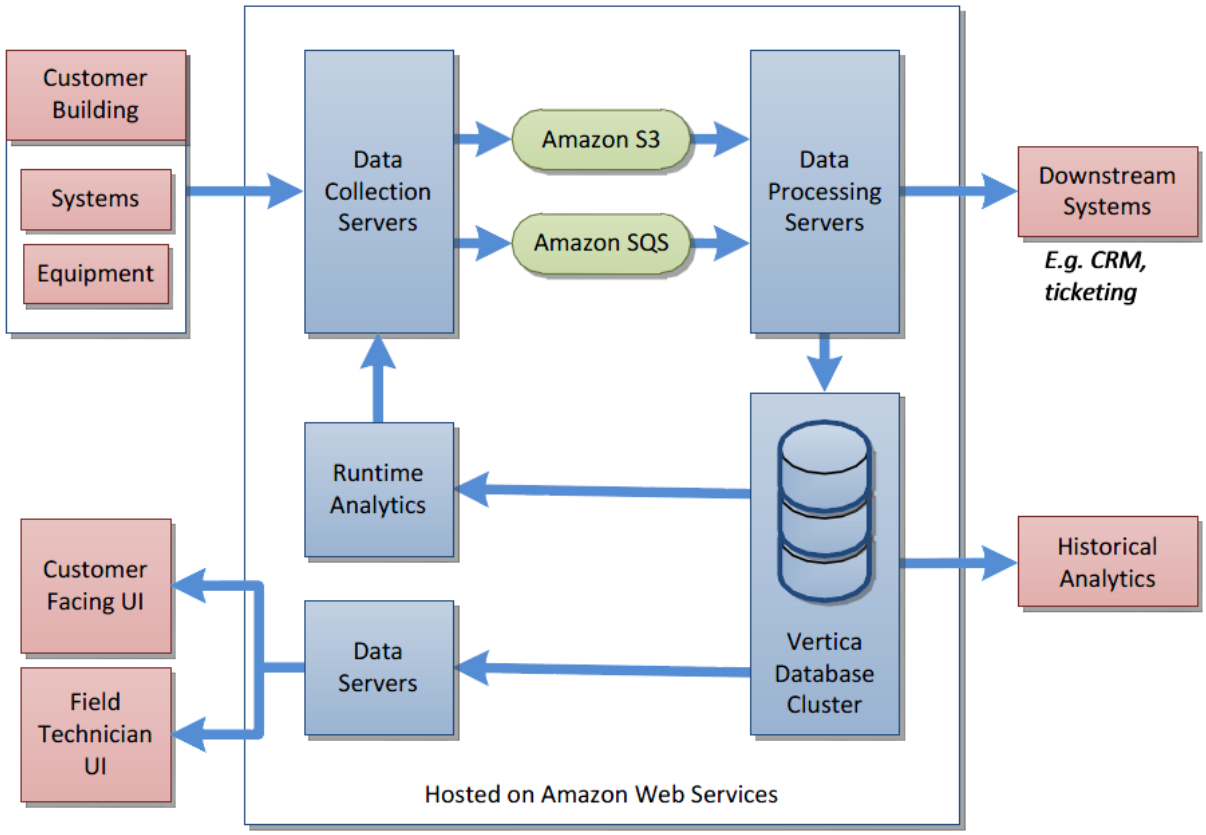


Figure 3.6: Key elements of the architecture.

In general, the communication system designing depends heavily on the available resources, system requirements, financial budget and other practical issues. However, satellite communication is normally used in the situation when there has no other connections available, and other technologies such as LTE, WiMAX, WiFi etc are more flexible to use if they are available.

3.4. Interfacing Wärtsilä Optimisers with 3rd Party Programs

After all requirements are clarified, general functions and the way of approaching the data through the user interface are clear. In order to design a user interface which is intelligent and flexible for users to extract data, two kinds of platforms, which support user interface, are considered:

- Stand alone program
- Web based API

Table 3.4 listed the advantages and disadvantages of stand alone and web-based application.

Table 3.4: Comparision of Standalone and Web-based Application.

	Advantages	Disadvantages
Stand alone solution	fast, secure and a lot of functionality	needs installations, only accessable where it is installed, different versions for different operating systems
Web-based solution	can serve more users, can be accessed from remote computers with different operating systems, easy to maintain and deploy	slow, limited functionality and harder to secure.

One of the main targets of this thesis work is to design a user interface to access data for all Wärtsilä internal experts who are located all over the world. So considering all aspects, the web-based solution suits this work the best.

Wärtsilä Optimisers is operated both on sites where equipment is used and in office where the usage is analyzing and monitoring. So that sites and office are connected in this way: system collecting data from Wärtsilä and third party equipments in site, transfering data

to office. Figure 3.7 (Teräväinen, 2013)shows the overview of the system architecture .

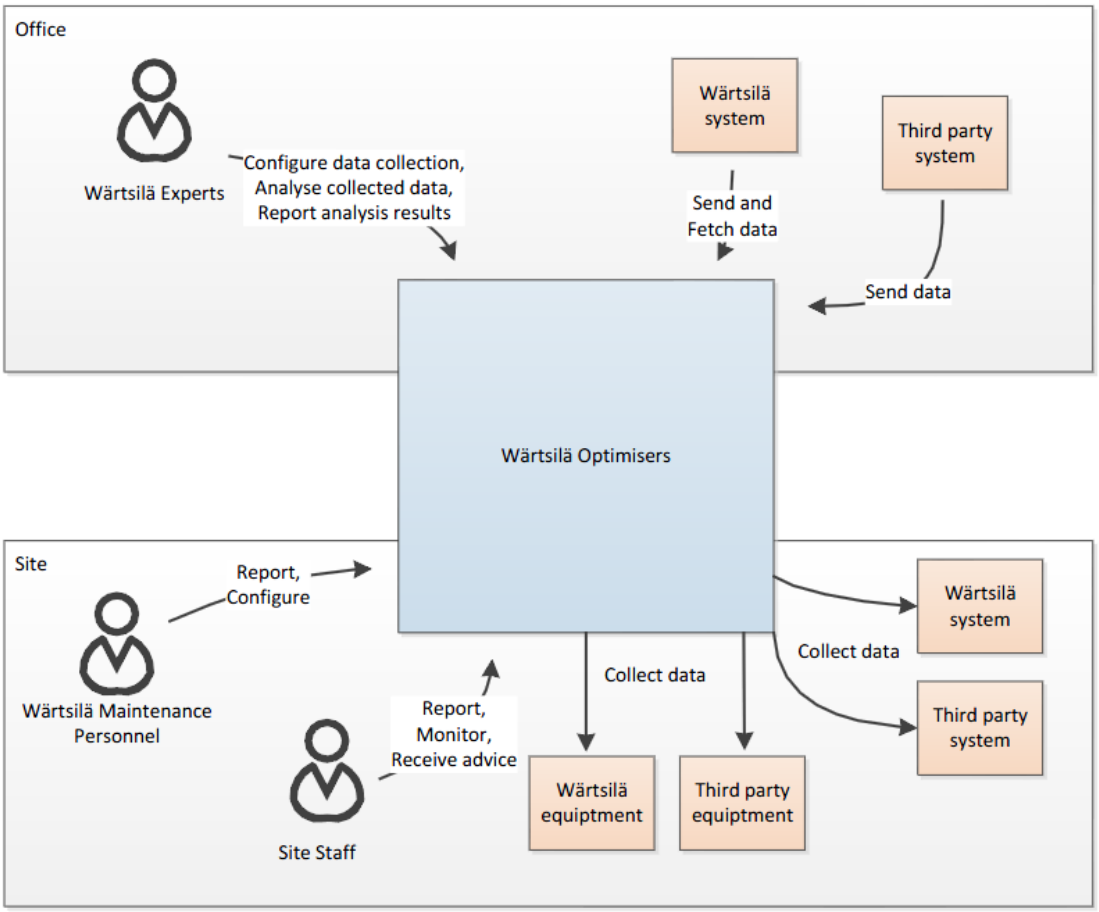


Figure 3.7: Optimisers system overview.

In the sites, all the data is collected to Site Core which is deployed onto a machine with specific hardware requirements. Then data is visualized by Site GUI and Dashboard. Site Core is connected to Center Core, and data collecting signals and methods are all configured by Center Core. Then the collected data is sent to and analyzed in Central. In Central, three different interface protocols can be utilized:

- SOAP(Simple Object Access Protocol)
- REST(Representational State Transfer)
- 3rd party interface

In this work, a platform interface is designed for internal experts based on the 3rd party protocol.

3.5. Designing the User Interface

In this thesis work, a static user interface was designed as a demo. A design is made in order to fulfill all these requirements. Figure 3.8 shows the general map of the whole designing:

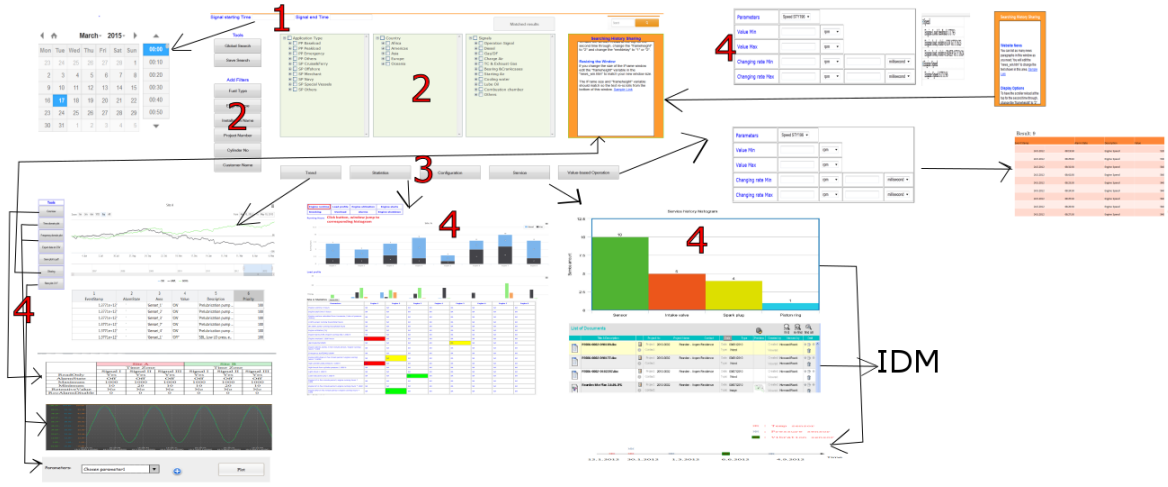


Figure 3.8: Designed user interface map.

As the foundation of a professional website for internal useage, security is always considered as a privilege. Therefore log in process is critical. The function is showed here in Figure 3.9.

Figure 3.9: Optimisers logging in page.

User Interface: Main Page

After the authentication phase, internal users will be lead to the main functionary page, as in Figure 3.10.

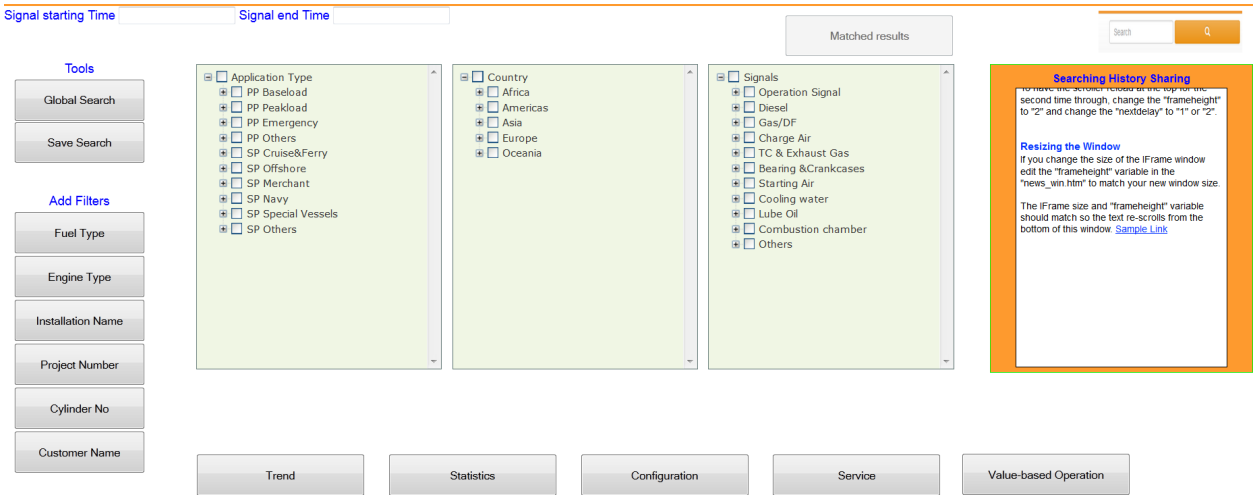


Figure 3.10: Main UI for Optimisers.

In the main page, it can be divided into few parts: time selection part, filter part, funtion selecting part and history searching sharing part. Figure 3.11 shows the main page processing flow chart.

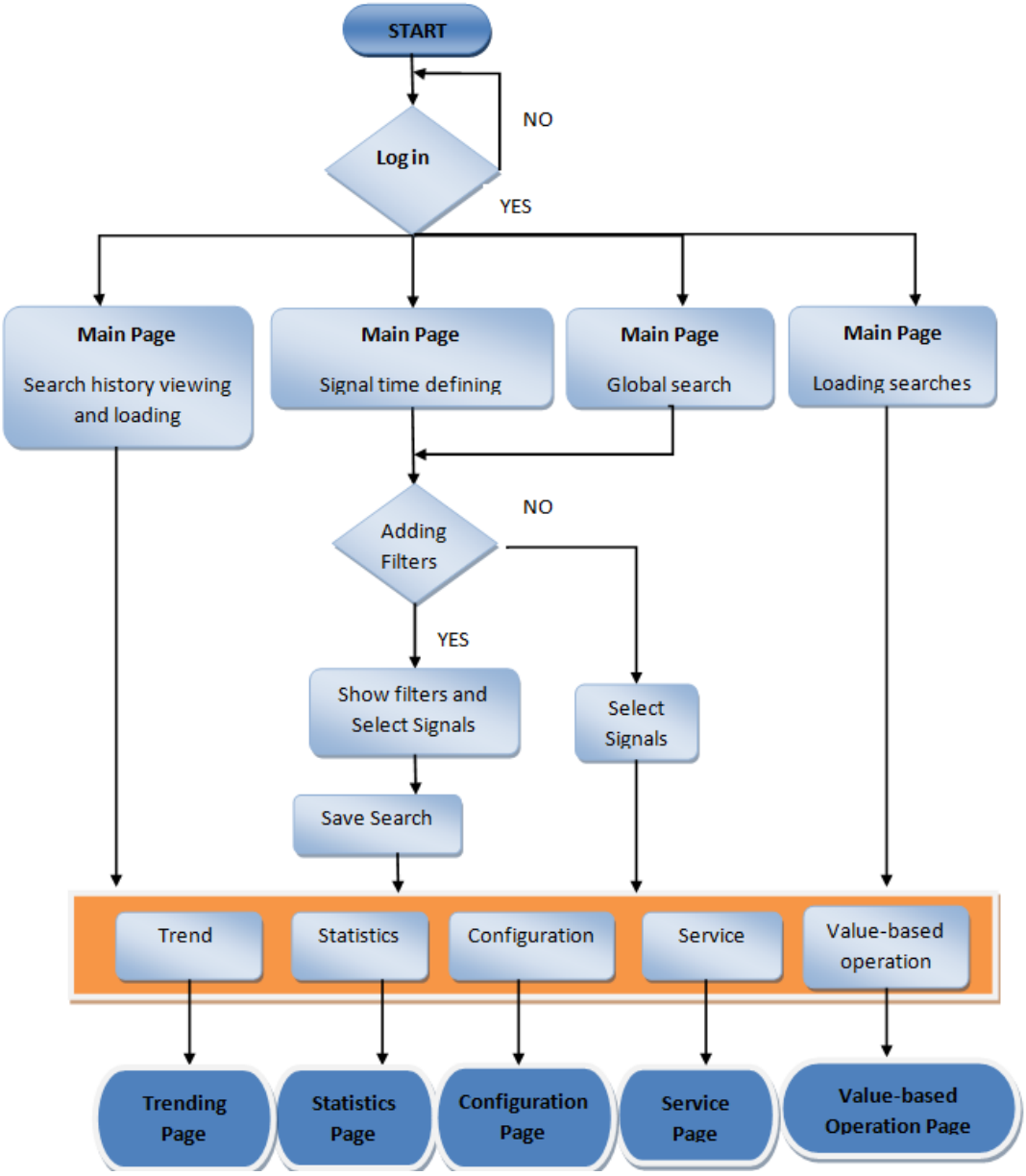


Figure 3.11: Main page processing flow chart.

For time selection part, user can choose the time duration they are interested in by using calendar date picker and clock time picker. The designing is done by using JQuery library, which is a open source JavaScript library for simplifying the client side scripting of HTML. And Date and Time picker is one of the functions that can be easily realised with this library, Figure 3.12 shows those two components.

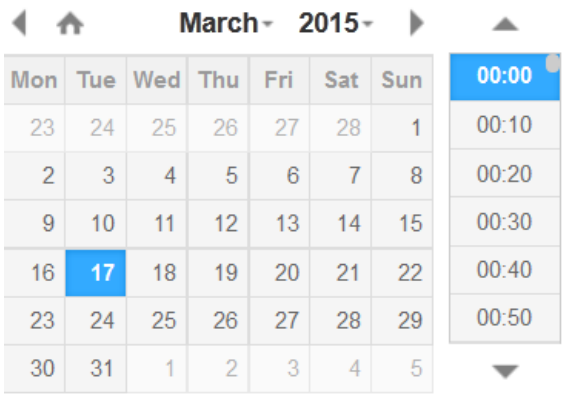


Figure 3.12: Signal time duration define.

For the filter part, another proposal is a straightforward solution which tilings all the filters in the page, as shows in Figure 3.13

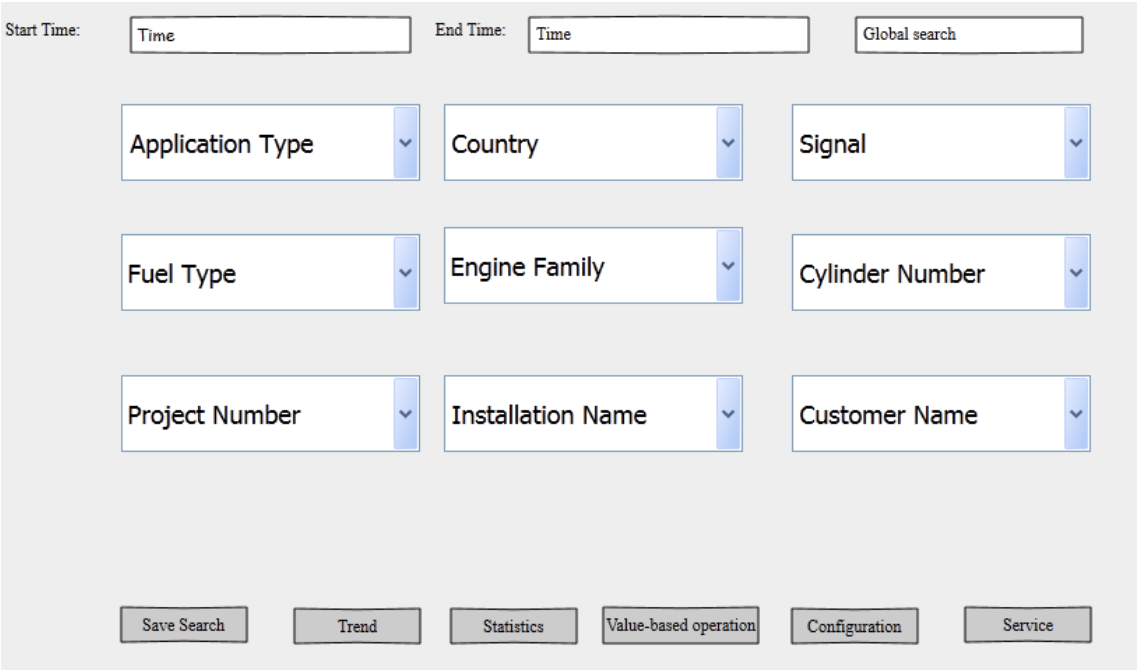


Figure 3.13: Main UI with all filters tiled.

However, considering in practice not all the filters are needed for most users, another

solution is only showing the three most commonly used and basic filters; and flexibility to add six optional filters by clicking buttons, as in Figure3.10. So depending on the searching keywords or information the user have, different filters can be grouped and corresponding data can be extracted as shown in Figure 3.10. In this work, this method is used.

With all the filters, the structure are tree based, here in this work all the filters are designed by using JQuery Tree Plugin library. Data are grouped in different subbranches of the tree, so that user can reach the destination faster than going through the whole tree. Totally nine filters are defined: Application, Continent , Measured parameters, Engine Type, Fuel Type, Cylinder number, Project number, Installation name and Customer name.

In general, when Add Filters button is clicked, the filter is added to the screen with corresponding available tree data based on the previous selection information. Meanwhile, every time when user has made some choices, the total amount of matched result will be displayed. In this way, user can also estimate the time needed for fetching the required data.

Based on different filters, user can search for *trend data*, *statistics data*, *service data*, *configuration data*or even *value-based operation* to do computations. In the following sections, all those functions will be explained, but because *service data* and *configuration data* are not available to access, therefore the interface of those funtions are not concered in this project.

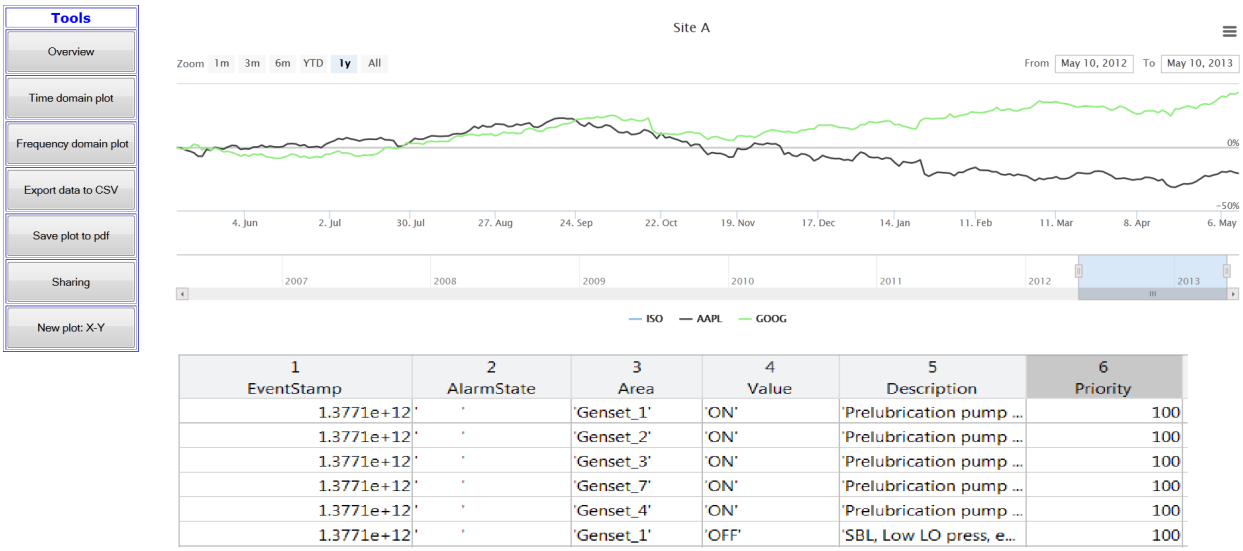
Other operations in main interface are:

- *Global search function*, as the name indicates it is a searching engine in the whole database. In theory it should work in this way: whatever content user puts inside there, the system will try to find the matching sections in all the filters, if anything found available in the filters, then the matched filters will automatically open and show the selected sections, moreover, matched results panel will also indicate the result. From here user can process further.
- *Save search* and *Load search* functions are used for saving current searching settings, and loading back saved settings. In this way, when user discovers something valuable, can still track it back easily later. This saved settings, can be a executable format file, so when loads this file all the filters will have the same parameters as the saved file.

- For the *Searching history sharing* block, it is a platform where users can share and download all the valuable history operations in this application. It is linked to one function which will be explained in subsection 3.5.

User Interface: Trending Page

For trending function, the general interface is proposed as in Figure 3.14.



1 EventStamp	2 AlarmState	3 Area	4 Value	5 Description	6 Priority
1.3771e+12	' '	'Genset_1'	'ON'	'Prelubrication pump ...	100
1.3771e+12	' '	'Genset_2'	'ON'	'Prelubrication pump ...	100
1.3771e+12	' '	'Genset_3'	'ON'	'Prelubrication pump ...	100
1.3771e+12	' '	'Genset_7'	'ON'	'Prelubrication pump ...	100
1.3771e+12	' '	'Genset_4'	'ON'	'Prelubrication pump ...	100
1.3771e+12	' '	'Genset_1'	'OFF'	'SBL, Low LO press, e...	100

Figure 3.14: UI for Trending function and alarm list.

The whole interface consists of Toolbox, Trending plot and Alarm list table. All those blocks are for one individual site, therefore when signals from multiple sites are chosen, the same block will be copied by site. Figure 3.15 shows the flow chart of the trending page processing flow chart.

- *Overview*, as presented in Figure 3.16, gives the general information about the basic information of the sites, timezone, signals and information related to the signals. They can be well presented in table format as shown in Figure: 3.16

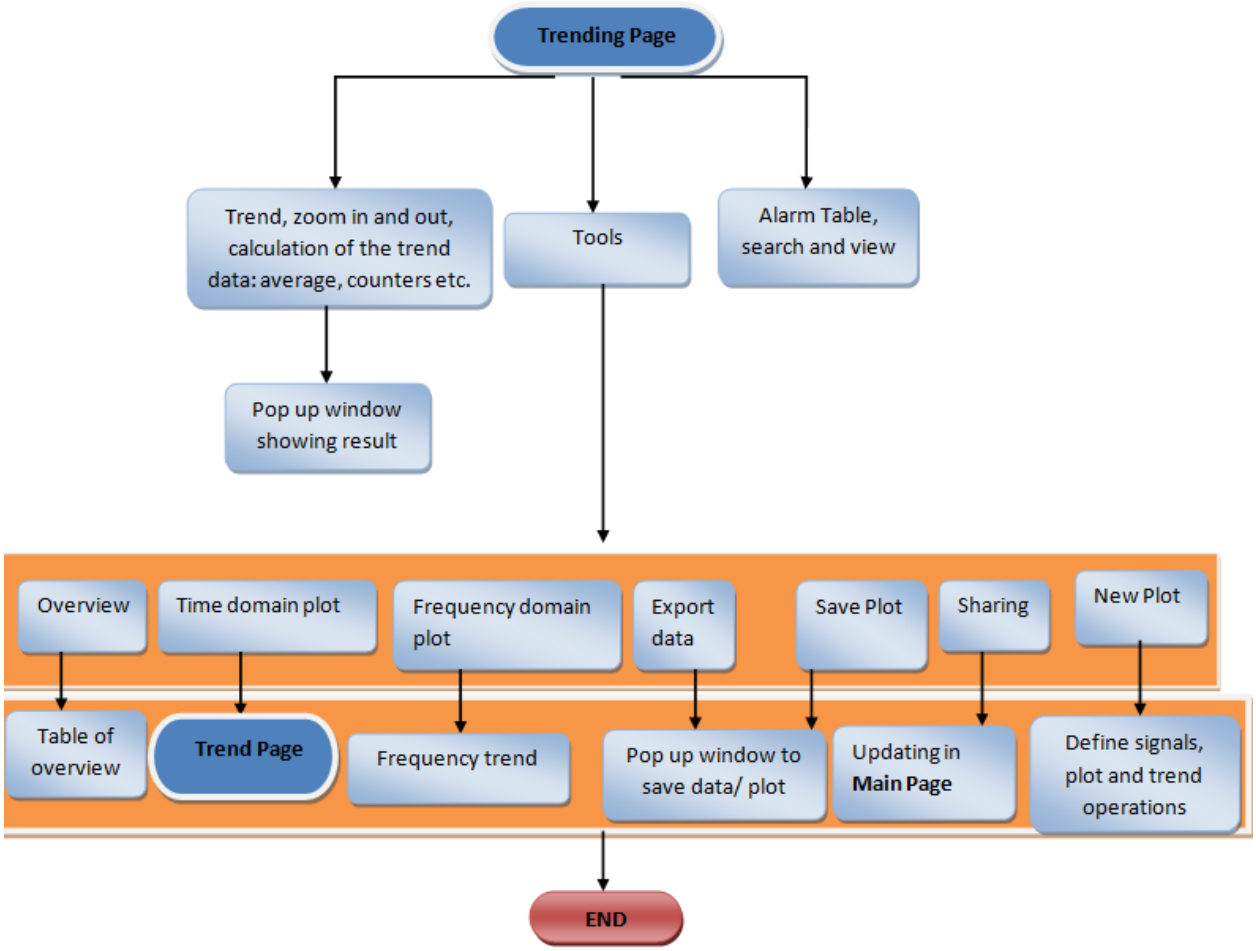


Figure 3.15: Trending page processing flow chart.

	Site A			Site B		
	Time Zone			Time Zone		
	Signal I	Signal II	Signal III	Signal I	Signal II	Signal III
ReadOnly	Yes	Yes	Yes	Yes	Yes	Yes
AlarmState	Off	Off	Off	Off	Off	Off
Maximum	1000	1000	1000	1000	1000	1000
Minimum	10	20	10	10	20	10
RetentiveValue	No	No	No	No	No	No
RocAlarmDisable	0	0	0	0	0	0

Figure 3.16: Overview of Toolbox.

- *Time domain plot* is the default window that opens when the trend button is clicked. Figure 3.14 shows an example, where all the signals are grouped together by site.

  All the signals the user has chosen within one site will be in the same trend window. User can select the time interval that user is interested in, either by inputing in the calendar text or moving the time bar. *Zoom in* function can be applied by manually zoom in or by click the buttons which have predefined time intervals. This trend function still needs the value for different signals in y axis, Figure 3.17 shows an option for this operation.

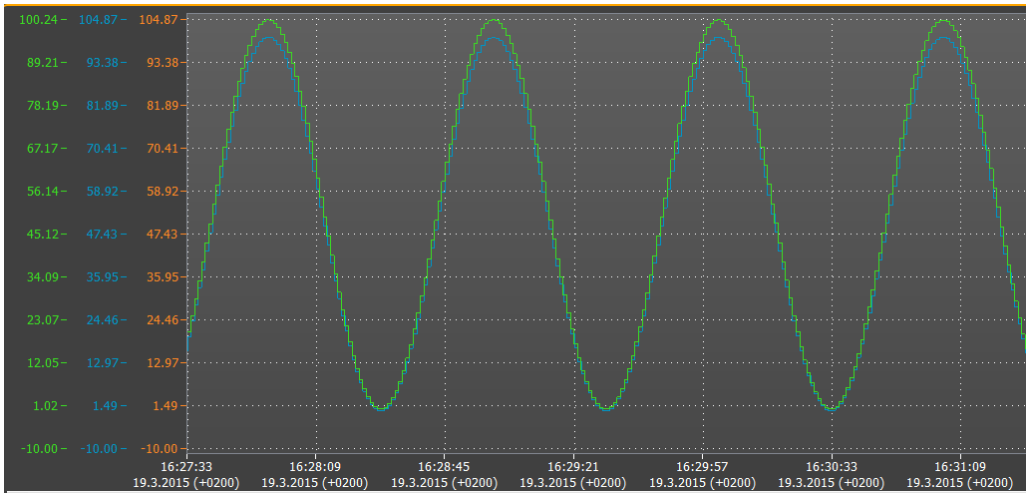


Figure 3.17: Time domain trend, with the values in y axis.

An additional requirement for the trend function is that the trend plot should be able to zoom both in x and y directions. In this way, user can see the details even within a short time interval. The exact values of the trend can be easily seen by putting mouse on it.

One feature which is beneficial to include to the trend function is the possibility to download the plot to other formats like PNG, JPEG, etc.

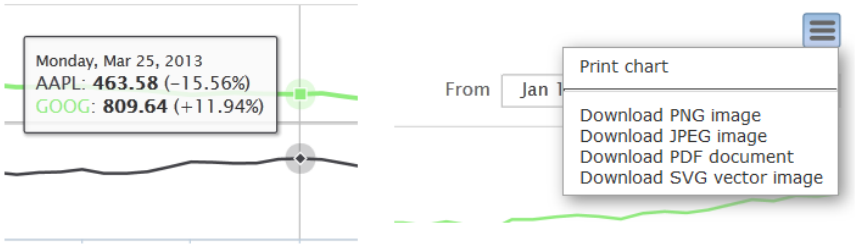


Figure 3.18: Trend value showing by pointing mouse on and plot download toolkit

The left side of Figure 3.18 shows that when mouse points to the trend, the data can be displayed. In this way users can see the exact value as well. The right side of Figure 3.18 shows a convenient method to download the trend into other formats.

Frequency domain plot is just a conversion of the time domain data to the frequency domain.

Export data to CSV will invoke the process of downloading the current time interval signal data and alarm data. And data will be save in CSV format to local files. The fucntion of $saveplottopdf$, is the process of downloading the current trend information to pdf files.

Function *sharing* is for those who has found something interesting or special and would like to share with others. By using the function sharing, the information can be shared either to specific group of users over e-mail or to every user by pinning it to the main page so that other users can notice and even download the shared files. This is the way how the information flow can move between all users.

New plot is the place where users can decide their own plot based on the signals that have been chosen before. In this way users can analyze the relationships between the signals in graphical format. For this function, more functions can be added, like export plot to pdf, modeling relationship into the closet module or formula, coordinate conversion, computation etc.

The *alarm list* is organised in table, where users can either go through the whole table list page by page or just use the search function. For example, when searching priority is 100,

then all the records with priority 100 will be listed only. This table is resizable, so users can also have the whole data in a plain table if needed.

All the components above: toolbox, data trend and alarm list table are site based, so each one of them represents the information of one site. Therefore, if more than one sites are chosen at the beginning, then the the sites will be listed one by one.

User Interface: Statistics Page

At the statistics page, the user first selects the *histogram view*or the *data view*. In the histogram view, the data is presented in the format of histogram separately for each site, and in the data view it is possible to view the site information individually or all together. Figure 3.19 shows the flow chart of the statistics page processing flow chart.

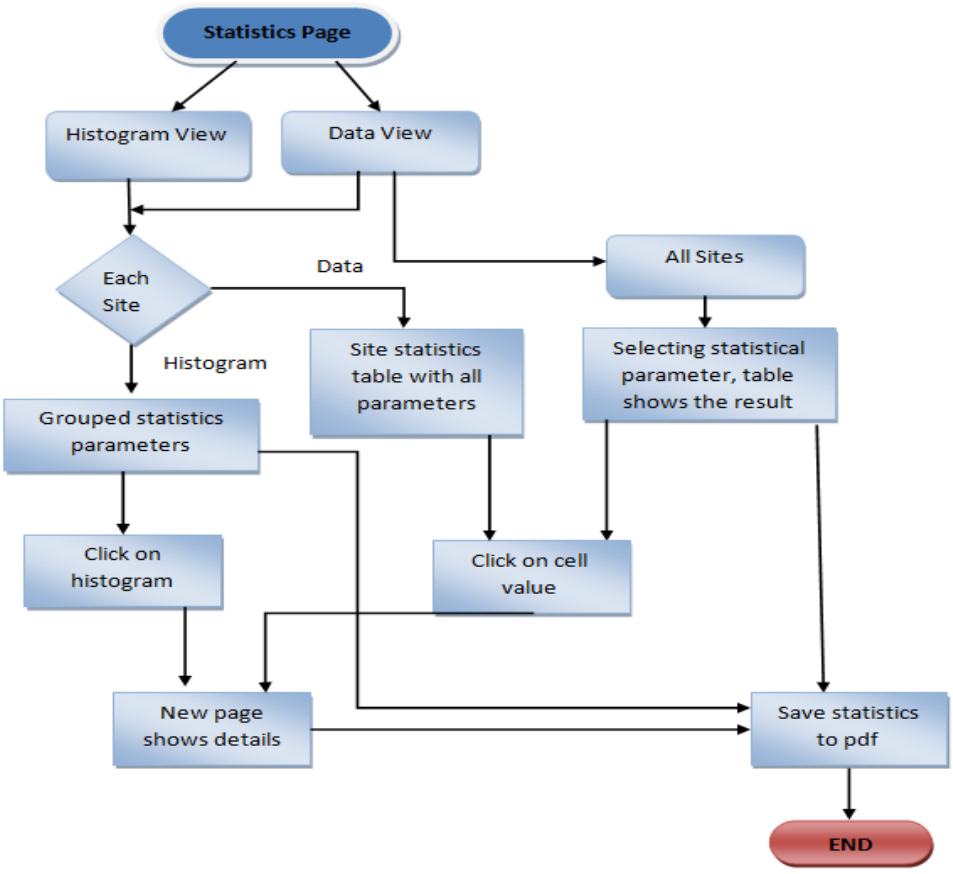


Figure 3.19: Statistics page processing flow chart.

For individual site statistics, both histogram and data formats are available. Figure 3.20 shows the main page of the histogram view.

Figure 3.20: Statistics in histogram view.

In the histogram presentation, all the pre-defined statistical parameters are grouped, and the statistical information is presented. The parameters are grouped as clickable buttons, so that switching among different groups can be done easily. For every individual site, all the parameters are listed one by one. Another way to present the single site statistics is data table, as shown in Figure 3.21 where all the parameters are listed inside a table with the engine number in column field.

Site A Statistics Save to PDF

Parameters	Engine 1	Engine 2	Engine 3	Engine 4	Engine 5	Engine 6	Engine 7	Engine 8
Engine runtime in hours	NA	NA	NA	NA	NA	NA	NA	NA
Engine total time in hours	NA	NA	NA	NA	NA	NA	NA	NA
Engine runtime calculated from timeseries / lube oil pressure (hours)	NA	NA	NA	NA	NA	NA	NA	NA
0-90% power running hours/total hours	NA	NA	NA	NA	NA	NA	NA	NA
80-100% power running hours/total hours	NA	NA	NA	NA	NA	NA	NA	NA
Engine utilization [%]	NA	NA	NA	NA	NA	NA	NA	NA
Engine starts (STB, Engine running ON) / 1000 h	NA	NA	NA	NA	NA	NA	NA	NA
Engine overload / 1000 hours	NA	NA	NA	NA	NA	NA	NA	NA
Alarms(ALM)/1000h	NA	NA	NA	NA	NA	NA	NA	NA
Engine Alarms (ALM,) in five minute period / engine running hours * 1000	NA	NA	NA	NA	NA	NA	NA	NA
Emergency stuff(EMG)/1000h	NA	NA	NA	NA	NA	NA	NA	NA
Engine EMG stops in five minute period / engine running hours * 10000	NA	NA	NA	NA	NA	NA	NA	NA
High cylinder peak pressure / 1000 h	NA	NA	NA	NA	NA	NA	NA	NA
High knock from cylinder pressure / 1000 h	NA	NA	NA	NA	NA	NA	NA	NA
Light knock / 1000 h	NA	NA	NA	NA	NA	NA	NA	NA
Load reductions (LR) / 1000 h	NA	NA	NA	NA	NA	NA	NA	NA
Engine LR in five minute period / engine running hours * 1000	NA	NA	NA	NA	NA	NA	NA	NA
Engine SF in five minute period / engine running hours * 1000	NA	NA	NA	NA	NA	NA	NA	NA
Engine SHD in five minute period / engine running hours * 1000	NA	NA	NA	NA	NA	NA	NA	NA

Figure 3.21: Statistics in table view.

As shown in Figure 3.21, all values are displayed inside table cells. Different conditions are then indicated by different colors, as follows:

- *Red color* indicates emergency or alarm, so immediate operation needs to be taken here in order to prevent or stop further problems.
- *Green color* indicates that the engine is in good condition, so no more extra attention is needed.
- *Yellow color* means the engine is in a third area where engine parmaters fall out of safe boundary, but not in the dangerous range.This may cause dangerous result, so checking up and maintance must be done as soon as possible.

The general engine condition can be shown fast by using this sort of intuitive representation. A property that would enable individual users to define and add new parameters would be beneficial. It can be added so that the permissions to add new parameters can be given by the higher level database manager.

For the case if users prefer to see the view of all the sites at the same time, a data table view is put forward. However it is very demanding to have three diemensions table on the website. Therefore it is structured with the name of site and number of engine as the table fields. The paramteres are listed in dropdown list where the user can select the form of the presentation. Moreover, the color indicator works the same way as in individual table; the whole sample can be represented as in Figure 3.22.

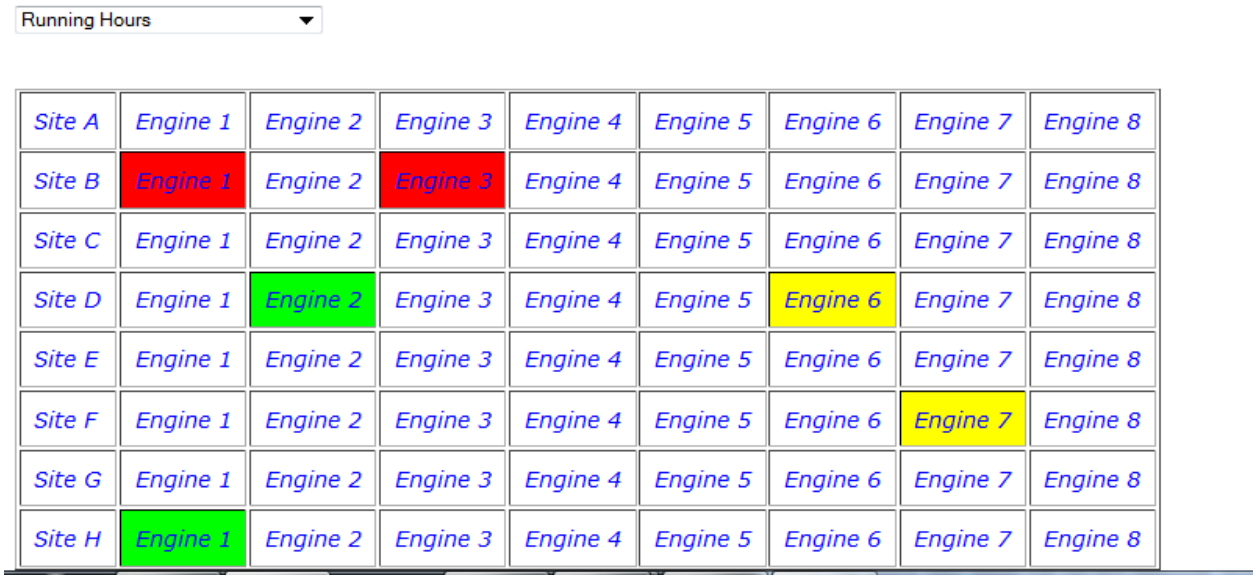


Figure 3.22: Statistics of all sites as a table view.

More operations can be added in the table view, for example *save table to pdf*. Each cell value can also be linked to its own trend plot window. Then if the user for example clicks the Engine overload/1000 hours for Engine 1 in site A because of the red alarm sign showed in 3.21, then the trend will show the whole life cycle of this value. This can give more detailed information than just a value number and colcor indicator. The ability to add new parameters or hide old ones from their individual view, and the shortcuts and roadmaps to different sites would be helpful for individual users.

User Interface: Value-based operation page

Value-based operation, as the name says, it is one function where users can excute operations based on the values of signal data. If the engine, for example, shuts down, the user

wants to understand the reason for that behavior. For that purpose, he wants to know the motor speed distribution during a certain period of time before the shutdown to be able to analyze the phenomenon. By using the *Value-based operation*, the user can define how many times the motor speed has been inside a specific speed interval and how many times it has been out of the interval during a certain period of time.

Figure 3.23 shows the flow chart of the value-based operation.

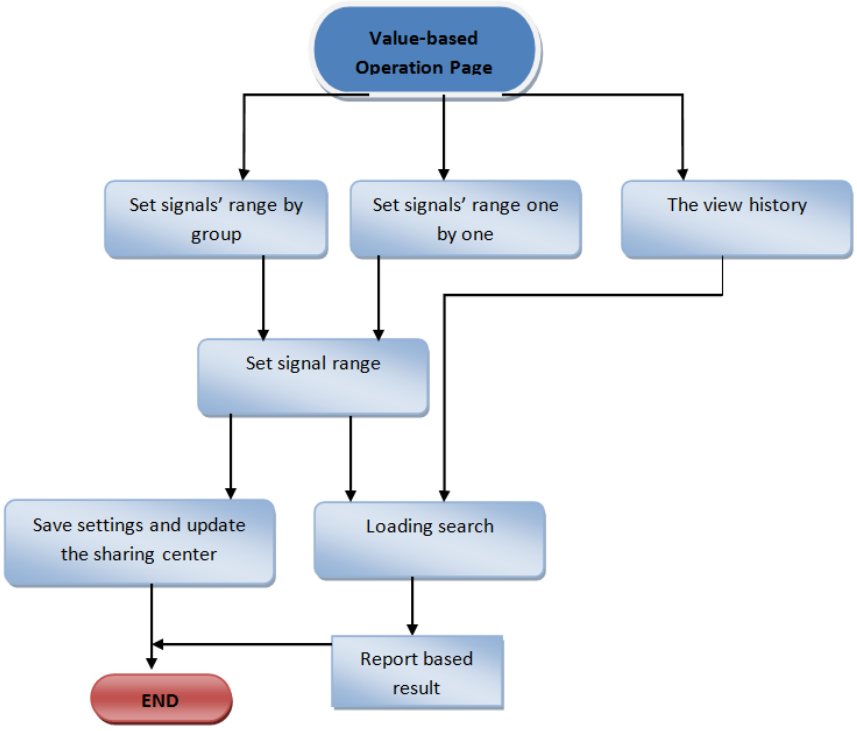


Figure 3.23: Value-based page processing flow chart.

Getting connection back to main page, after signals have been selected within certain time interval, the next step in *Value-based operation* is to define the signals limits. For example, in order to know the times of engine overspeed, the speed limit is defined as minimum 750rpm. As mentioned before, it is possible to select multiple signals at the same time, so the process of setting signals limits should be done for every signals, therefore the proper approach for this question can be:

- Setting signals limits one by one; in this way every signal can have different limits.
- Setting signals limits by group. In each group the signals have the same limits, if

this method is applied.

After that Figure 3.24 shows the value-based operation page where users can set the value limits.

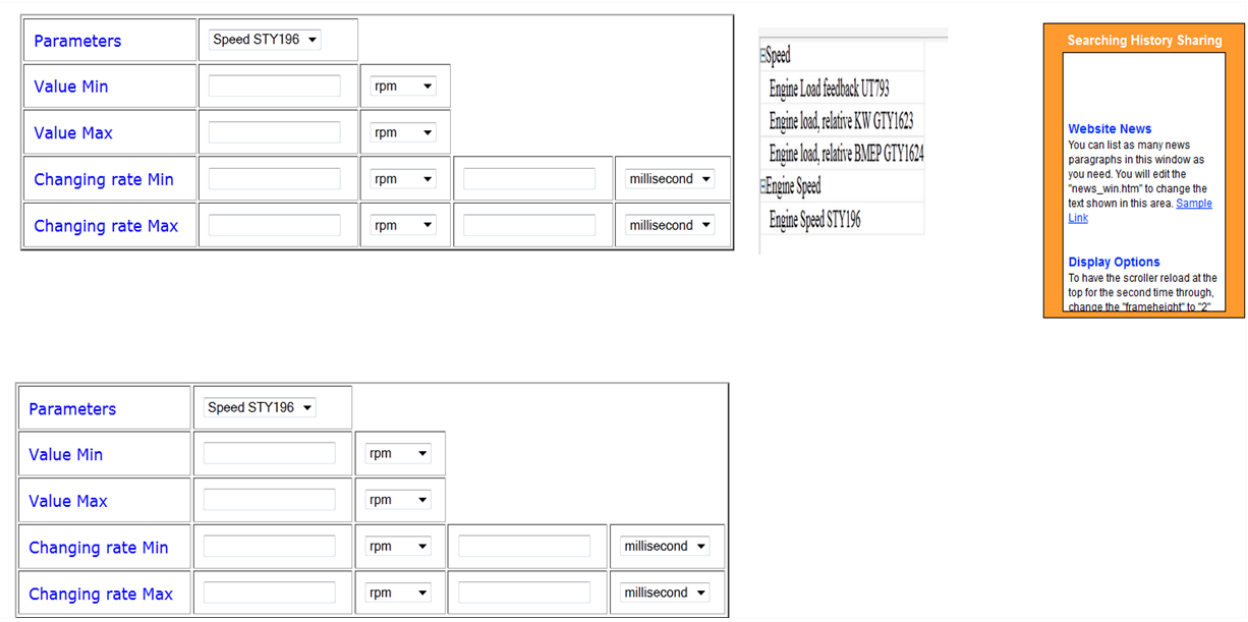


Figure 3.24: value-based operation page.

As shown in Figure 3.24, the user can set up the limits for the signal values he/she searches. Then the possible operations are: Then the possible operations are,

- *Ok button* , which will allow the process of calculation
- *Save settings* has the same function as in main page; users can save their settings into some excuteable file and then share it in its own page sharing center and also in the main page sharing center.
- *Load settings*, is the function the users can use to load the saved file

When process starts to calculate based on the defined range of the parameter values, it will give the result in numerical form. The report is presented as shown in Figure 3.25, where the record of engine speed over 750rpm is given.

Event Stamp	Time	Alarm State	Description	Value
24.5.2012	08:19:30	ON	Engine Speed	755
24.5.2012	08:29:30	ON	Engine Speed	758
24.5.2012	08:32:10	ON	Engine Speed	760
24.5.2012	08:36:35	ON	Engine Speed	763
24.5.2012	08:42:25	ON	Engine Speed	759
24.5.2012	08:46:30	ON	Engine Speed	757
24.5.2012	08:51:20	ON	Engine Speed	758
24.5.2012	08:55:20	ON	Engine Speed	760

Figure 3.25: Result table of engine speed over 750 rpm.

User Interface: Service Page

Service page shows the service history of every site and every component. So all the service history can be attractable. However, in reality it is not always possible to get all that information updated. Especially with components replacing, which happens the highest frequency compare with other service items. For example, in the case of an incorrect measurement caused by sensor failure or some other malfunctioning, the missing value can be estimated by using the other values. So knowing the history of service for all components help experts to diagnose the problem and to find its source during troubleshooting.

Figure 3.26 shows the service page execution flow chart.

For service interface page, two functions are needed:

- *Connecting to service documents in Integrated Document Management (IDM)* , or directly showing all the history list document.
- *Providing specific period operation statistics report*, which contains not only the statistics of the engine, but also gives recommendations and predications for the future. An example can be the CBM monthly report which offers the similar function

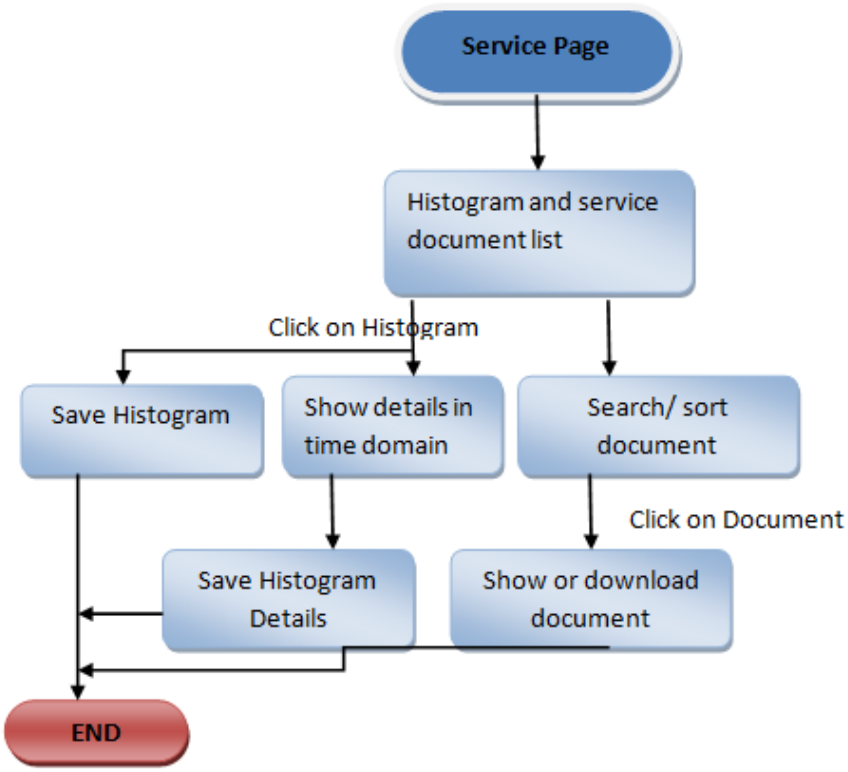


Figure 3.26: Service page execution flow chart.

in a monthly basis.

Moreover, document can be categorised, in this way if the same component will have the same standard report template. With the sensor, for example, the sensor number, manufacturer and ISO code should all be included. However, in the maintenance work a different template should be used in order to record all the activities. Figure 3.27 shows the statistics of services history, then followed by the site based documents with the order of time sequence.

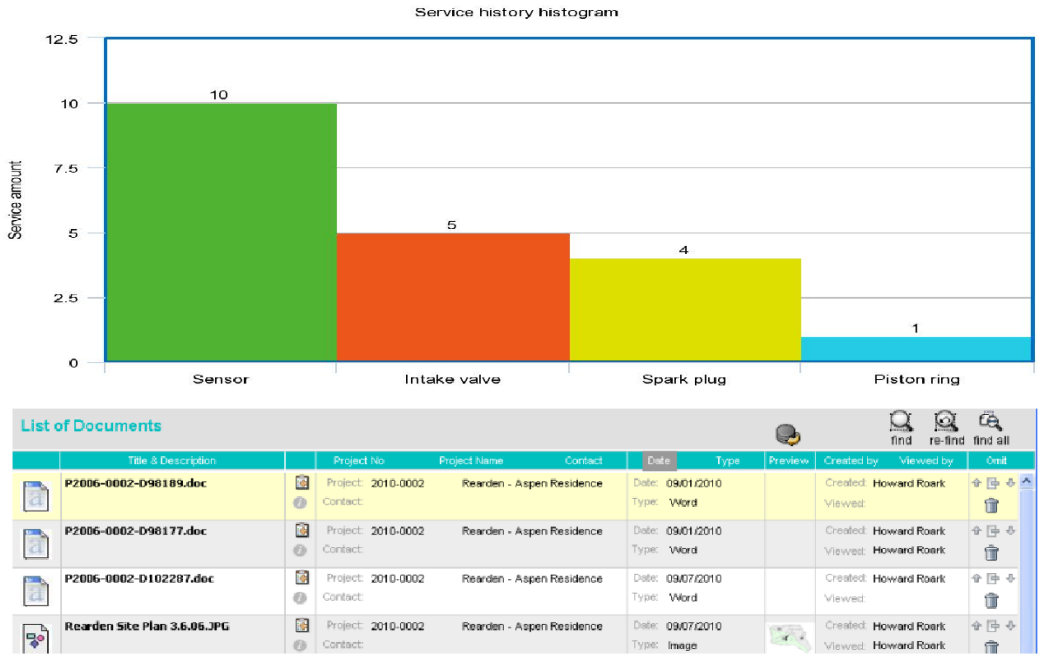


Figure 3.27: User interface for Service page.

Services are categorised into different types; for example the services related to sensors can be one category, spark plug problems can be another one, etc. Moreover, by clicking at every histogram bar, the detailed history record of this specific category will be shown as presented in Figure 3.28

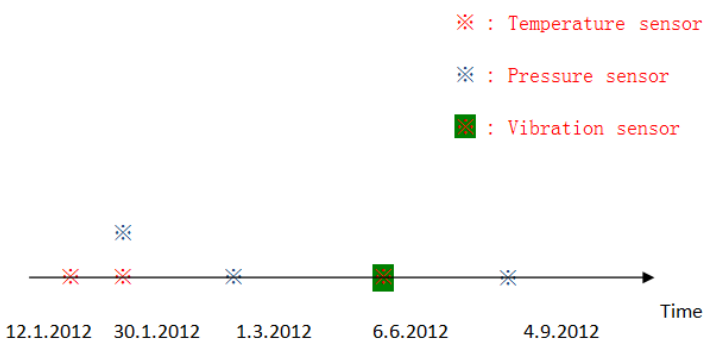


Figure 3.28: Service history for specific category.

Figure 3.28 presents an example of the service history categorization, that can be utilized in trouble shooting. For some cases the location information is also needed for recording, and it can be treated as one category. For different cases different solutions may be needed to meet the requirements. However, the main target is to record and present the data in such a way that it can be correctly analyzed by the experts.

By this way, experts can see the service history statistics both in a general and in detailed way. This may help the experts in troubleshooting, service schedule planing, improving the operation performance in asset management.

User Interface: Configuration Page

Configuration information is also needed for analyzing and troubleshooting. However, currently this information is not available in Wärtsilä Optimsiers. Therefore, the user interface for this operation depends much on later designing.

Figure 3.29 shows the configuration page execution flow chart.

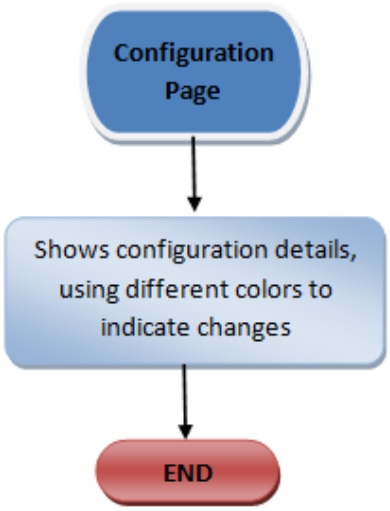


Figure 3.29: Configuration page processing flow chart.

The main operations that are needed for configuration page are summaried:

- Site software version.

- Site parameters values, especially with the one which are different with standard settings.
- Added signals or measurements in the site.
- Modified functions in site.

To summarize, the most important operations for Wärtsilä Optimisers interface are covered in this demo. There might be still many issues to be solved in the practical implementation. However, in reality, how to handle this may very a lot still. But the idea is that all the functions presented here should be included in the Wärtsilä Optimisers user interface. More functions can be added or modified during designing process.

The design of the data warehouse system also impacts to the performance of the user interface, and that must also be considered.

Chapter 4

COMPARISON BETWEEN BIG DATA AND RDBMS

Data, this important and potential resource for society, is leading us to an era of data-driven business and strategy. As a consequence, techniques related to data collecting, data storing, data mining etc, are currently under intensive research and development.

The data can be collected in different formats, such as photo from social media, sensor measurement data, data measured by the mobile devices etc. Both the amount and the cumulation speed of the collected data are increasing significantly. This sets completely new requirements to the computer architectures used for data collection and storing. Furthermore, the data storage techniques also effect on data extraction and analysis. Nowadays, database platforms not only provide the capacity of data storing, but also many kinds of data mining and data analysis tools. Various database platforms, which are applying either relational database or non-relational database approach, are trying to reshuffle the database industry. Relational databases have a long developing history through many years, with reliable, secure and efficient querying properties. They have also a strong position in the markets. However, the concept of Big Data is increasingly attracting attention. Once developed more mature, it is supposed to handle huge amounts of various types of data with a fast execution time. Since there are big expectations related to the concept of Big Data, there is a huge interest into it in the database industry. Big Data is also related to the concept of the Internet of Things (IoT), which is expected to help companies to make faster and more intelligent decisions and improve profits (Wikibooks, 2014). Especially with historical analysis, which is much handy in Big Data analysis,

users can access more data with higher resolution and faster than before. However, Big Data is still an emerging technology and concept without clear specs and standardization. The situation can change in a long run. As suggested by Andreq (Oliver, 2015), in the future Big Data might be the back end which drives all the applications. Even though the concept of Big Data is still emerging, we can present some comparisons between the new concept and the traditional database systems.

4.1. Big Data vs Relational Database

Traditional Relational Database Management Systems (RDBMS)have been applied as a standard through the age of the Internet. The principle in the traditional RDBMSs is that the data is organized in a highly structed manner. There are high expectations related to the Big Data, but a remarkable part of the discussion is also the question that what are the novelties it provides compared to the relational databases. Usually when talking about Big Data, NoSQL, Hadoop and other parallel methods are considered. NoSQL refers to Not Only SQL, which represents its unstructured way of storing data. It is a very well-adapted infrastructure for a database that stores a massive amount of data. In Big Data, Hadoop is also considered. It is not a database but rather a software exosystem or cloud-based platform that enables massive parallel computing. It is enabled by certain types of NoSQL distributed databases, which can allow data to be spread across thousands of servers with little reduction in performance. A comparison between RDBMS and NoSQL is presented in Table 4.1. (ResearchGate, 2013; Software, 2013; Gaywala, 2014)

Table 4.1: Comparision of NoSQL and RDBMS.

	RDBMS	NoSQL
Description	Traditional row-column SQL database management system	non-relational database or distributed database

Architecture	SQL database, mainly Tables with row-column structure	NoSQL has four categorization: • Document base • Key value pair base • Graph database • Wide-column sotre
Systems	MySql, Postgres, MS SQL, Oracle etc	MongoDB, Cassandra, BigTable etc
Schemas and Flexibility	Fixed schema, data can be amended, but requires the whole database updating offline	Dynamic schema, information can be added on the fly, high flexible.
Advantages	Highly strucutred database, fits for complex queries intensive environment and relative data	Well suitable for storing information of a certain type; especially with hierarchical and document base data. Great retrieval speed based on a key. Represent entities in a more natural way
Disadvantages	Massive volume data, unstructed and semi-structured data	Lack of maturity, data consistency might be a concern, not meant for transaction based application
Scalability	Vertical scaling, more data means bigger server, scale RDBMS across multiple servers, however it is difficult and time consuming and expensive	Horizontal scaling, means across servers. Thses servers can be cheap commodity hardware or cloud instances.

Table 4.1 refers that SQL and NoSQL have different and in some cases almost opposite characteristics, but currently many companies are using them concurrently as a hybrid systems which integrate Big Data functions into relational database management. Thus, there is no existing "one system for all" approach; the selection of the applied technology depends on the application as follows:(Rouse, 2014; Philip, 2014)

- Data being analyzed is structured or unstructured. The structured data, even with large volumes, can still be entered, stored, queried and analyzed in a simple and

straightforward way in traditional database. Therefore RDBMS is a better choice for this. Unstructured data is both complex and voluminous which can not be handled or queried by a traditional database, but if the Big Data operations are applied, one can add, aggregate and analyze a vast amount of data from multiple sources without structuring it beforehand. Therefore Big Data operations work better in storing, managing and analyzing large volumes of unstructured data.

- Scalable analytics infrastructure needed. For the case of constant or manageable growing data, then traditional database is the right choise. If the data grows exponentially or needs to have the flexible ability to add servers, then Big Data operations always perform better.
- Fast data analysis. In cases where companies rely on time sensitive data analysis, traditional database is better. For such cases where fast performance is not critical, such as scanning historical data, Big Data functions can analyze large unstructured datasets better.
- At last but not the least is the cost efficiency. When adopting new technologies, the benefits by using it should always be compared with the cost. Even though the tradeoff between the costs and the increased efficiency is sometimes difficult to analyze, it is important to do before making the investment decision.

Moreover, with the possibility to apply a hybrid solution, multiple systems running on premises and cloud will become increasingly popular in the future(Woodie, 2014).

4.2. Data Warehouse In Big Data Concept

The real motivation for industrial companies to utilize Big Data is rising from the needs to achieve better efficiency and profitability in three major categories:

- *Asset and operation optimization* Industrial companies are on the way of improving profit by gathering and analyzing a vast amount of machine sensor data. Companies are trying to improve operation efficiency by utilizing Big Data analysis.
- *Prediction about future asset maintance* It can prioritize maintance tasks, resource allocation and enable the capital to be spend more effectively based on risk assessment.

- In the higher level of mastering Big Data, companies can step into innovative, revenue generating services. Information technology will move from the era of automation based improvements alone to an era of intelligence and value creation.

In general, the Big Data analysis can be divided into two main categories:

- Batch data processing deals with data which is collected over a certain period of time. Collected data is further divided intio batches and processed. For example, in marine industry, the operation data can be send to server once in 24 hours. Once the data analysis is performed, the data is divided into batches and processed by clusters of processors to get the best result out of the Big Data. Thus this kind of processing is used to analyze large sets of history data. Typically Incoop, Scalding and IncMR can be used here.
- Real time data processing is always dealing with real-time or recently arrived data and gives output at near real time or with very latency for proper functioning of system. In terms of Big Data technology, real time data processing usually processes the data, which is coming from various heterogeneous systems. A distributed real time computation system is usually needed for that purpose. Systems like Apache Storm, Trident and S4 support real time data processing.

Furthermore, hybrid solution systems such as Spark, Shark and SummingBird are all equipped with libraries which contain techniques for Big Data processing (Sundaresan & Kandavel, 2014). The choose of the technique that will be used must be based on the nature of data and the application purposes. It is expected that in the future it is possible that a stream processing layer which might be Storm, Spark Streaming etc, can do data processing both for batch and in real time.

However, to truly leverage Big Data analysis in industry, new platforms, data models and analytic capabilities are required. The most important requirements or challenges are summarized as follows: (Kelly & Floyer, 2013)

- *Data availability* It is true that more and more data are generated from Industrial Internet, but white spaces still exists. Every sort of data that could be utilized is still not available, especially in high quality form.
- *Data storage* Since the amount of data is increasing exponentially, one challenge is how and where to store a huge amount of data in order to be accessed rapidly
- *Data privacy*,does not cover only the legal issues but also issues but also techno-

logical and ethical ones. Depending on industry, different rules or policies might be applied. Thus, issues like hampering data sharing and access must be considered as well.

- *High scalability* According to Wikibon's analysis, the data collected by Industrial Internet is growing at twice the speed of other sources of Big Data. Moreover, as a part of the technological development more sensors will be added and higher accuracy data will be obtained. Then eventually more data will be created. So any system used to collect, store and analyze Industrial Internet Big Data must be scalable and take advantage of open solutions such as Hadoop.
- *High security* Since Big Data implementations typically include open source code, there is a risk for unrecognised back doors and default credentials to exist. With Industrial Internet, the data must be highly secure and the system must be updated periodically with new methods to keep up with the developing threats.
- *High flexibility* Industrial Internet consists of many technologies, software and machines. As a consequence, there exists numerous data types, numerous workload and numerous analytic requirements. So any system which applies to this must be flexible to handle this kind of changing environment of technologies.
- *Lacking of professional data science in Big Data skills and tools* This may sound as an easy problem to solve, but in fact many companies do not have the capability for that. They may have even failed in adopting Big Data to their system, partially because of this problem.

The Internet of Things is most probably one of the key factors to drive the trend of the Big Data in the future by creating machine-generated data. Companies are targeting to adopt a data-driven business. In that purpose, they will look to Hadoop or other platforms to support their growth. Once the company is willing to build an industrial application which utilizes the Big Data, the fist two steps in the right technique and platform selection are the following ones. First, one must clarify the requirements of that particular application in which the Big Data is supposed to be utilized. Second, one must recognize the possible obstacles and find a way to go over them. With the Big Data environment itself, weather it is powered by Hadoop, mangoDB NoSQL, Teradata or other system, massive amounts of sensitive data may be managed at any given time. This sort of data requires high security, therefore, the security sets one additional requirement in the system design for the users.

In short term, utilizing Big Data is still challenging and costly. But in the future, the Big Data might be a given and premise, it may be the back end for many kinds of applications.

4.3. Data Warehouse in Relational Database Concept

Relational database has been in the boundary of abandon or continue among many companies since the boom of Big Data started. The Big Data may remarkably change the way how the database systems are designed in the nearby future. Based on the comparision of Big Data and relational database, we can can see that the traditional relational database systems are facing their biggest problems in the handling of the high data velocity, massive data volume and various data varity types of data. But relational database has better performance in inserting, updating, storing and protecting data. Moreover, more and more relational database companies review and evaluate the Big Data technology, determine its value and viability, and then integrate it into their database system products, such as MSSQL, PostgreSQL, MySQL etc.

Thus the relational database management system is going to be developed in such a way that it integrates the traditional databases and functionalities from Big Data. Considering all aspects, relational database still has its position in the database industry.

The current database system, which is used in Wärtsilä Optimisers, is MS-SQL. SQL Server Parallel Data Warehouse is a highly scalable appliance for enterprise data warehousing, which in fact supports Hadoop. Furthermore, data which is collected from Wärtsilä sites is normally in datasets, which contain very specific attributes. These attributes are the measured values of location, temperature, pressure, vibration or even weather the valve is open or not, and so on. The sensor measurements actually generate small datasets in real time that we assimilate into big datasets which can be utilized to provide a historical view. With the small data sets consisting of sensor measurements, one can trigger events based on what is happening now. Those events together with trending information derived from machine learning algorithms can be run against the datasets of Big Data. For example in a wind trubine, a variety of sensors are used to determine wind direction, velocity, temperature and other relevant parameters. The operation of the wind turbine can then be controlled based on the measurement data and the control algorithms. Furthermore, sensor measurement data can be combined with a large set of relevant data.

Then the historical analysis can be done to further develop machine learning algorithms and predict future behaviors. By using MSSQL is possible to keep the advantages of both Big Data and RDBMS.

Chapter 5

CONCLUSIONS AND FUTURE WORK

5.1. Conclusions and Future Work

In this thesis work, Wärtsilä based data mining project with Matlab platform is implemented and applied to Wärtsilä WebWOIS. Based on the database and framwork structure of WebWOIS, Matlab platform is designed to extract data from Hierarchical Data Format directly according to the queries of the user. Authentication, data source files downloading, data extraction, the filling of missing values and data format conversion are the main operations in this task. The system can still be upgraded and developed further:

- In the current work valid URLs are concatenated directly with hard coding. As a consequence, any parts of the URLs are updated, designers have to modify it manually. For example, when WebWOIS will be updated to version 0.7 someday, designers have to change it manually. Therefore a solution for dynamic updating and concatenating the URLs can reduce much work and time in the future.
- The login system is programmed with limited security manner only when user clears all the objects in Matlab workspace or exit Matlab, the user login information is deleted.As a consequence, there is a high possibility that if other users use your computer and they may get your login information. Even if this is only possible for Wärtsilä internal users, but this may lead to disclosure of password. Therefore, this part must be improved in the future.
- The time stamps in the signal data and alarm table are all in the Unix time format,

which makes the time stamp hard to read for users. A conversion of Unix time to human readable format such as ISO format is recommended. This can be done in Matlab with a conversion function. However, this conversion can take extremely long time, especailly when the data size is big. A faster way to make the time stamp conversion is critical and needed in the future.

- Adding more application and calculation examples would help the users to utilize the data more efficiently.

Current Wärtsilä Optimisers user interface is proposed with preliminary needs gathering and system understanding. As a part of this work, the requirements for the user interface were clarified by performing user interviews, assembly line study, a study of the type of the collected data etc. Moreover, optional data compression methods, different communication standards, possible database techniques and possible interfaces are all discussed in this work.

Some tasks that can be planned based on this work are the following ones:

- Wärtsilä Optimisers user interface online testing and evaluation.
- Higher flexibility for Wärtsilä Optimisers user interface in designing phase, such as providing links to go back to previous page etc.
- Wärtsilä Optimisers system re-designing or re-constructing based on different communication standards, applied signal sampling frequencies, unavailable signals sensing and collecting, database or data warehouse modification etc.

5.2. Recommendations

After all the work and comprehensive study about data mining and database, combining with the current system configuration, following recommendations are given for related research in asset management system development.

- Adding sensors to measure the required but currently unavailable signals.
- Improving current sensor networks, to be able to reach sampling frequencies which are high enough for each signals. Updating system to Ethernet based solution.
- Applying the automatic adjustment of the parameters of SDT to compress data as

explained and tested in (Yilin & Wenhai, 2010), to reduce data amount and transmission time.

- Data warehouse system with compressed data which vary between 4.8GB/month to 7.9GB/month from one engine, might be problematic to keep up by using a relational database in a long run. A hybrid solution, which integrates relational database and Big Data databases and functionalities can provide a better performance. However, in a long run, the data management system must have the ability to handle Big Data.
- In future, Big Data may become the back end of many applications. However, with all the uncertainties, the transition from relational databases to Big Data can be risky and costly. Therefore, a well-designed transition phase is needed to be able to keep the reliable performance.
- Network communication methods must be specified based on available resources and application requirements.

References

Bhaskar, V. (2015). "Extraction Methods in Data Warehouse." "http://www.folkstalk.com/2011/04/extraction-methods-in-data-warehouse.html".

Caprock, H. (2015). "Meeting Customers Right Where They Need It : A Satellite Communications Story." "http://www.harriscaprock.com/blog/category/solutions/".

Chhipa, M. K. (2013). "Performance Analysis of Various Transforms Based Methods for ECG Data." *International Journal of Scientific and Research Publication*, vol. 3, no. 5.

Communications, M. S. (2015). "External communication." "http://tinyurl.com/qc2a7hn".

Deri, L., Mainardi, S., and Fusco, F. (2012). "tsdb: A Compressed Database for Time Series." *TMA 2012, LNCS 7189*, pp. 143–156.

Electric, G. (2015). "InSight Wireless Gateway."

et al Int, M. A. J. C. (2014). "Importance of Data Mining with Different Types of Data Applications and Challenging Areas." *Ms. A J. Chamatkar et al Int. Journal of Engineering Research and Applications*, pp. 38–41.

Fink, E. and Gandhi, H. S. (2011). "Compression of Time Series by Extracting Major Extrema."

Gautschi, M. (2013). "Autonomous measuring systems with remote data transmission." Tech. rep., KELLER.

Gaywala, V. (2014). “Difference between SQL and NoSQL database.” "http://tinyurl.com/p385nr6".

General Electric, A. (2014). “Industrial Internet Insights Report For 2015.” Tech. rep., General Electric, Accenture.

Goldstein, R., Glueck, M., and Khan, A. (2011). “Real-Time Compression Of Time Series Building Performance Data.” vol. 14–16.

Grobelnik, M. (2012). “Big Data Tutorial.” "http://www.planet-data.eu/sites/default/files/presentations/Big_Data_Tutorial_part4.pdf".

Guruprasad.K.Basavaraju (2014). “Internet of Things - overview.” "http://www.codeproject.com/Articles/833234/Internet-of-things-Overview".

Han, J., Kamber, M., and Pei, J. (2012). *DATA MINING CONCEPTS AND TECHNIQUES*. Waltham, USA: Morgan Kaufmann, 3rd ed.

Imtech (2013). “Imtech Marine extends Ku-band VSAT network.” "http://tinyurl.com/nrxerbr".

Inc, J. N. (2012a). “Introduction to Big Data: Infrastructure and Networking Considerations.”

Inc, S. E. (2015). “What’s the difference between 4G WiMax and 4G Satellite Platforms?”

Inc, S. I. (2012b). “Big Data Meets Big Data Analytics.” Tech. rep., SAS Institute Inc.

Kelly, J. and Floyer, D. (2013). “The Industrial Internet and Big Data Analytics: Opportunities and Challenges.” "http://tinyurl.com/odp7jct".

Kennesaw (2010). “Cs3310/01 Class Notes.” "http://science.kennesaw.edu/~bsetzer/3310fa10/build/notes/notes0816.html".

Kim, H., tul Sewaiwar, and Chung, Y.-H. (2014). “Shore to Sea Maritime Communication

with Visible Light Transmission." *Recent Advances in Electrical Engineering and Computer Science*, pp. 68–71.

Kotu, V. and Deshpande, B. (2015). *PREDICTIVE ANALYTICS AND DATA MINING*. Waltham, USA: Morgan Kaufmann.

Lin, Z. and Pearson, S. (2013). "An inside look at industrial Ethernet communication protocols."

Oinam, S., Kumar, H., and Patil, S. B. (2013). "Compression of Time series signal using Wavelet Decomposition, Wavelet Packet and Decimated Discrete Wavelet compression Transforms Techniques and their Comparison." *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 2, no. 3.

Oliver, A. C. (2015). "Big Data is dead – long live big data." "http://www.infoworld.com/article/2907872/big-data/big-data-is-dead-long-live-big-data.html".

Padhy, N., Mishra, D. P., and Panigrahi, R. (2012). "The Survey of Data Mining Applications And Feature Scope." *International Journal of Computer Science, Engineering and Information Technology (IJCSEIT)*, vol. 2, no. 3.

Pang, Z. (2013). *Technologies and Architectures of the Internet-of-Things (IoT) for Health and Well-being*. Ph.D. thesis, KTH Royal Institute of Technology.

Philip, N. (2014). "Hadoop vs. Traditional Database: Which better serves your Big Data business needs?" "http://www.qubole.com/blog/big-data/hadoop-vs-traditional/".

ResearchGate (2013). "The difference between RDBMS and NoSQL." "http://tinyurl.com/oczt58y".

Rogers, A. (2014). "IoT technology architecture." "http://tinyurl.com/qjqrktb".

Rouse, M. (2014). "Big Data analytics." "http://tinyurl.com/c35rblw".

Smith, S. (2008). *Study of Marine Communications Systems and Selection Procedures for Implementing Solutions*. Bachelor's thesis, Helsinki Metropolia University of Applied Sciences.

Software, S. (2013). "Big Data 101 NoSql, Hadoop, Analytic DS, RDBMS Differences for Business People." "http://tinyurl.com/o54r8wq".

Sundaresan, B. and Kandavel, D. (2014). "Big Languages for Big Data: A study and comparison of current trend in data processing techniques for Big Data." *Design and Implementation of Porgramming Languages Seminar, At TU Darmstadt*.

Teräväinen, T. (2013). "Wärtsilä Optimisers Architectural Design Document." Tech. rep., Wapice.

UNIQ (2014). "IEEE 2014 PROJECTS." "http://www.ieeefinalyearprojects.org/Data_mining_projects_in_java_dotnet.html".

Walker, M. (2012). "Big Data Analytics Infrastructure." "http://tinyurl.com/ovm3k6k".

Wikibooks (2014). "I dream of IoT/Chapter 5 : IoT and Big Data." "http://tinyurl.com/osbxumb".

Wikipedia (2014). "CC-Link Industrial Networks." "http://tinyurl.com/3m4dww8".

Wikipedia (2015a). "5G." "http://en.wikipedia.org/wiki/5G".

Wikipedia (2015b). "Asset management." "http://tinyurl.com/28scx2".

Wikipedia (2015c). "Data Fusion." "http://tinyurl.com/prdr2md".

Wikipedia (2015d). "EtherNet/IP." "http://tinyurl.com/3szkp3h".

Wikipedia (2015e). "POWERLINK." "http://tinyurl.com/3ln94tp".

Wikipedia (2015f). “PROFINET.” "http://tinyurl.com/pkofb2j".

Woodie, A. (2014). “Hybrid Analytics Yield Big Data Flexibility.” "http://tinyurl.com/q2r5nbw".

Ye, N. (2014). *DATA MING*. Boca Raton: CRC Press.

Yhibodeau, D. (2013). “A Big Data Solution for Time-Series Data.” Tech. rep., Charter Solution, Inc.

Yilin, Q. and Wenhai, W. (2010). “Automatic Parameter Control SDT Algorithm for Process Data Compression.” *Computer Engineering*, vol. 36, no. 22.

APPENDIX A

WebWOIS API code

```
clc
clear all

% load the function file

url_wois = 'http://fis8038.accdom.for.int/wois-pub/wois_datareader.m';
url_json = 'http://fis8038.accdom.for.int/wois-pub/jsonreader.m';
url_login = 'http://fis8038.accdom.for.int/wois-pub/logindlg.m';

[~, ~, ext ] = fileparts(url_wois);
download1=strcat('wois_datareader',ext);
file1=websave(download1,url_wois);

[~, ~, ext ] = fileparts(url_json);
download2=strcat('jsonreader',ext);
file2=websave(download2,url_json);

[~, ~, ext ] = fileparts(url_login);
download3=strcat('logindlg',ext);
file3=websave(download3,url_login);

% settings
site = 'CAN';
tag = 'BAG011UP01CV';
start_time = '2013-01-11 00:00:01';
end_time = '2013-01-28 20:42:18';
group = 'None';
hide_priority_100 = 0;
hide_priority_500 = 0;
hide_priority_700 = 0;
hide_off_values = 0;
title = 'BAG011UP01CV - Active power setpoint';
```

```
33
34 % Signal data:signal_data, Alarm list table: alarm_table in workspace
35 % 5 optional input parameters
36 [signal_data, alarm_table] = wois_datareader(site, tag, start_time, end_time, group, ↩
       hide_priority_100, hide_priority_500, hide_priority_700, hide_off_values);
```

```
1  function [data,alarm_table]=wois_datareader(site,tagname,start_time,end_time,opt1,opt2↩
       ,opt3,opt4,opt5)
2
3  %Function for downloading and reading time series data and alarm data.
4  %
5  %Input site,tagname, start_time and end_time are all known parameters from
6  %server.
7  %
8  % opt1: optional parameter which defines show only current genset
9  % opt2: optional parameter which defines hide priority 100
10 % opt3: optional parameter which defines hide priority 500
11 % opt4: optional parameter which defines hide priority 700
12 % opt5: optional parameter which defines hide off values
13
14 %Output data is the timeseries data
15 %Output alarm_table is the alarm table
16
17 % Version 1
18 % Date: 19.03.2015
19
20
21 %%
22
23 %%Time converting part: from ISO time format to UNIX time index
24 start_time=int32(floor(86400 * (datenum(start_time) - datenum('01-Jan-1970'))));
25 end_time=int32(floor(86400 * (datenum(end_time) - datenum('01-Jan-1970'))));
26
27 sunix=int2str(start_time);
28 eunix=int2str(end_time);
29
30 % website address assembling: when WebWOIS version is changed, only change 0.6
31 url1='http://fis8038.accdom.for.int/wois-0.6/sites/';
32 url2='/download_h5_file?tag=';
33 url3='.h5';
34 url=strcat(url1,site,url2,tagname,url3);
35
36 url4='/tags.json';
37 url_tags=strcat(url1,site,url4);
38
39 if strcmp(opt1,'None')
40     opt1='';
41 else opt1=strcat('&genset=',opt1);
42 end
43
44 if (opt2==0)
```

```
45      opt2='';
46  else  opt2='&hide_priority_100=1';
47  end
48
49  if (opt3==0)
50      opt3='';
51      else  opt3='&hide_priority_500=1';
52  end
53
54  if (opt4==0)
55      opt4='';
56      else  opt4='&hide_priority_700=1';
57  end
58
59  if (opt5==0)
60      opt5='';
61      else  opt5='&hide_off_values=1';
62  end
63
64
65  url5='/alarmdb.json?sunix=';
66  url6='&eunix=';
67  url_alarmdb=strcat(url1,site,url5,sunix,url6,eunix,opt1,opt2,opt3,opt4,opt5);
68
69
70  %find the name of the file you want
71  [~,  ~, ext ] = fileparts(url);
72  name=strcat(site,'-',tagname);
73  wanted_h5=strcat(name,ext);
74
75  [~,  ~, ext1 ] = fileparts(url_tags);
76  tag_name=strcat(site,'-tags');
77  wanted_tagsjson=strcat(tag_name,ext1);
78
79  alarmdb_name=strcat(site,'-alarmdb');
80  wanted_alarmdbjson=strcat(alarmdb_name,ext1);
81
82
83  % download 3 files totally: h5 for data, json for tag and json for alarm list
84   [user,password] = logindlg('Title','Login Title');
85   options=weboptions('Username',user,'Password',password,'Timeout',Inf);
86   websave(wanted_h5,url,options);
87   websave(wanted_tagsjson,url_tags,options);
88   try
89
90   websave(wanted_alarmdbjson,url_alarmdb,options);
91   [alarm_cell]=jsonreader(wanted_alarmdbjson);
92   alarm_struct=[alarm_cell{:}];
93   alarm_table=struct2table(alarm_struct);
94   alarm_table=[alarm_table(:,5) alarm_table(:,2) alarm_table(:,4) alarm_table(:,6) ↩
         alarm_table(:,1) alarm_table(:,3)];
95   delete(wanted_alarmdbjson);
96   catch
```

```
97      alarm_table='null';
98      fprintf('Alarm list does not exist');
99   end
100  %% data processing
101 %Assembling the h5 datasetname
102 dataset_selection=strcat('/',tagname,'/table');
103
104 %%Read data from h5 file and convert it to mat file
105 %only mat file can process the comparison and calculation etc.
106 data1=h5read(wanted_h5,dataset_selection);
107 data_table=struct2table(data1);
108 data_cell=table2cell(data_table);
109 data_mat = cell2mat(data_cell(:, 1));
110 data_mat1 = cell2mat(data_cell(:, 2));
111
112 %Find the neighbour points(because the index interval is not fixed)
113 % this range is wider than the exact time range
114 %double can keep the precision of index and value
115 thresholdpoint_start = find(data_mat > start_time, 1)-1;
116 thresholdpoint_end = find(data_mat >= end_time, 1);
117 for k=1:thresholdpoint_end-thresholdpoint_start+1
118     data_ss(k,2)=double(data_mat1(thresholdpoint_start+k-1,1));
119     data_ss(k,1)=double(data_mat(thresholdpoint_start+k-1,1));
120
121
122 end
123
124 %Filling miss points
125 %create new range
126 Frange=[data_ss(1,1):data_ss(end,1)]';
127 % find the position where the values fall on the range
128 loc=find(ismember(Frange,data_ss(:,1)));
129 %create new matrix to insert data
130 FilledData=[Frange zeros(size(Frange))];
131 loc=[loc;length(Frange)+1];
132 % propogate the data to the next point known
133 for ind=1:length(loc)-1
134     range=[loc(ind):loc(ind+1)-1]';
135 FilledData(range,2)=data_ss(ind,2);
136 end
137
138 % Go through again to find the exact points
139 thresholdpoint_start_real = find(FilledData ==start_time, 1);
140 thresholdpoint_end_real = find(FilledData == end_time, 1);
141 data=FilledData(thresholdpoint_start_real:thresholdpoint_end_real,:);
142 % converte from UNIX to readable ISo time format. but ...too slow
143 %x=datestr([1]*(double(Data(:,1))./86400+datenum(1970,1,1,0,0,0)),'yyyymmddTHH:MM:SS');
144 % simple plot of the data
145 plot(data(:,1),data(:,2));
146 %alarm list, from json file, to table
147 delete(wanted_h5,wanted_tagsjson);
148 end
```