Technical University of Denmark

CleverFarm - A superSCADA system for wind farms

Giebel, Gregor; Juhl, A.; Hansen, K.G.; Giebhardt, J.; Pahlke, T.; Waldl, H.-P.; Rebbeck, M.; Brady, O.; Ruffle, R.; Donovan, M.H.; Bjerge, C.

Publication date:
2004

Document Version
Publisher's PDF, also known as Version of record

Risø-R-Report

Risø-R-1444(EN)

CleverFarm® - A SuperSCADA system for wind farms

Gregor Giebel (ed.), Axel Juhl, Klaus Gram Hansen, Jochen Giebhardt, Thomas Pahlke, Hans-Peter Waldl, Matt Rebbeck, Oisin Brady, Rachel Ruffle, Martin Heyman Donovan, Claus Bjerge

Risø National Laboratory
Roskilde
Denmark
August 2004

Author: Gregor Giebel (ed.), Axel Juhl, Klaus Gram Hansen, Jochen Giebhardt, Thomas Pahlke, Hans-Peter Waldl, Matt Rebbeck, Oisin Brady, Rachel Ruffle, Martin Heyman Donovan, Claus Bjerge
Title: CleverFarm® - A SuperSCADA system for wind farms
Department: Wind Energy Department

Risø-R-1444(EN)
August 2004

Abstract:
The CleverFarm project started out to build an integrated monitoring system for wind farms, where all information would be available and could be used across the wind farm for maintenance and component health assessments. This would enable wind farm operators to prioritise their efforts, since they have a good view of the farm status from home. A large emphasis was placed on the integration of condition monitoring approaches in the central system, enabling estimates of the remaining lifetime of components, especially in the nacelle.
During the 3½ years of the project, software and hardware was developed and installed in two wind farms in Denmark and Germany. The connected hardware included two different condition monitoring systems based on vibration sensors from Gram&Juhl and ISET, plus a camera system developed by Overspeed. Additionally, short-term predictions of the wind farm output were delivered by DMI and Risø's Prediktor system throughout the period of the project. All these diverse information sources are integrated through a web interface based on Java Server Pages.
The software was developed in Java, and is delivered as so-called CleverBeans. The main part of the software is open-sourced.
The report contains the experiences and results of a one-year experimental period.
This report is a slightly edited version of the final publishable report to the EU Commission as part of the requirements of the CleverFarm project.

ISSN 0106-2840
ISBN 87-550-3291-5
ISBN 87-550-3292-3 (Internet)

Contract no.:
ERK6-CT-1999-00006

Group's own reg. no.:
1105 051-1

Sponsorship:
EU Commission, DG Research

Cover : The logo of the CleverFarm project

Pages: 57
Tables: 6
References: 8

Risø National Laboratory
Information Service Department
P.O.Box 49
DK-4000 Roskilde
Denmark
Telephone +45 46774004
bibl@risoe.dk
Fax +45 46774013
www.risoe.dk

Contents

Preface

This report is essentially the publishable part of the final report to the EU Commission of the CleverFarm project (“Advanced management and surveillance of wind farms”, funded by the EU under ERK6-CT-1999-00006). The project ran from April 2000 to September 2003. Partners were Risø National Laboratory, Roskilde (DK), Gram&Juhl Aps, Haderslev (DK), ISET (Institut für Solare Energieversorgungstechnik e.V.), Kassel (DE), Carl von Ossietzky Universität, Oldenburg (DE), Overspeed GmbH, Oldenburg (DE), RES (Renewable Energy Systems Ltd), St. Albans (UK), DMI (Dansk Meteorologisk Institut), Copenhagen (DK), and E2 energi (signed as SEAS), Copenhagen (DK).

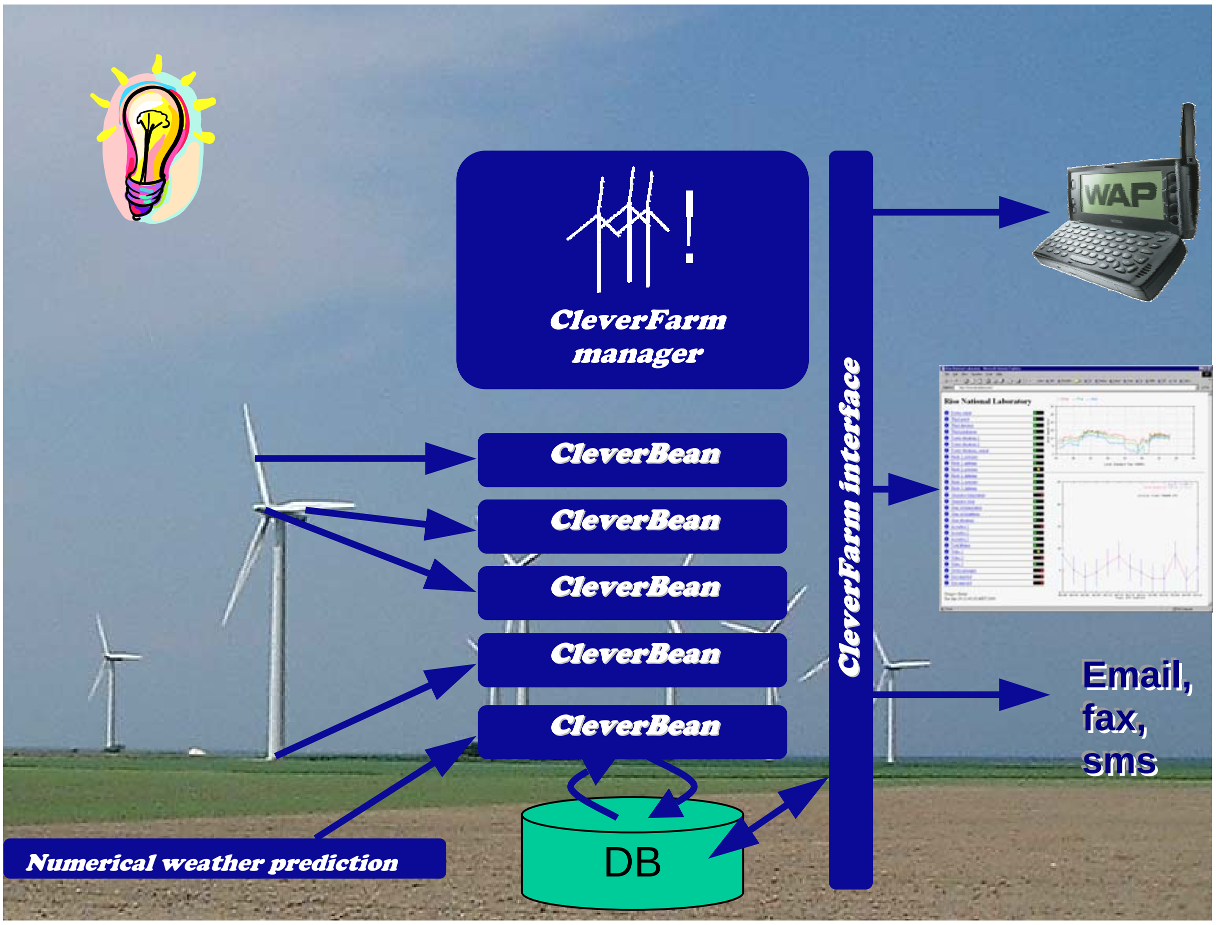


Figure 1: The idea of CleverFarm.

1 Introduction

1.1 The rationale

Wind turbine technology has ventured in recent years from prototypes and first deployments towards large power plant scale projects. With this, also the ownership structure of wind farms changed: from single farmers to co-operatives, and to large multi-national developers specialised in building and running wind power projects. At the same time, the best sites for wind energy were already taken, leading to more remote sites and offshore sites being developed. Both these developments lead to an increased wish for remote monitoring of turbines. Ideally, the turbine would know on its own accord when it would need maintenance, and call the maintenance crew autonomously. The crew then would have all the information they need to have before they go out to the turbine and do the necessary tasks. Having knowledge of the type of fault that has happened would help the maintenance crew to deal with it efficiently. This also could mean to wait until the next scheduled maintenance is due. The potential savings for this alone are considerable, if you think of the plans for offshore wind farms tens of kilometres from the coast, where access would probably be by helicopter (as we see already in the case of Horns Rev).

The idea behind this project was to take the existing techniques developed for optimising and enhancing the performance of wind farms, integrate them into one system and implement the system at a number of wind farms. The techniques include remote measuring of the status and production of the wind farm, short-term prediction of the expected wind speeds at and power output from the wind farm, models for wake calculations, remote control of wind farm production and so on.

1.2 State-of-the-art before CleverFarm

Anecdotal evidence from one major wind farm operator in Denmark speaks for itself: For staying in contact with the different wind farms, no less than 7 separate remote access systems were used. Not only were these not compatible with each other, in some cases they could not be run simultaneously, and one of the systems could not even be installed on the same machine together with the others. So to get a good overview of the current status, it required two computers, running 7 different and incompatible applications. CleverFarm instead intended from the start to be platform and vendor independent. It also never was thought as a replacement for the turbine's SCADA system, but rather as some kind of SuperSCADA, using the results of the SCADA system itself and correlating these with other monitoring approaches. After the start of the project, the IEC set up a task for the standardisation of wind turbine data (IEC 61400-25), which some of us are involved in. This common interface / definition of wind farm data should enable a more unified approach to wind farm surveillance. Also, in the last years the major turbine manufacturers have developed their own online monitoring platforms, which are accessible with a browser and therefore do not disturb other manufacturers' systems.

Similar developments have taken place in the field of short-term prediction of wind power. While at the beginning of the project only a handful of teams was offering wind power predictions based on Numerical Weather Prediction (NWP), with Risø being there since 1990, at the recent European Wind Power Conference in Madrid (2003) there were 35 papers in the field, including a few completely new models. See [1] for a thorough overview of the developments. Also, at the beginning of the project, only one group was working in the direction of integrating forecasting into a larger Energy Management System, while now there is a handful of suppliers.

Condition monitoring is state of the art in many kinds of industrial processes, *eg* power station turbines, pumps, electrical drives etc. Because of its promising perspectives to increase operational reliability and safety, condition monitoring is also a field of ongoing, very active research.

Modern type wind turbines are equipped with redundant safety systems and often provide remote supervision systems. These have contributed to a decrease in down time because the plant operator and the manufacturer can immediately be notified in case of malfunctions via remote data access. However, in contrast to a full fledged Condition Monitoring System (CMS), these systems only perform an action when a malfunction already has occurred and do not give much assistance in diagnosing the underlying faults.

Few years before the CleverFarm project was launched, first approaches to install CMS in wind turbines were made. One of the actors in this field was ISET. Within a five-year research project, funded by the German government and by an industrial partner (Brüel & Kjær Vibro GmbH, Darmstadt, Germany), a commercially available CMS, the VIBRO-IC system, was developed. As a first step towards wind turbine condition monitoring, algorithms and monitoring routines have been developed and were field tested at some Danish type wind turbines of the 600 kW class. In this field test, a basic set of monitoring functions for rotor asymmetries, degradation of rotor aerodynamics (*eg* icing), tower oscillations, power characteristics and generalized gearbox and bearing vibration levels were evaluated. This system is also used in the actual CleverFarm CMS hardware (see section 2.2 for the technical description).

The situation in Germany in the last year has triggered a drastic increase in maintenance demands from insurance companies. Many wind power developments were faced with the threat of non-continuation of their insurance contracts, if they would not replace the major rotating components after 5 years. This, of course, would make wind energy very expensive. In a series of talks with the insurance companies, developers and other stakeholders in the industry have managed to convince the insurers to also accept condition monitoring, but have put up some strict demands on these systems [2, 3]. This should help condition monitoring in general and the CleverFarm and OptiFarm systems greatly on the German market.

1.3 The CleverFarm project

On this background, a group of turbine operators, research institutions and CMS specialists got together to integrate their respective efforts in a unified system. This project was funded by the EU (ERK6-CT-1999-00006). During its course from April 2000 to October 2003, three turbines in two wind farms in Denmark and Germany were fully instrumented with vibration sensors on the rotating machinery, video cameras and the modularised software suite. The resulting software is now downloadable from www.CleverFarm.com.

This report gives an overview over the project, and of the results achieved.

2 The system – Hardware

The basic idea of the CleverFarm system was to use off-the-shelf components in an intelligent way to monitor the behaviour of wind farms. Therefore, we used the existing hardware of the partners to monitor the loads on the main components, plus to give extra information about the behaviour of the farm from joining that information.

2.1 Gram&Juhl vibration monitoring

The Gram&Juhl DAM system (Dynamic Analysis Module) obtains the scalability, high reliability and fixed channel price by distributed computing and integration of accelerometer, signal condition unit, computational unit and communication interface. The sensor communication interface is digital and can be daisy chained by only four wires, facilitating and minimizing cabling. The sensors are designed for rough environments, *eg* offshore, with respect to both electrical interference and corrosion providing a high reliability.

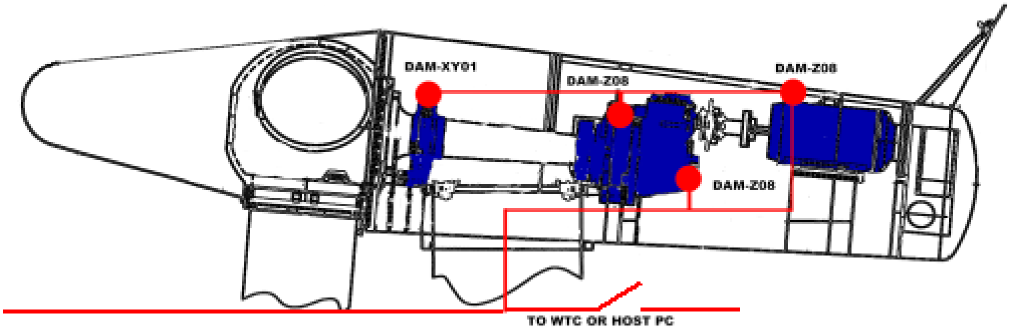


Figure 2: The sensor positions for the DAM sensors in Nøjsomhed.

The DAM sensors are capable of doing the following analyses which is mainly intended for Machine Condition Monitoring: timeseries, autospectrum, zoomed autospectrum, configurable overall, envelope time domain, envelope frequency domain and RPM. The following analyses will be added in the near future: CPB, kurtosis and order vectors. For Structural Vibration Monitoring (SVM) a very narrowband low frequency analysis with minimum time delay is implemented. This analysis is combined with real-time alarm signaling through a relay to the Wind Turbine Controller WTC. Four independent frequencies on either X, Y or XY direction(s) can be surveyed in the same sensor. On top of this, motion analysis can be made to monitor harmonic movements, *ie* their direction and amplitude at a given frequency, *eg* tower resonate frequency and edged and flap oscillations. A selected frequency guard can output its level on the analog output. All sensors have inbuilt interface for RPM or external TRIGGER making them well prepared for handling wind turbines with variable speed and synchronizing analysis among sensors. The functionality of the DAM sensors is to a very high degree based on software, which remotely can be updated, so upgrading in the future, where third level estimation techniques is expected to be included in the software, is made easy.

2.1.1 System Components

The DAM System consists of these main components:

Turbine Computer (IPC)

The Turbine computer interfaces directly to the Wind Turbine Controller (WTC) and the DAM system via custom serial-line protocols. A Monitoring system (WPS) is running on the Industrial PC (IPC) and provides information about the state of the Turbine. The DAM WEB application running on the IPC receives information from two data providers (DP): The DAM DP forwards vibration-monitoring data from the intelligent sensors and the WPS DP forwards Turbine data.

The main tasks of the DAM WEB are:

1. Merge data from data providers (conditional recording, see below)
2. Evaluate trend alarms
3. Fire monitoring alarms for the Park Computer
4. Provide information for the CleverFarm central database beans

The main tasks of the DAM DP are:

1. Configure the DAM sensors for the required measurements
2. Receive vibration measurements and native alarms from the DAM sensors and forward them to the DAM WEB
3. Monitor the DAM sensor network

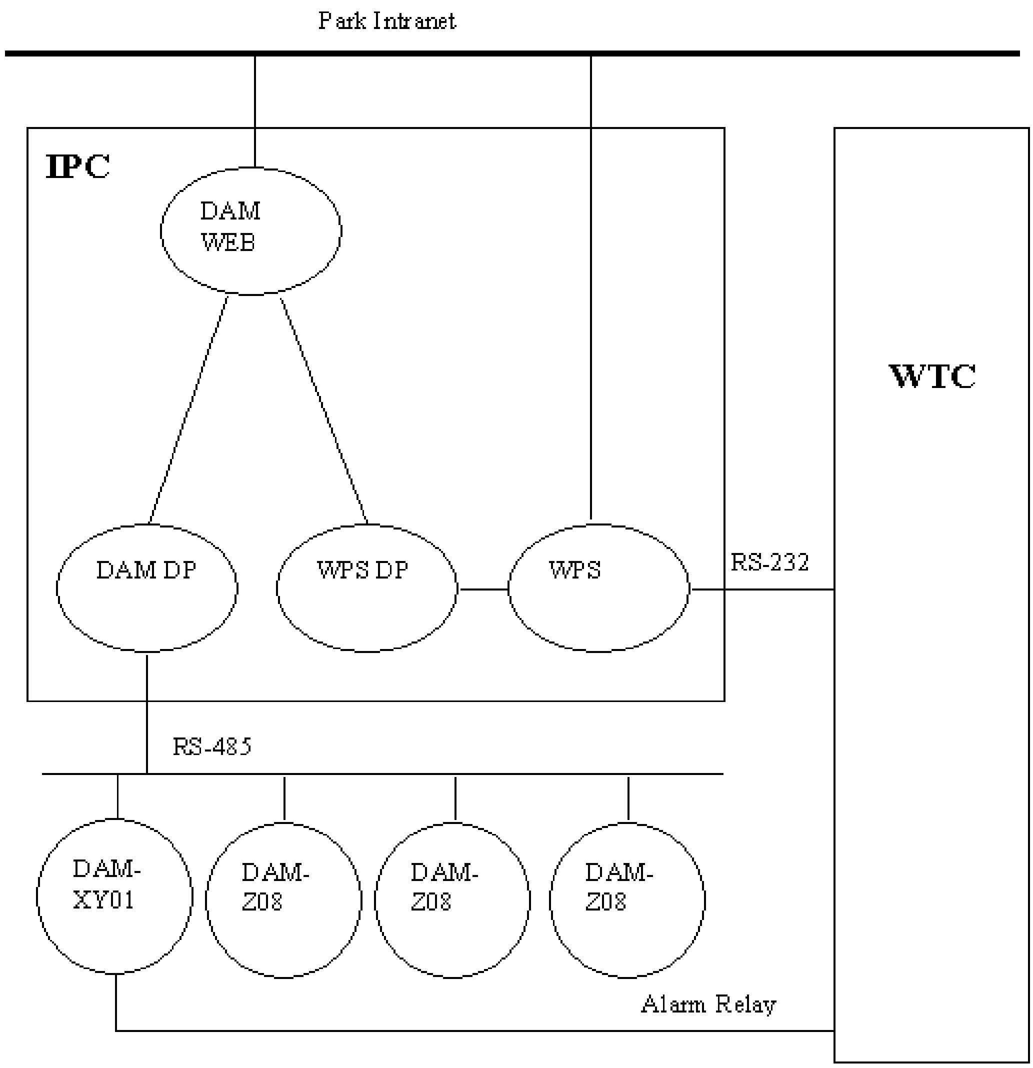


Figure 3: Layout and connections of the Gram&Juhl DAM system.

Conditional recording

All data from the data providers (DAM DP, WPS DP) are merged according to a set of configurable rules:

1. Measurements are tagged with turbine state parameters (*eg* ActivePower).
2. Data are "binned" in a matrix. The matrix can have one or more dimensions. Typically, it has just one dimension, ActivePower. Only measurements that fall within an open bin are stored on the Park Computer, following which the bin is closed for a specified time interval.
3. A bin is an interval of ActivePower with stability conditions applied.

DAM Sensors

A total of four sensors are installed:

1. One dual-axis sensor on the main bearing for monitoring structural movements and the state of the main bearing (DAM-XY01)
2. One single-axis sensor on the low-speed stage of the gear
3. One single-axis sensor on the high-speed stage of the gear
4. One single-axis sensor on the generator

The DAM Sensors perform the following tasks:

1. Run a schedule of vibration measurements (FFT, Overall, Envelope…) . Each sensor can run up to 32 measurements in the schedule.
2. Transmit measurement results to the DAM DP
3. Continuously monitor structural motion (main bearing only)

2.1.2 Technical Specifications

IPC

The industrial computer installed is a PIP6 manufactured by MPL AG (www.mpl.ch). It operates fanlessly. It runs the Windows 2000 Operating System with a CPU speed at 266MHz.

Sensors

Details about the sensors can be found at Gram & Juhl's homepage, www.gramjuhl.dk.

A few specifications are stated here:

DAM-Z08 Sensor 	Dynamic range >70dB Calibrated at 159.2 Hz Frequency accuracy 30ppm Coupling AC -3dB 1.5 Hz Full Scale Range typ. ±65 m/s² Nonlinearity 0.2% of Full Scale Transverse sensivity ±2% Sensivity temprature drift 0.5% Overload detection Yes Noise 225 µG/$\sqrt{Hz}$ • Configurable frequency span: 1Hz -8 kHz • Autospectrum with Zoom and Envelope. 400 lines within span. • Cepstrum non liftered • Averaging on all measurements except time • Time series 32760 samples at configurable sampling rate • RPM Tracking through digital re-sampling
DAM-XY01 Sensor 	Dynamic range > 70dB Calibrated at 10 Hz Frequency accuracy 30 ppm Coupling DC or AC (digital filter) Full Scale Range ±180 m/s² Nonlinearity 0.2% of Full Scale Transverse sensivity ±2% Overload detection Yes Sensivity temprature drift 0.5% Noise 1mG/$\sqrt{Hz}$ **Vibration guards** • No. of independent guards 4 • Sensing direction X, Y or XY • Centre Frequency]0 Hz .. 100 Hz] • Bandwidth >2% • Filter Response Minimum Phase • Averaging Exp. running mean **SVM and MCM measurements** • Motion pattern (Major axis, minor axis, orientation) • Averaging on all measurements except time • Configurable frequency span DC–1 kHz • Autospectrum with Zoom and Envelope. 400 lines within span • Configurable Overall. • Cepstrum non-liftered • Time series 16384 samples at configurable sampling rate

2.2 ISET

2.2.1 Description of the VIBRO-IC hardware installation

Main unit hardware

The VIBRO-IC base unit with integrated embedded PC is mounted on a rack with a dimension of 800 x 400 x 450 mm (height x width x depth). It is placed between the main switchboard and the switchboard for the network equipment on the platform in the tower base of the turbines 7 and 8 in the Nøjsomheds Odde wind farm (see Figure 4).

VIBRO-IC needs some measurements out of the turbines control system. The required digital signals and the analogue power signal have to be connected to a CAN-I/O-Module. This module has a dimension of approx. 140 x 160 x 55 mm (height x width x depth) and is mounted below the VIBRO-IC rack. The turbine manufacturer has built in a connection box in the tower base, from which the signals can be obtained with galvanic separation. This is done by using opto coupling devices for the digital signals and transformers for voltage and current measurement.

Nacelle sensor equipment

For the measurement of nacelle oscillation and vibration on housings of gearbox and generator some additional sensor equipment has to be installed. Measurement of the transverse, axial and torsional oscillation of the nacelle requires three acceleration sensors (accelerators). One in axial direction (based on rotor shaft direction), one in transverse direction in front of the vertical tower axis and one transverse rear the vertical tower axis.

The mounting positions for the axial and transverse front accelerators are shown in Figure 5 left. The visible structure is the main rotor bearing. This allows a combined use of the front acceleration sensors also for monitoring this bearing. The rear accelerator is mounted at the supporting structure of the nacelle, about 4 m away from the vertical tower axis (Figure 5 right).

Figure 4: VIBRO-IC installation in the tower base of turbine 7.

The vibration sensor on the gearbox is mounted near to one of the existing measurement points (Figure 6 left). The generator's vibration sensor is mounted with a glue socket on one of the generator bases as shown in Figure 6 right.

Figure 5: Acceleration sensors on the supporting structure of Turbine 7.

Figure 6: Vibration sensors on Gearbox and Generator

Since a signal for the absolute rotor position (1 pulse per rotor revolution) could not be obtained from the turbine's control system, an additional inductive distance sensor had to be installed behind the protective cover of the hub/rotor shaft. As shown in Figure 7 the sensor faces the hub bolts. With a small metal marker glued on one of the bolts, the sensor generates the required 1 pulse per rotor revolution signal.

Figure 7: Inductive distance sensor for rotor position signal.

There are some more pulse signals derived from the turbine's control system in the nacelle. The manufacturer has provided these signals in a separate box with opto coupling devices in it. All the pulse signal and sensor cables in the nacelle are brought together in a connection box (see Figure 8), which is also placed in the nacelle. From this box, a 6x4 core cable leads to the VIBRO-IC and peripherals in the tower base.

Figure 8: Connection box in the nacelle of Turbine 7.

2.2.2 VIBRO-IC monitoring functions

The VIBRO-IC systems monitor the actual state of a WEC. Several characteristic values will be calculated from time series measurements. These characteristic values will be investigated due to fault patterns to predict faults in WEC components before major breakdowns and consequential damages occur.

Values measured by the VIBRO-ICs are: the power characteristic, nacelle oscillations and vibrations on gearbox/bearings. The power characteristic of the WEC will give some overall information of the degradation of performance, *eg* caused by increased surface roughness of rotor blades or problems with the yaw system. Power will be used for classification of other characteristic values.

The following **Table 2.2.1** gives an overview over the direct measurements performed by VIBRO-IC. The measured values can be divided into three types: "discrete" can have only a determined set of values (*eg* [0, 1]). "Scalar" is a scalar floating point value, "complex" is a complex value with amplitude and phase. These measurements are describing the actual condition of the WEC and therefore are called "characteristic values".

Pos.	**.1.1 Measurement**	**Updrate**	**Value [Dimension]**
1	Active power of WEC	1 s	scalar [kW]
2	Wind speed	1 s	scalar [m/s]
3	Generator grid coupling status (off/stage1/stage2)	1 s	discrete [-] (off:0, stage1:1, stage2:2)
4	BCU generator bearing	60 s	scalar [BCU]
5	BCU gearbox (optional)	60 s	scalar [BCU]
6	Peak value generator vib. signal	60 s	scalar [g]
7	Peak value gearbox vib. signal	60 s	scalar [g]
8	RMS value generator vib. signal	60 s	scalar [g]
9	RMS value gearbox vib. signal	60 s	scalar [g]
10	Rotational speed rotor	10 s	scalar [1/s]
11	Rotational speed generator	1 s	scalar [1/s]
12	Transverse (due to rotor shaft) nacelle acceleration at rotor frequency (1p-freq.), front mounted sensor	120 s	complex [g] and [degree]
13	1p-transverse nacelle acceleration, rear mounted sensor	120 s	complex [g] and [degree]
14	1p-axial nacelle acceleration	120 s	complex [g] and [degree]
15	2p-rotor shaft oscillation, front mounted sensor	120 s	complex [g] and [degree]
16	3p-transverse nacelle acceleration, front mounted sensor	120 s	complex [g] and [degree]
17	3p-transverse nacelle acceleration, rear mounted sensor	120 s	complex [g] and [degree]
18	Acceleration amplitude near 1^{st} tower bending mode Eigen frequency f_{E1}	120 s	scalar [g]
19	Amplitude near $f_{E1} + \Delta f$	120 s	scalar [g]
20	Amplitude near $f_{E1} - \Delta f$	120 s	scalar [g]

Table 2.2.1: Direct measurements VIBRO-IC.

From the above mentioned measurements additional characteristic values are calculated. These values are listed in **Table 2.2.2** in the same way as in **Table 2.2.1**. The statistical deviation of turbulence intensity of wind speed (pos. 25) is calculated in 50 classes.

Pos.	.1.2 Measurement	Updrate	Value [Dimension]
21	5-minute mean value of power of WEC	300 s	scalar [kW]
22	5-minute mean value of wind speed	300 s	scalar [m/s]
23	Power characteristic of WEC (power over wind speed with standard deviation of power	300 s	2 scalar values [kW] in 40 classes (start value 3 m/s, class width 0.5 m/s)
24	Turbulence intensity of power	300 s	scalar [1]
25	Turbulence intensity of wind speed	300 s	scalar [1]
26	Statistical deviation of turbulence intensity of wind speed	300 s	scalar [1] in 50 classes
27	CREST (Peak/RMS) generator vibration signal	60 s	scalar [1]
28	CREST gearbox vibration signal	60 s	scalar [1]
29	Rotational speed rotor	10 s	scalar [1/s]
30	Resulting 1p-transverse nacelle acceleration (complex mean value of 1p-transv. front and rear)	120 s	complex [g] and [degree]
31	Resulting 1p-torsional nacelle acceleration (complex differential value of 1p-transv. front and rear)	120 s	complex [g] and [degree]
32	Frequency f_{E1} of 1st tower Eigen bending mode	120 s	scalar [Hz]
33	3p-torsional nacelle acceleration (complex differential value of 3p-transv. front and rear)	120 s	complex [g] and [degree]

Table 2.2.2: Calculated characteristic values

The characteristic values for oscillation and vibration monitoring are derived from two different types of sensors (refer also to section 2.2.1). The nacelle oscillation induced by rotational speed of the rotor at very low frequency are measured with static accelerometers, *ie* with a lower cut-off frequency of 0 Hz. Figure 9 shows the schematic mounting positions of the sensors at labels 2, 3 and 4.

Labels 5 and 6 in Figure 9 show the position of the gearbox and generator vibration sensors. These sensors operate at a frequency range from 1 Hz to 20,000 Hz to measure the vibration induced by bearings and gearwheels. The absolute rotor position, measured by an inductive distance sensor (label 1 in Figure 9, picture in Figure 7) is needed to perform phase sensitive narrow band analysis, a specialized FFT-algorithm for signal processing.

The VIBRO-ICs had to be integrated into the wind farm’s network structure, which is based in Ethernet with TCP/IP. To supply VIBRO-IC with networking and Internet capabilities, Embedded PC (EPC) technology is used. Figure 10 shows the chosen configuration, where a VIBRO-IC with its serial RS232 connection is combined with an EPC. This EPC cyclically reads out the relevant fault prediction data from the VIBRO-IC. The EPC has an integrated web server and is application programmable (C++ on WindowsCE). With this configuration, all relevant actions, like software updates or parameter refinement, can be done via Internet tools, *eg* web browsers. Data download can be done via FTP. Security of data transfer over the Internet can be achieved using Virtual Private Network (VPN) technology.

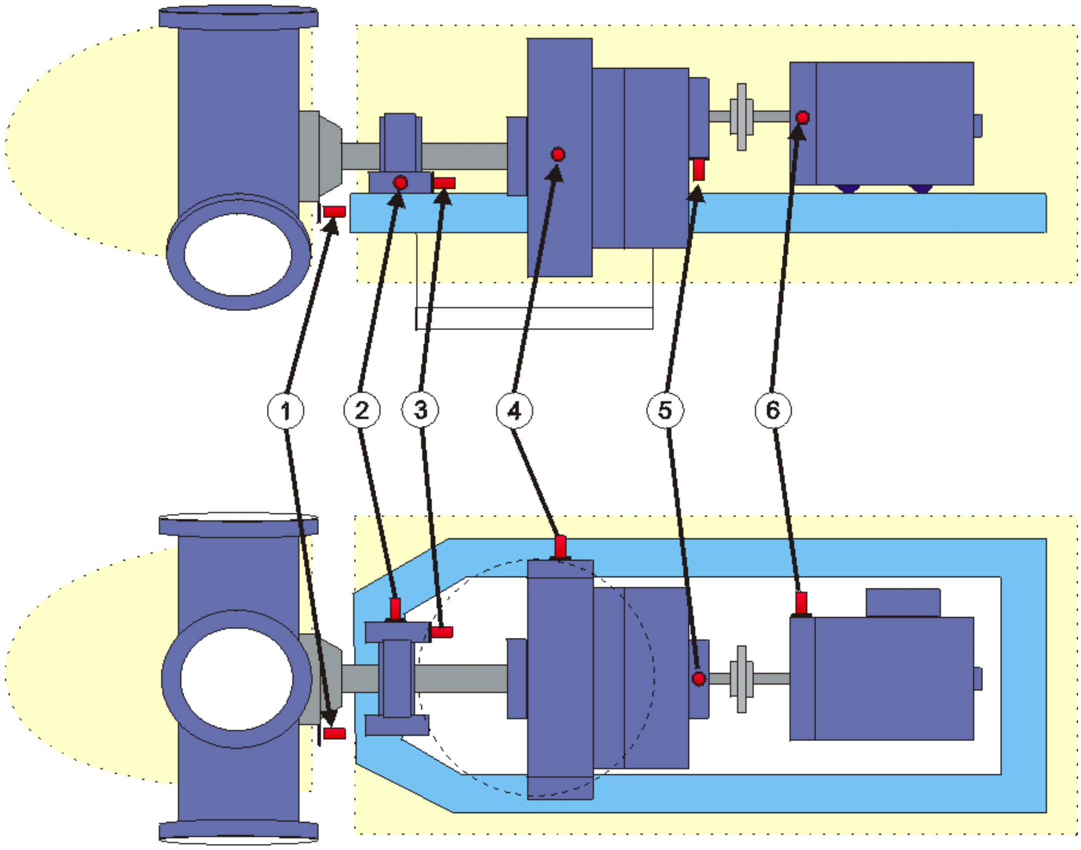


Figure 9: Condition monitoring sensor configuration for the VIBRO-ICs in Nøjsomheds Odde.

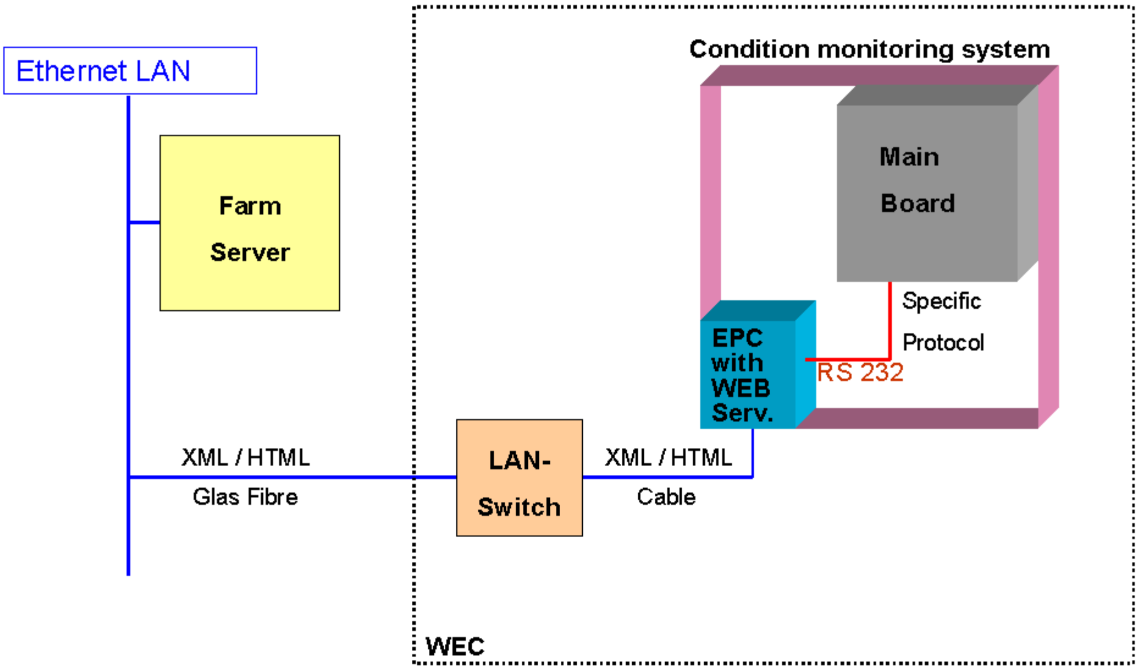


Figure 10: VIBRO-IC connected to Embedded PC and Ethernet LAN.

2.3 Overspeed Remote Video System

The goal of the remote video system is a visual access to the farm for different purposes. In one simple view, the user can check for external damage (*eg* offshore from ships or after lightning strikes), weather conditions (important for the accessibility of an offshore wind farm), a status of the other turbines in the farm, and a redundancy check for some of the other measurements. A frequent indoor application is the surveillance of measuring instruments and slowly developing problems and failures like oil leakages in the nacelle. Of course, serving as a web camera, one purpose also is public relations.

In our test farm Nøjsomhed, an industrial grade video camera is mounted on the nacelle. It is supported with a pan/tilt mounting which is able to work under the harsh conditions on top of a running wind turbine. The camera itself has an optical zoom, and manual and autofocus. It enables a quick overview of the environmental conditions at the wind farm [4].

On the video computer (in the bottom of the turbine), a server is running through which the camera is controlled via RS 485. The remote access to this computer is via TCP/IP, so the whole functionality can be used via Internet. Offline the camera server can be programmed such that it takes pictures from set positions (“markers”) at a given time interval. These pictures are stored together with time stamp in a DB and accessed through an interface from CleverFarm. The user interface for the camera has password protected controls, but the image itself could be delivered without password check.

The pan/tilt/zoom facilities are controlled via an internet browser running a Java plug-in (see Figure 22).

Figure 11: Example of a panorama view from the nacelle in Nøjsomheds wind farm.

Since the installed camera has pan-tilt-zoom facilities, a software-driven panorama could be done at predefined times of the day. Since the amount of data being streamed from a camera is clearly too much for basically any database, these panoramas (created from a complete circle of the camera, feeding a software like Quicktime VR) should be stored. The same thing could be done for the blades, turning every blade into the viewport of the camera and then turning the blade while going up and down with the camera. However, since this would mean to stop the turbine for the duration (~10 min), it might not be done frequently, or only under low-wind conditions.

An additional webcam is installed inside the nacelle, to check for obvious problems. Similarly, cameras using other wavelengths could be advantageous – *eg* infrared for night-time surveillance.

Through a video multiplexer, up to eight indoor cameras could be installed. So also ad-hoc use of video surveillance is possible, *eg* for maintenance personnel.

Figure 21 shows an overview of the software systems used for the video surveillance. The software component on the client side is the VideoControl Java applet, which enables the users to control the camera (zoom and focus) and the pan-tilt head from inside their Web-Browser (see Figure 22).

On the video computer installed in the turbine a server application manages the connections of different applet instances and passes the commands received from them to the different cameras and pan-tilt heads via the serial interface and sends status information about the camera and the head back to the applets. This application also performs user authentication and manages user rights that can be configured for different access categories.

The images from the camera are grabbed by a software component (marked Image Grabber) and saved in JPEG files. The VideoControl applet reads these images via HTTP from an Apache web server installed on the video computer. A scheduling application can be used to take pictures of certain views in regular intervals. This application is a separate software component, which is not shown in the Figure.

The pan-tilt head used in the installation includes a custom-built controller circuit that is connected to the video computer using a RS485 serial bus. The remote-controllable video camera is connected to this controller using a RS232 connection. The video signal is transmitted to a video card installed in the video computer as a composite signal.

Figure 12: Detail view from the video camera in Grüppenbühren.

2.4 Central server

The main aim of the central server is to provide data access to online and historical data. It provides the database with all acquired data together with interface and access routines. From here, current and historical data is distributed again to the different services like user interface, input/output routines and of course the clever algorithms. The connectivity to the outside world is also a part of the server, though we have not put much emphasis on this aspect.

To this aim, a PC was equipped with two large (80 GB) hard disks, which were collected in a software RAID array (Redundant Array of Inexpensive Disks), which provided for data mirroring, thereby making the data safe even if one disk would die in the process.

The server was connected with Ethernet to the central farm network. While an ISDN connection would have been possible with a modem in the farm server itself (for backup purposes), we chose to use an ELSA ISDN gateway appliance. This proved to be beneficial in the case that the Bonus-supplied VPN connection would fail, which it occasionally did.

3 The system – Software

3.1 The central server platform

In the initial brainstorm session, we found the following demands on the system:

- High degree of modularity
- If possible, plug&play
- Use of open standards
- Ease of configuration
- Generic solution for module configuration
- Generic interfaces for modules/data
- Platform independence
- Modern software architecture
- Remote software updates should be possible
- Many levels of access and security
- Self-starting and stable

For a number of reasons, Linux was chosen as operating system for the central server platform. These included the good security against viruses, the high stability (especially important at a remote site), and the price/licensing conditions. As the project went, another advantage of Linux was very beneficial: the good possibilities of remote access. Practically everything on Linux can be administered remotely via Telnet or ssh (Secure Shell), thereby not having the overhead of a GUI over (potentially) just an ISDN connection. One possibility Linux is offering is currently employed by OptiFarm, which is based on the Knoppix distribution: The security of this particular installation is increased since it is run from a CD-ROM, which means that the basic OS is written once-and-for-all with all its options to CD and can then be used without the possibility of being altered. This defies some of the remote administration possibilities and should only be applied once the initial problems are ironed out. The Linux system used at Nøjsomhed is SUSE 7.1, since Risø (who set up the central server) already had experience with it.

For the database, one important issue was the price and licensing of the DB. The fees for the DB should not be very high in comparison to potential licensing proceeds from CleverFarm installations. At the same time, a well-tested standard solution was preferred, which could run on Windows and Linux. This was found in MySQL (*mysql.com*), an open-source DB distributed by a Swedish company. For testing and other non-commercial purposes, the DB is given away for free, but commercial applications would have a moderate licensing fee. At the same time, MySQL is very widely distributed in many projects, making it a well-audited software with many people delivering bug-fixes. A number of good user interfaces for administration have been developed lately, such as phpMyAdmin for remote administration with a web-browser.

We wanted to write a system that was platform-independent. This meant either to use scripting languages, with the inherent disadvantages of speed and the potential for diverse environments, or to use Java. We chose to use the Java programming language and runtime environment for the CleverFarm system, since some partners already had good experience with it, and since a whole wealth of free software libraries was available for it. The good built-in networking capabilities were another plus for Java. At the beginning, Java 1.3 was used, but since the advent of Java 1.4, we were using this. The implementation was used from Sun, since it was the reference implementation and available for Windows and Linux. Currently, version 1.4.2 is installed in Nøjsomhed.

The choice of user interface fell on server-generated graphs and HTML pages. In the Java world, this is done using Java Server Pages (jsp) technology. Jsp pages are written in code that is a mixture of HTML and Java code, and compiled to proper Java classes at the first call of the page. The compiled classes run after that with the full speed of the Java environment, and *eg* generate views into the database on demand. The standard implementation of a jsp server is Tomcat (*jakarta.apache.org*), which is an open-source software developed by the Apache collaboration. Apache is the most-widely used web-server in the world, running on about half of all the systems serving web-pages, including the servers employed in this project. Half-way through the project, we switched from Tomcat 3 to Tomcat 4, but the change was transparent to the writers of the jsp pages.

The wind farm in Nøjsomheds Odde has an internal Ethernet network. Therefore, the “plumbing” issues of networking were not an issue for CleverFarm. Actually, we decided early in the project that the actual connection of the network was of no concern to us, since this was outside of the scope of the project. Nowadays, most new wind farms are built with an internal Ethernet network anyway, so this is not a problem any more. However, remote access to the internal farm network is a different matter. In the case of Grüppenbühren, a fixed DSL connection could be laid down. This only left the problem of changing IP-adresses for the farm, which could be overcome using a dynamic DNS (Domain Name Server) approach. In the internet, there is a number of services offering dynamic DNS services for free, where a little client application publishes its IP number to a central server, which in turn can redirect the DNS look-up to a sub-domain of its own domain. OptiFarm in Grüppenbühren *eg* is available as http://grueppenbuehren.dnsalias.net/, although its IP number is frequently changing. Another solution is used in Nøjsomhed: Here, the wind farm is integrated in SEAS’ own internal network. From the outside, secure access is possible via VPN (Virtual Private Network), which is an industry standard for the secure and encrypted tunnelling of private connections over the (normally unsafe) internet.

3.2 The CleverFarm manager

The structure of the CleverFarm system allows Java Beans, essentially modules of code which perform whatever the bean author wants, to be plugged into the system as required. As the CleverFarm manager is simply loading the beans, there are no restrictions on what these beans can do. The input and output of the beans can be from/to XML or a MySQL database which in turn can be viewed by the user using jsp pages with a standard browser. This allows third parties to incorporate data collection and analysis code into the system, without the need to adhere to a strict CleverFarm specification.

The CleverFarm manager handles the execution of these beans, based on a run list contained in the CleverData DB table BeanRegister. An entry is manually created in the BeanRegister for each bean, containing the following parameters:

BeanName	The name of the Bean
MethodName	The method (function or subroutine) to be called when activating the bean
RunParams	Parameters to pass to the beans method
RunFrequency	How often the bean should run

Table 3.2.1: The main bean parameters in the BeanRegister.

When the manager is initialised, the values from the BeanRegister are copied to the ProcessQueue table, with repeating entries for periodic processes. The ProcessQueue then contains the execution list for a one-day period following the launch of CleverManager. The CleverManager is itself re-launched every day.

The CleverManager cycles through the process list, waiting for the startTime of each bean, then executing the bean as a new thread. The time the bean finishes is logged in the database.

Using it:

There are no direct user interfaces with the CleverManager, other than the entries in the BeanRegister and the process queue (*ie* output such as end time). The current status of the CleverManager can be determined by looking at the ProcessQueue table. Both these tables are available for viewing through the CleverFarm dynamic web pages.

Improvements:

Ability to upload beans, modify their properties and launch them from web pages which interface with the BeanRegister and ProcessQueue table.

3.3 The CleverBeans

The general behaviour of CleverBeans is that they are small pieces of code, which can be self-contained, and which comprises a method to be called by the CleverFarm manager. For ease of writing, a template CleverBean exists, which can be extended. We also have drafted a guide to CleverBean programming, to be distributed to third parties.

A number of open-source libraries were used in the architecture of the beans: log4j, JFreeChart and Cewolf, the ftp library by Bruce Blackshaw and a few more.

3.3.1 The Prediktor Bean

Zephyr/Prediktor

The prediction of wind speeds and power 48 hours in advance is done by Risø's Zephyr/Prediktor software. This was implemented as a CleverBean. It downloads four times a day the newest predictions of DMI's HIRLAM model, refines them to get to local predictions and puts the predictions into the database. From there, they get called via a custom jsp page, which outputs the data in tabular and graphic form.

The advantage of having a prediction model in the farm controller is twofold: the value of the produced electricity can be increased in regions where false or no predictions cost connection charges, and the maintenance of the turbine can be planned better than without the system.

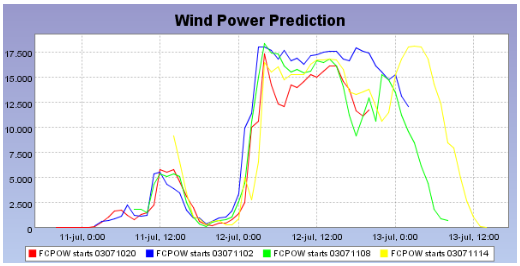


Figure 13: Four consecutive predictions for the Nøjsomheds Odde wind farm.

The inner workings of Prediktor have been described a number of times elsewhere [5]. On a more general level, short-term forecasting is described in detail in a document written for the EU Anemos project [1]. Summarising, Prediktor works by using Numerical Weather Prediction (NWP) output from a meteorological institute (here the Danish Meteorological Institute DMI), refines the 10-m wind speed and direction from the simple interpolation to a wind in hub height of the site using the equations from WAsP (www.wasp.dk), uses the power curve to convert the wind speed to power, and finally takes out the park and wake effects using the PARK program.

DMI-HIRLAM

The operational weather prediction system at DMI is called DMI-HIRLAM. The goal of DMI-HIRLAM is to provide high accuracy meteorological forecast products, with a special priority on forecasts valid for the short range, up to two days ahead. The system provides guidance to meteorological staff and to numerous customers in general. Furthermore, the results are used as input to specialised forecast models, *eg* storm surge model, road conditions model, ozone forecasting system, and atmospheric dispersion models.

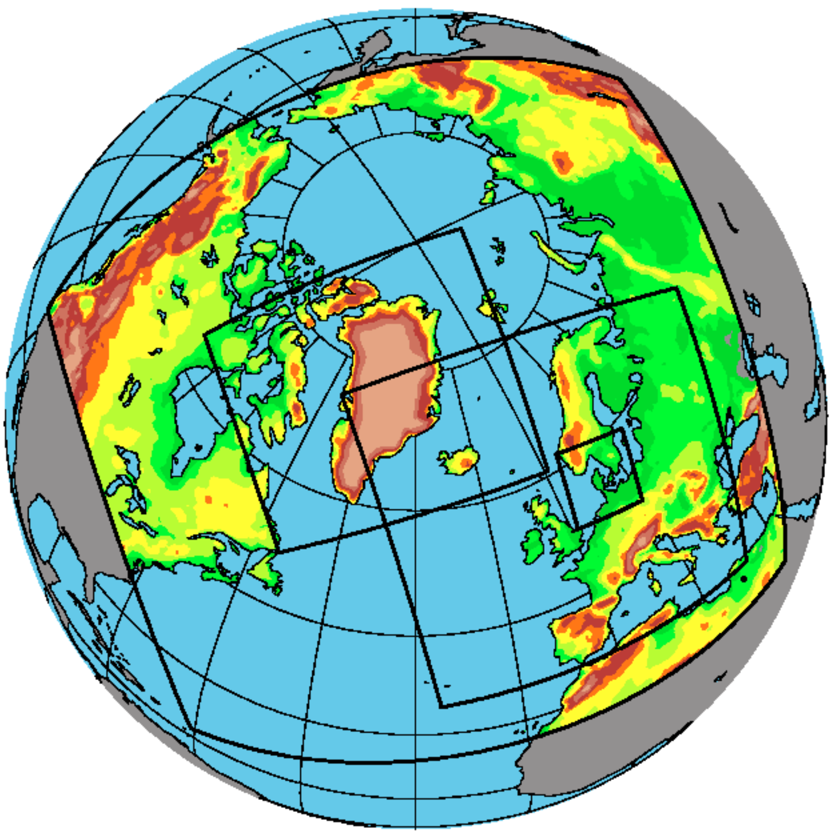


Figure 14: The operational setup of the HIRLAM model.

Meteorological data are received through modern communication systems, processed and assimilated into the weather prediction model DMI-HIRLAM which describes the dynamical and physical processes of the atmosphere.

The forecast model is a grid point model. This means that the numerical solution of the governing equations is carried out in a system of grid points. The model dynamics solves the equations of motion for the atmosphere using the hydrostatic approximation. The diabatic processes are described by the physical parameterizations covering the processes of radiation, condensation, convection and turbulence. A surface scheme describes the time dependent fluxes of heat, moisture and momentum at the lower boundary of the atmosphere. The computer power needed to run the operational forecasting system is supplied by a powerful NEC-SX6 super computer configuration.

The DMI-HIRLAM forecasting system consists of several nested models each covering a limited area of interest (see Figure 14). The European Centre for Medium Range Weather Forecasts (ECMWF) global model fields provide the lateral boundary values for a coarse mesh model "G" which is run with a horizontal resolution of 0.45°. The G-model supplies the boundary values necessary to run a higher resolution model "E" for Europe and a model "N" covering Greenland. These models running at a resolution of 0.15° are providing more detailed forecasts. A high resolution model "D" (0.05°) providing additional details is run for a domestic area around Denmark. The DMI-HIRLAM models are run with 40 vertical model levels. Each model makes its own analysis every 3 hours in order to create an adequate initial state for that model. The nesting strategy including analysis cycles for each model is described in DMI Technical Report no 02-05.

A verification of the DMI-HIRLAM forecasts is performed regularly for the meteorological parameters pressure, temperature, humidity, wind etc. Figure 15 shows the evolution of the average hit rate of the 24 hour 10m wind speed forecasts (difference smaller than 2 m/s) at Danish stations.

This figure demonstrates a continuous improvement of wind the forecasts over the recent years. Another comparison between observed and modeled Weibull distributions at coastal areas is shown in Figure 16. It indicates the role of the horizontal resolution in the DMI-HIRLAM model. The DMI-HIRLAM D model, which has a higher resolution than the DMI-HIRLAM E model, represents the observed Weibull distribution slightly better than DMI-HIRLAM E.

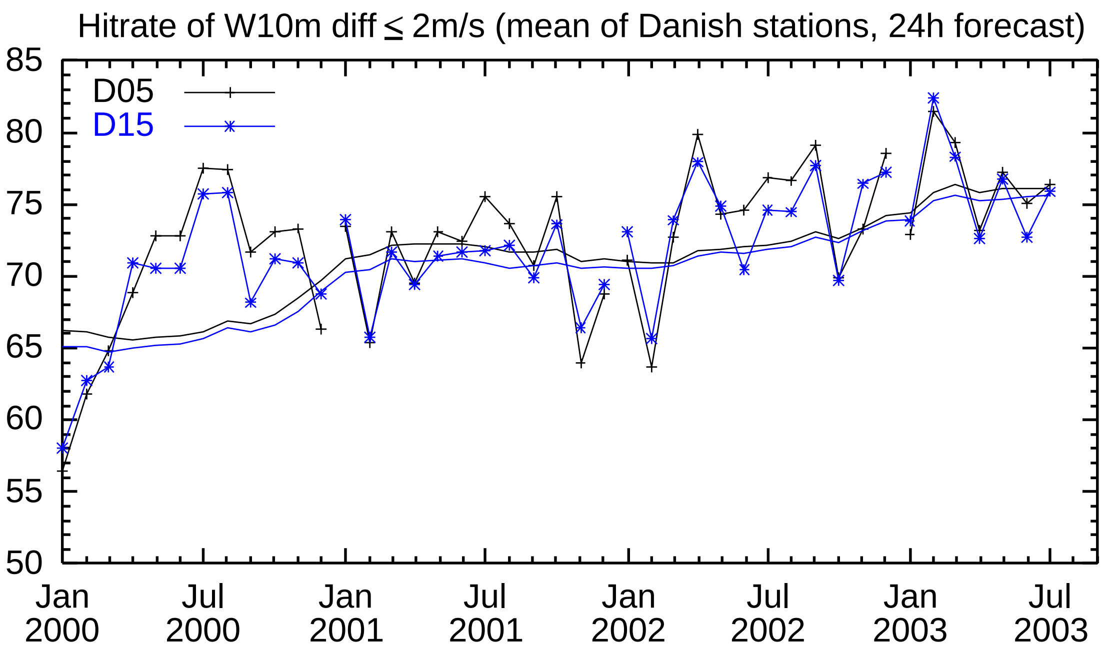


Figure 15: A verification for the DMI Hirlam system. Evolution of the hit rate of 24 hour forecasts for 10m wind speed over Danish stations.

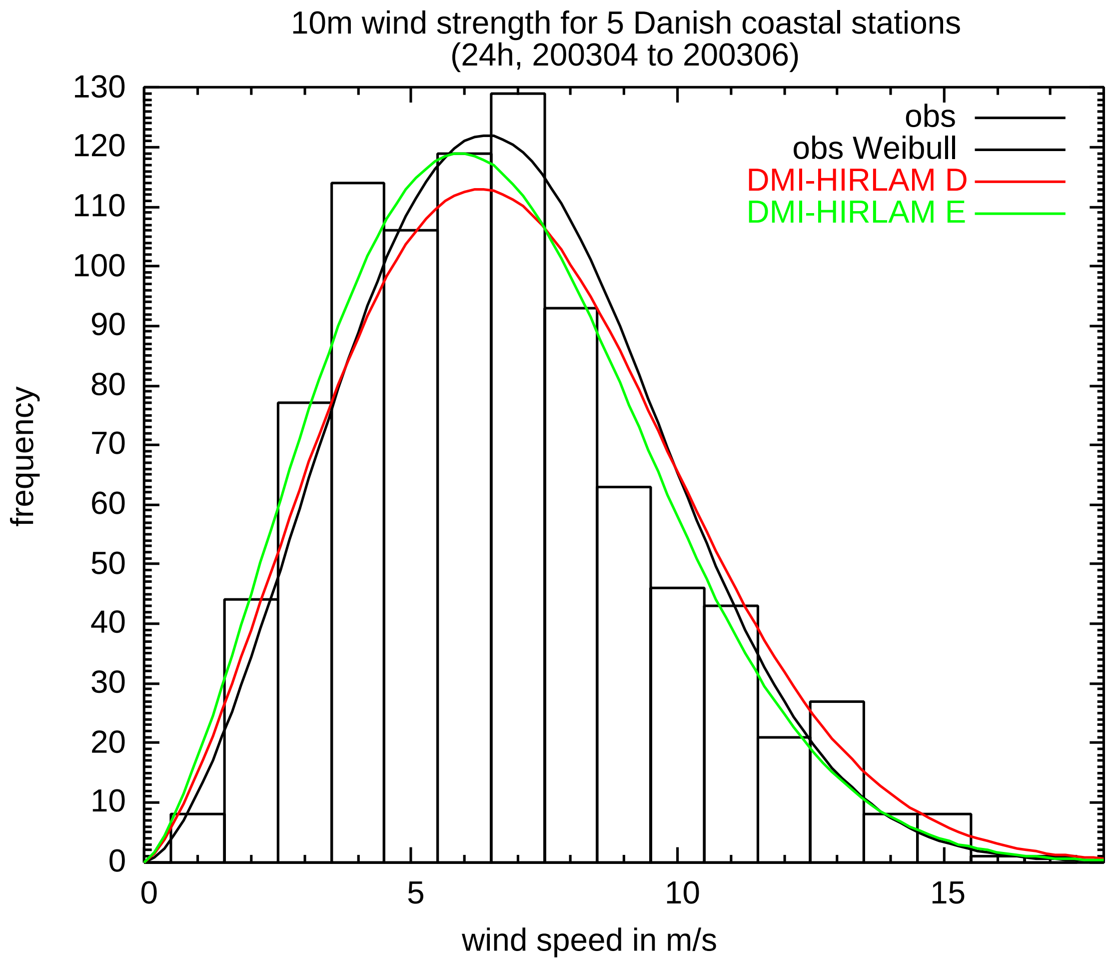


Figure 16: Observed frequency distribution of the wind speed and the respective Weibull distribution from 5 Danish coastal stations versus the Weibull distributions for the 24 hour forecasts from DMI-HIRLAM E and DMI-HIRLAM D.

3.3.2 Vic2DB

The Vic2DB bean collects the data from the installed VIBRO-IC systems in turbine 7 and turbine 8. The data are stored as XML files on the web server data area of the Embedded PCs. The XML files are downloaded via an URL stream by the farm server. The data are parsed by specialised functions which come with the Java developer kit used for the programming work. As described in Table 2.2.1 and Table 2.2.2 resp. (see section 2.2 of this report) the VIBRO-IC system provides several direct and calculated measurement values with different update rates. To handle the data transfer with no extensive effort for collecting data with different update rates, the VIBRO-IC data are grouped. There are two groups of data, one with a download rate of 100 seconds, the other with a 300 second download rate. These two groups cover all the different update rates given in Table 2.2.1/2.2.2 with an appropriate sampling rate. The following Table 3.3.2.1 shows the allocation of the measured signals for both data groups.

The meaning of these measured values and some data evaluation and processing examples will be presented in section 5.2 of this report.

The downloaded data is stored in a database on the central server (see section 2.4). The database works with the SQL standard and can be accessed via Internet protocol driven parser functions, for example by JSP scripts. From this database, all data access for evaluation and visualisation is done.

download rate 100 sec.	download rate 300 sec.
Current Active Power	5-Min mean value Power
Current Generator status	5-Min mean value Wind Speed
BCU Value Generator	Turbulence intensity Power
BCU Value Gearbox	Turbulence intensity Wind Speed
Crest Value Generator	Alarm Power characteristic P(Vw)
Crest Value Gearbox	Alarm BCU Generator bearing
Amplitude 1p transverse front	Alarm BCU Gearbox bearing
Phase 1p transverse front	Alarm Crest Generator bearing
Amplitude 1p transverse rear	Alarm Crest Gearbox bearing
Phase 1p transverse rear	Alarm Mass imbalance of rotor
Amplitude 1p axial	Alarm Aerodynamic imbalance
Phase 1p axial	Alarm 2p Transverse oscillation
Amplitude 2p transverse front	Alarm 3p Torsion oscillation
Phase 2p transverse front	Alarm Tower Eigen oscillation
Amplitude 3p transverse front	
Phase 3p transverse front	
Amplitude 3p transverse rear	
Phase 3p transverse rear	
Amplitude 3p torsion	
Phase 3p torsion	
Amp. 1st tower Eigen bending mode	
Frequency 1st tower Eig. bnd. mode	

Table 3.3.2.1: VIBRO-IC signal allocation for URL download

3.3.3 Bonus2DB

Modern turbine SCADA systems include a web interface. In the case of the Bonus turbines in Nøjsomhed, this is the Wind Power Supervisor. Its architecture allows to be polled frequently, providing the most important turbine data in XML format for the whole farm. A CleverBean is currently collecting the data with 5-sec resolution, for analysis. The data collected includes rotor rpm, generator rpm and oil temperature, wind speed, ambient temperature, yaw angle, the electrical measurements for all three phases, and a few general status codes.

Since the CleverBean architecture is very straightforward, it is easy to implement an interface to a different SCADA system, which in turn can be used for the vibration sensors for their binning routines.

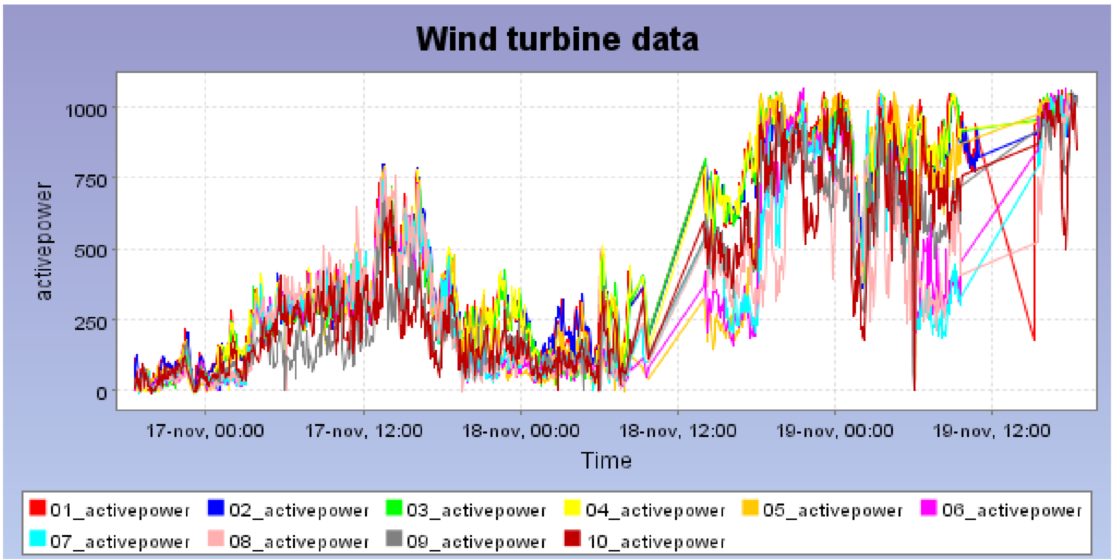


Figure 17: The active power of the first 10 turbines at Nøjsomhed for two days. To limit memory usage in the graphing application, only 1000 data points are shown per graph.

Given the simplicity of the bean, it has also been adapted to other projects, *eg* to automate data acquisition from the meteorological mast at Risø. The Prediktor bean has also been adapted to the Lokalmodell of the Deutscher Wetterdienst, for additional redundancy. Another possibility would be to use freely available data from the US National Center for Environmental Protection, since the ensemble predictions run by them go out to more than a week, thereby enabling medium-term maintenance planning.

3.3.4 CleverDam

The CleverDam bean collects the data on a regular basis from the Gram & Juhl DAM systems installed in turbines 7 and 8. The date are retrieved for different operating conditions and stored in XML format on the central CleverFarm database.

The XML data are simply retrieved from the Web server of the DAM WEB system that serves data directly on XML format.

Data has been binned according to the operating state of the Turbine as expressed by the WPS ActivePower signal. The bins used for sorting data are evenly divided in 100-kW intervals up to the maximum power of 1000-kW. “Count” is the number of measurements stored. A total of almost 13000 measurements has been stored in the database at the day of writing this report.

Based on the collected data, plots can be generated on the CleverFarm server. An example is given here. Note that measurements can be re-produced allowing for comparison over time.

The following measurements are stored every third hour in the database:

GramJuhl Dam System status

Count	Measurement	Condition
2373	Noeisomheden/IPC007/HS/FFT 0-8000	WPS-ActivePower 0-100
1220	Noeisomheden/IPC007/HS/FFT 0-8000	WPS-ActivePower 0-400
708	Noeisomheden/IPC007/HS/FFT 0-8000	WPS-ActivePower 101-200
4	Noeisomheden/IPC007/HS/FFT 0-8000	WPS-ActivePower 1051-1500
371	Noeisomheden/IPC007/HS/FFT 0-8000	WPS-ActivePower 201-300
258	Noeisomheden/IPC007/HS/FFT 0-8000	WPS-ActivePower 301-400
250	Noeisomheden/IPC007/HS/FFT 0-8000	WPS-ActivePower 401-500
138	Noeisomheden/IPC007/HS/FFT 0-8000	WPS-ActivePower 501-600
323	Noeisomheden/IPC007/HS/FFT 0-8000	WPS-ActivePower 501-949
94	Noeisomheden/IPC007/HS/FFT 0-8000	WPS-ActivePower 601-700
76	Noeisomheden/IPC007/HS/FFT 0-8000	WPS-ActivePower 701-800
65	Noeisomheden/IPC007/HS/FFT 0-8000	WPS-ActivePower 801-900
6	Noeisomheden/IPC007/HS/FFT 0-8000	WPS-ActivePower 901-949
360	Noeisomheden/IPC007/HS/FFT 0-8000	WPS-ActivePower 950-1050
2293	Noeisomheden/IPC008/HS/FFT 0-8000	WPS-ActivePower 0-100
1259	Noeisomheden/IPC008/HS/FFT 0-8000	WPS-ActivePower 0-400
634	Noeisomheden/IPC008/HS/FFT 0-8000	WPS-ActivePower 101-200
369	Noeisomheden/IPC008/HS/FFT 0-8000	WPS-ActivePower 201-300
233	Noeisomheden/IPC008/HS/FFT 0-8000	WPS-ActivePower 301-400
252	Noeisomheden/IPC008/HS/FFT 0-8000	WPS-ActivePower 401-500
119	Noeisomheden/IPC008/HS/FFT 0-8000	WPS-ActivePower 501-600
246	Noeisomheden/IPC008/HS/FFT 0-8000	WPS-ActivePower 501-949
110	Noeisomheden/IPC008/HS/FFT 0-8000	WPS-ActivePower 601-700
93	Noeisomheden/IPC008/HS/FFT 0-8000	WPS-ActivePower 701-800
75	Noeisomheden/IPC008/HS/FFT 0-8000	WPS-ActivePower 801-900
20	Noeisomheden/IPC008/HS/FFT 0-8000	WPS-ActivePower 901-949
289	Noeisomheden/IPC008/HS/FFT 0-8000	WPS-ActivePower 950-1050

Table 3.3.4.1: Measurements collected by the CleverDam bean

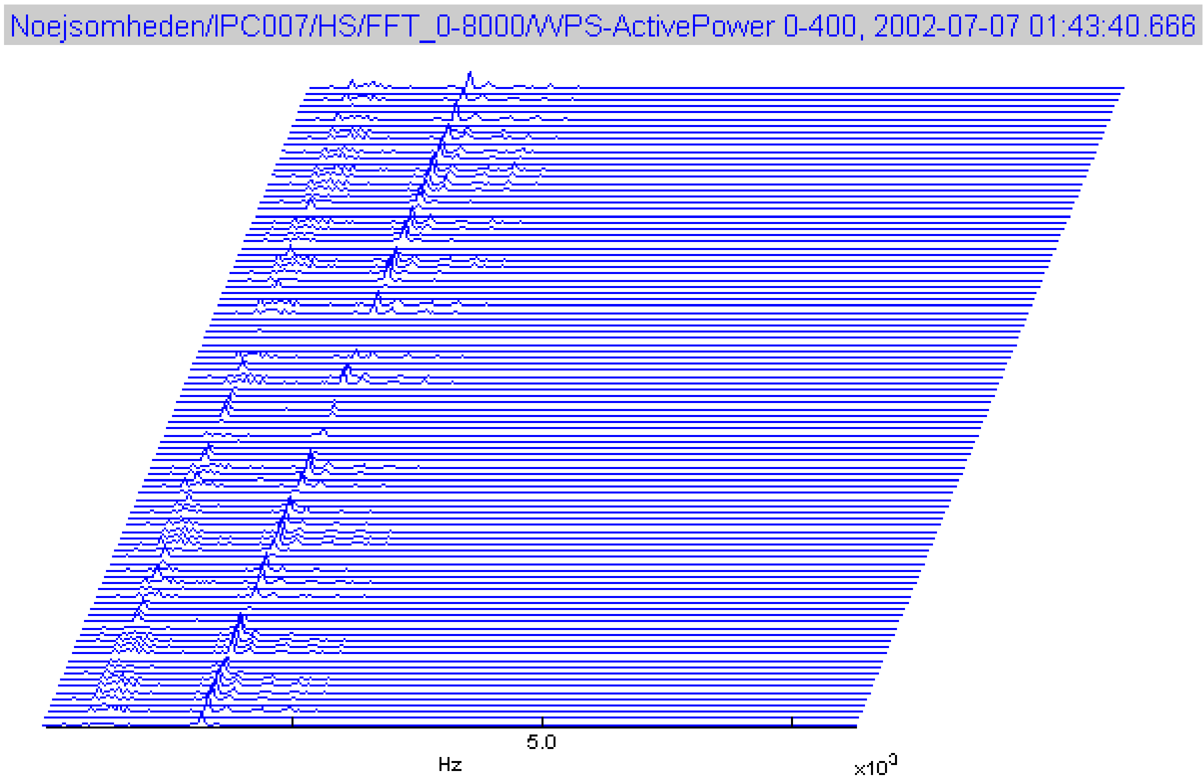


Figure 18: A trace of binned measurements. FFT measurements are plotted as a waterfall. Oldest measurements are at the bottom of the plot.

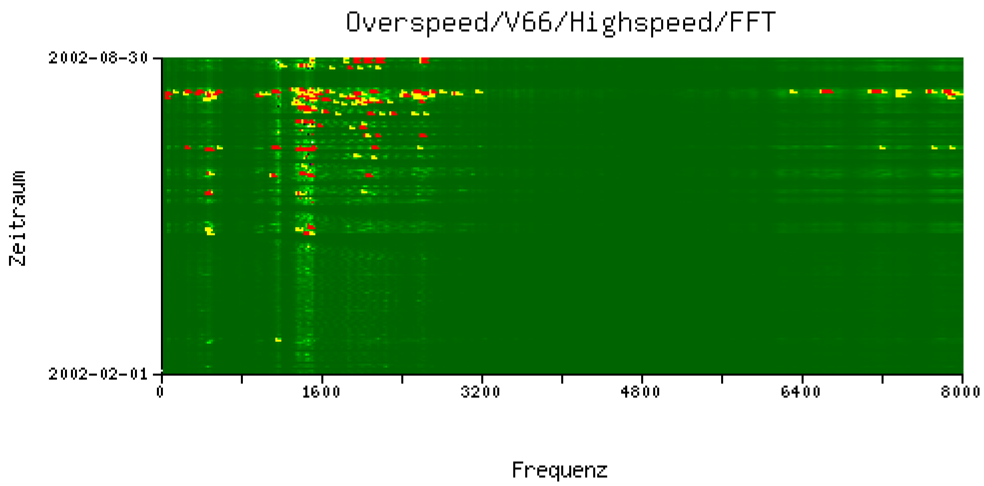


Figure 19: The sensor results as a sonagram.

3.3.5 Video

As mentioned above, the video camera is able to take pictures offline from given positions at specific time intervals. Through the video CleverBean, these parameters can be set. The taken pictures then could be accessed in the video database. In the other direction, the video system takes the yaw angle from the CleverFarm data management, which is needed to take the offline pictures at given positions not situated on the turbine. *Eg*, if a photo should be taken from a neighbour turbine, the position of this turbine plus the yaw angle of the turbine where the camera is mounted is needed. The online control applet of the video camera is used as-is as part of the CleverFarm web interface.

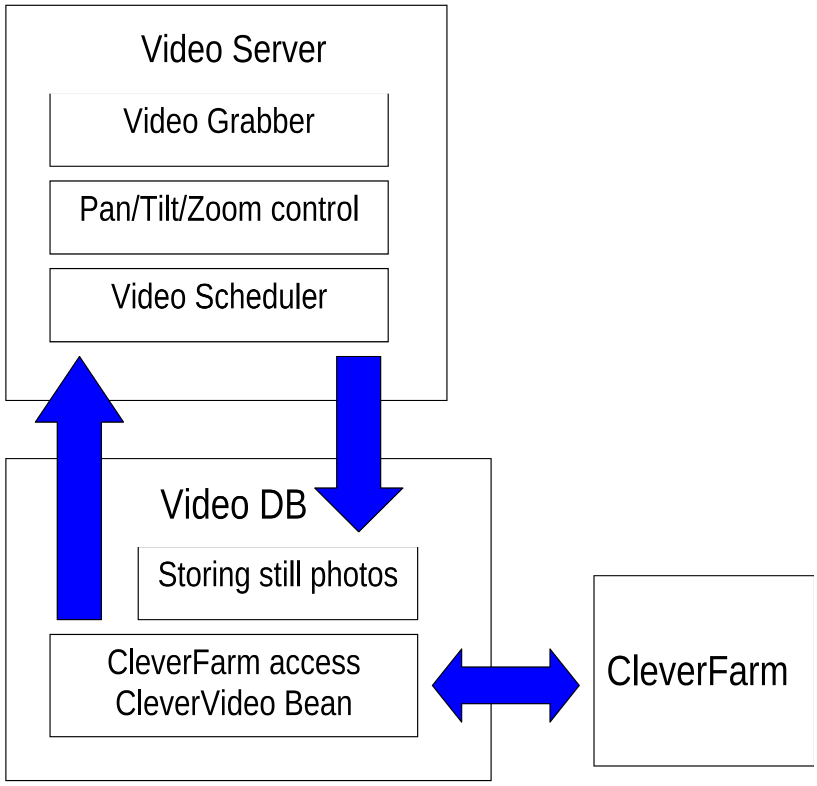


Figure 20: The flow chart of the video sub-system.

The internals of the applet and the camera server application are also shown in Figure 21. The function follows a concept similar to a standard n-tier application. Please note that only the main system parts are explained here while the application naturally consists of more classes of lesser importance for a general understanding.

The applet consists of the graphical user interface (GUI), a scheduler that buffers the action commands received via the GUI, control classes that perform the functionality requested and communication classes speaking a custom TCP/IP protocol with the server application. The retrieval of the video images is done separately by an image loader thread.

The server application consists of several manager threads and buffer threads that enable an operation without any busy wait states or deadlocks. For each incoming connection a ClientConnection object is provided, these are controlled by the ConnectionManager. The commands received are gathered and parsed by the CommandProcessor and forwarded to the individual HeadControllers via the HeadList and head representation classes. Head hereby refers to the pan-tilt head on which a camera is mounted. The HeadController is a scheduler that calls the action methods of the head and camera representation classes. These methods send the appropriate commands to the serial communication class that functions as a buffer. The head and camera representation classes are implemented for different types of pan-tilt heads or cameras. The output of the head controller circuit and the camera are received by the serial communication classes and forwarded to the head and camera representation classes that parse the input and send relevant status information to the OutputBuffer that distributes it to the ClientConnections. The ClientConnection objects send the information back to the

VideoControl applet which displays relevant status changes to the user, *eg* a new pan or tilt angle of the camera head.

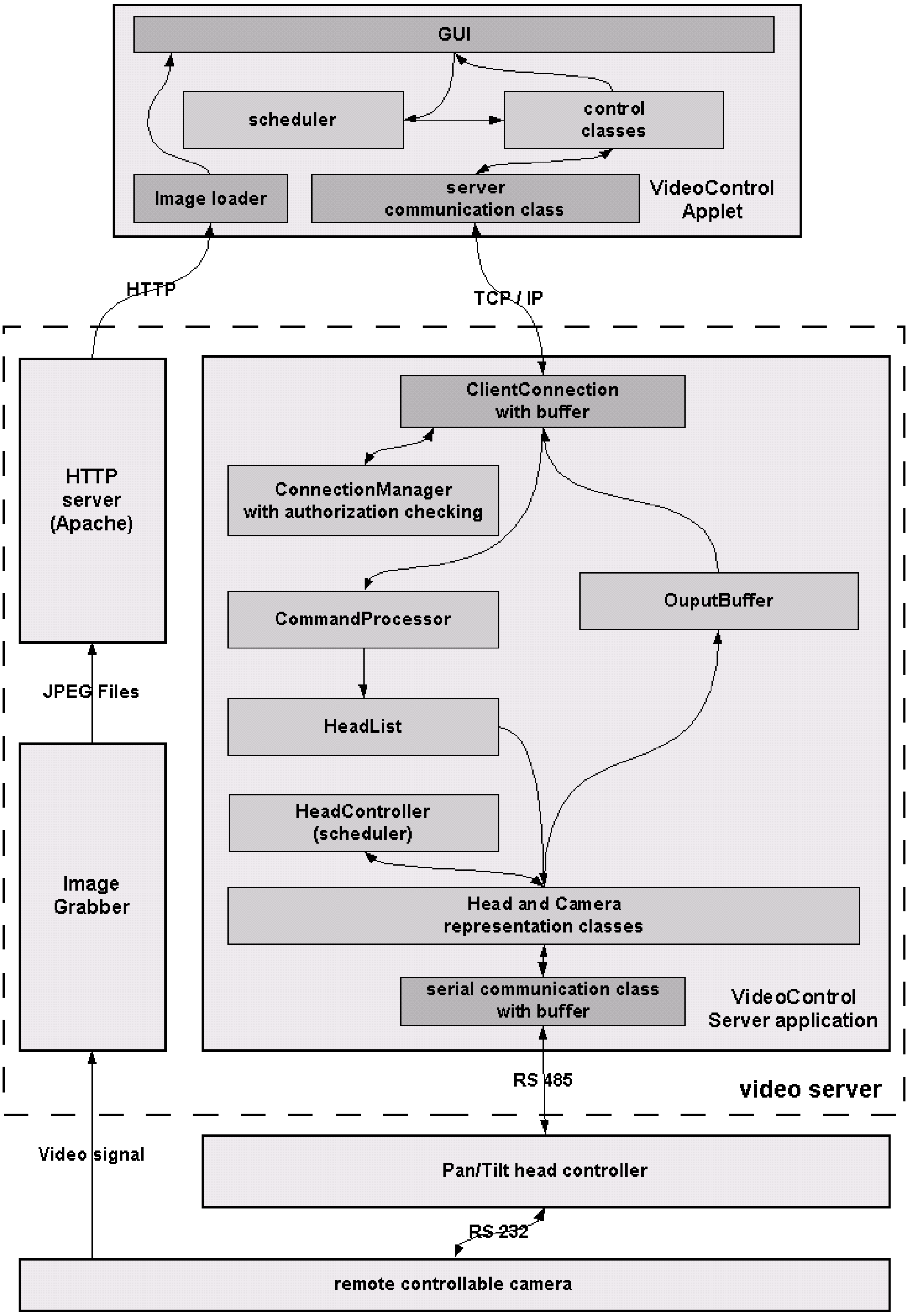


Figure 21: Overview of video surveillance software systems.

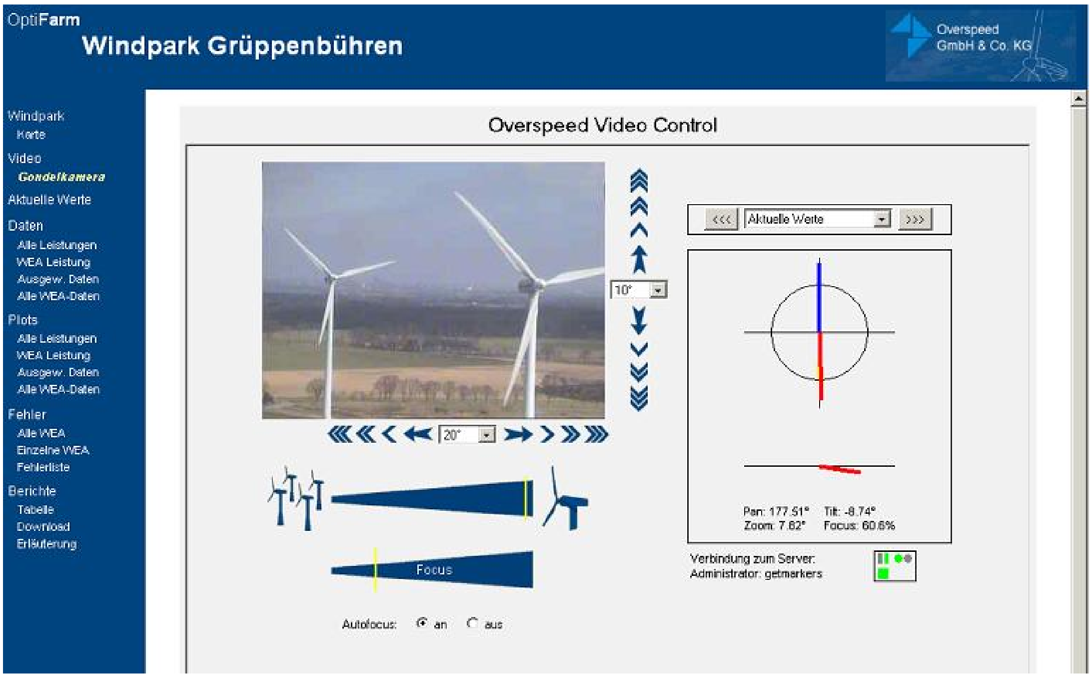


Figure 22: Interface (java applet) to the pan/tilt/zoom camera. The camera could be totally controlled by the remote user. In addition, a scheduler is implemented to take still pictures from given positions at certain time intervals.

VideoBean

The video bean provides an interface to put and delete new task from the video scheduler (which takes photos from given positions, called markers, at given time intervals) and to retrieve historical photos. It has the following functionality

- setMarkers: sets a new marker
- getMarkers: retrieves a list of markers set
- setSchedulerEvent: puts a new event (marker and time interval or time point) to the scheduler event list
- getSchedulerEvents: gets all events from scheduling list
- getPictures: retrieves pictures for a given marker and time interval

3.3.6 OptiFarm

Interface

OptiFarm is a competing product to CleverFarm by Overspeed, Oldenburg. Nevertheless, the focus here is the technical operation of the wind farm. In practice, the data access to the wind farm system is always a problem. Because this is solved for different kinds of wind turbine types in OptiFarm, an interface between OptiFarm and CleverFarm is introduced. Here the turbine operational variables could be fed from OptiFarm to CleverFarm. In this way, CleverFarm can be operated on top of the data gained by OptiFarm, so the functionality of CleverFarm like the data mining facilities could be used in addition to the existing system.

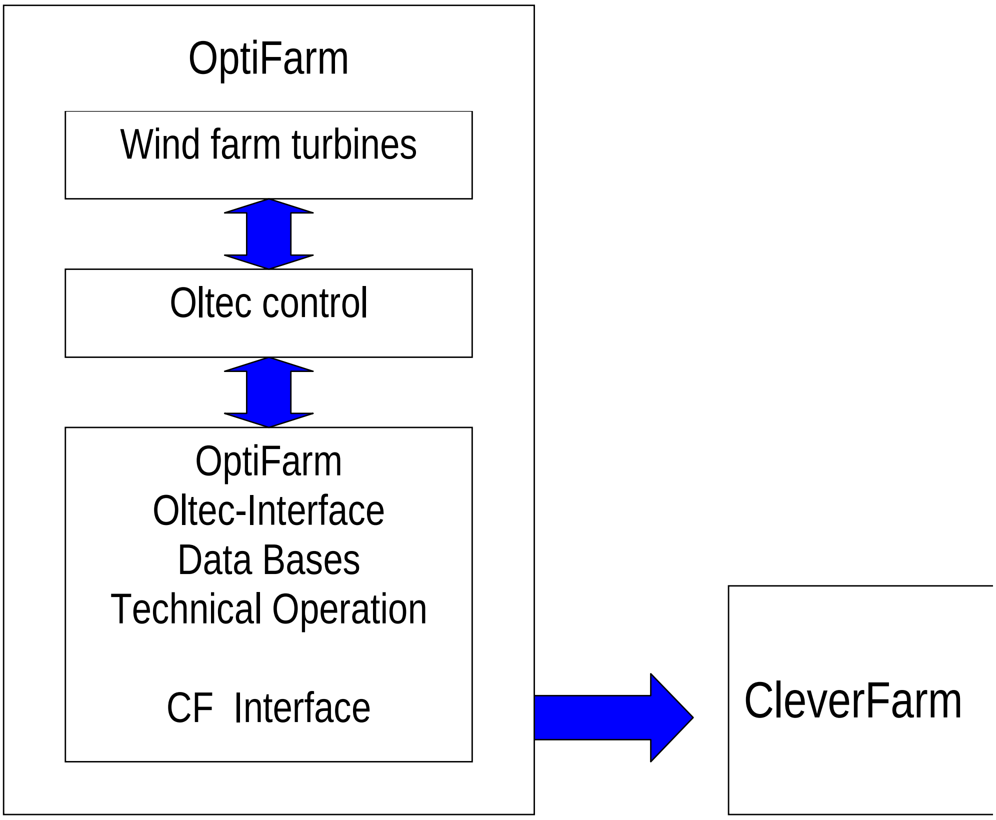


Figure 23: The flow chart of the OptiFarm-CleverFarm interface.

CMS integration

One practical problem with the appliance of CMS is the fact that all these systems bring their own user interface. This means that the wind farm operator has to be familiar with all these variants. In addition, each CMS also has an own alarm system, so that the end user must be again aware of different alarm channels like sms, email, fax and its configuration. To overcome these problems, a CMS alarm interface was written which leads to the following advantages:

- General presentation of current alarm status (as a traffic light) and of the measured spectral data as a sonagram (see Figure 24). First, the sensor overview just shows if any alarms are reported currently by the CMS by a colour dot at the sensor position (red, yellow, green).
- On the next refinement level, a sonagram shows the temporal development of the frequency lines. The y-axis shows the time (here: about half a year), and each horizontal line represents one spectrum. The colour of this is green if everything is fine, and the brightness of the green colour corresponds to the spectral amplitude. If the set alarm levels are crossed, the colour of the spectragram changes to first yellow and then red. In this way, the operator can see on one view how the data that produced the alarm did develop in time.
- By clicking into the sonagram, the detailed spectra are shown, giving the current values and the yellow and red alarm level.

In CleverFarm, the G&J CMS is interfaced by a common database. Other possibilities are email or ftp interfaces.

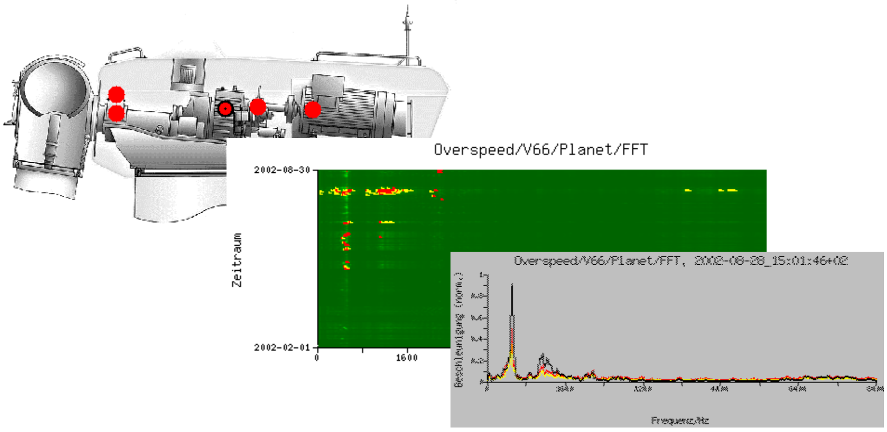


Figure 24: Alarm and interface hierarchy of the CMS interface. On top level, only the traffic light alarm information for each sensor is shown. Then, the temporal development of the measured spectra is visualized as a so-called sonagram (in the middle). From there, the detailed information of the according measurement including the current measurement and the red and yellow alarm spectra are shown (lowest picture).

3.3.7 Shadow flicker

In some cases of current wind farm installations in Central Europe, local inhabitants in the near vicinity of a wind farm are influenced by the moving shadow of the rotor blades at certain times in the year. *Eg*, in Germany there are official limits for the time where shadow flicker is allowed to occur. For CleverFarm, Overspeed did implement an interface to their shadow flicker shut-off system *Shade-off*. From CleverFarm, *Shade-off* gets information about the current state of the turbines, esp. if the rotor is turning at the moment and if there is power produced. From an internal model calculation, *Shade-off* fixes the times when shadow flicker could occur and checks this against the legal limits. In cases where a shutdown of turbines is necessary, *Shade-off* tries to optimise such that the turbines are preferably shut down in periods with low wind speeds.

The command to switch off is handed back again to the farm via the CleverFarm interface.

3.3.8 Online wind farm energy modelling

In a wind farm, for several turbines the power is reduced due to the mutual shadowing. The upwind turbine takes momentum from the flow, which reduces the wind speed behind the rotor to a certain extent. The amplitude of this effect must be known for two reasons:

The energy production of the turbines in a farm depends on the position in the farm, the prevailing wind direction etc. In order to compare and check the production from different turbines, the measure data must be corrected for these wake effects.

In case of turbine failures, the energy lost due to the standstill also depends on which turbine in the farm is affected.

To overcome these two problems, an online simulation of the wind farm effects was implemented. Basis is a wind farm modelling system developed at Oldenburg

University, FLaP, and implemented as an online tool by Overspeed. From CleverFarm, *FLaP online* takes the turbine characteristics like position, type, power and thrust characteristics, and the operational data like error state, wind speed and wind direction. The results are given back to CleverFarm as energy yield corrected according to the farm effects, and the modelled lost energy in case of turbine failures.

3.4 CleverMiner

The CleverMiner tool was developed to provide a generic means of exploring information from the CleverFarm system. Specifically it provides web-based plotting for any combination of x,y values from the CleverFarm database. The columns available for the user to plot are entered into a table 'colnames' in the CleverData database. This allows a mapping to be made between the database column name and a 'friendly' name, which the user sees in the drop-down list. Figure 25 and Figure 26 give examples of typical output.

3.4.1 How it works

The CleverMiner uses the following components:

jFreeChart java library

JFreeChart is a freely-available open-source java charting library. As it takes care of the mechanics of plotting graphs, the developer only has to provide a suitably-formatted dataset, greatly reducing development time. See *www.jfree.org*.

Custom servlet

The CleverMiner servlet is a server-side process, written in Java, which obtains the dataset to be plotted based on information provided by the user. The data is retrieved from the CleverFarm MySQL database, passed to jFreeChart which outputs a png graphic of the desired graph. This graphic is then returned by the servlet to the requesting web page.

Custom jsp pages

The CleverMiner jsp is a java server page that initially displays a list of available tables from the database. When the user validates this choice, the drop-down lists (start/end date and variables), which can be seen in the above figures, are then dynamically filled from the database. Once the user selects the date/time range, values to be plotted and plot type (scatter, time series or area), this is sent to the tomcat server through the URL, which then passes it to the servlet.

3.4.2 What's Next

The number of variables available for plotting is only limited by the number of entries the user places in the colNames table. However, the following enhancements could be considered:

- More than two data series on the same graph.
- Plot data from more than one table on the same graph
- Ability to select time range as well as date range
- Skip factor to only plot, say, every other data point for clarity.
- Save results to text file
- Additional processing, *eg* data binning or trend analysis

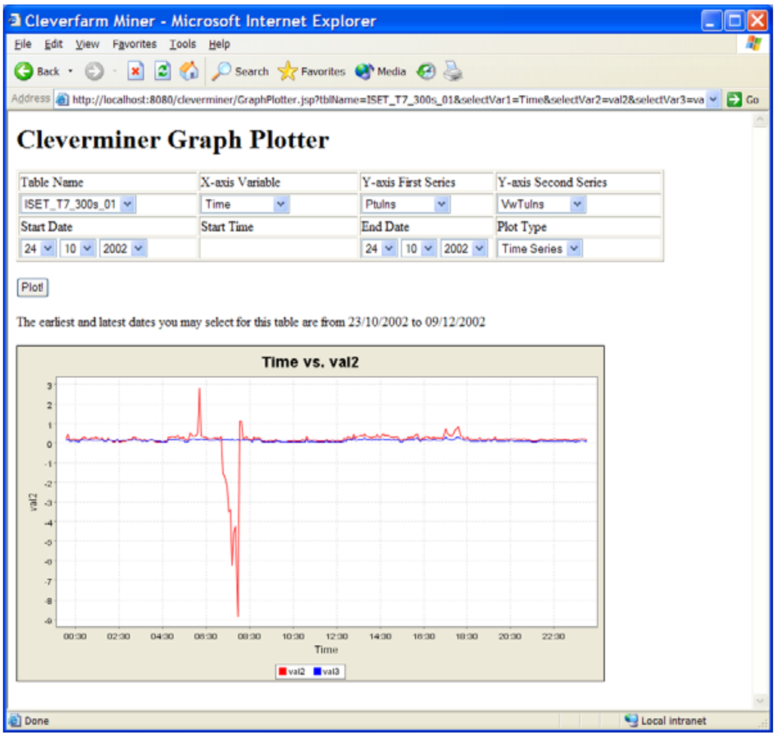


Figure 25: Time series plot of power and wind speed turbulence intensity for T7.

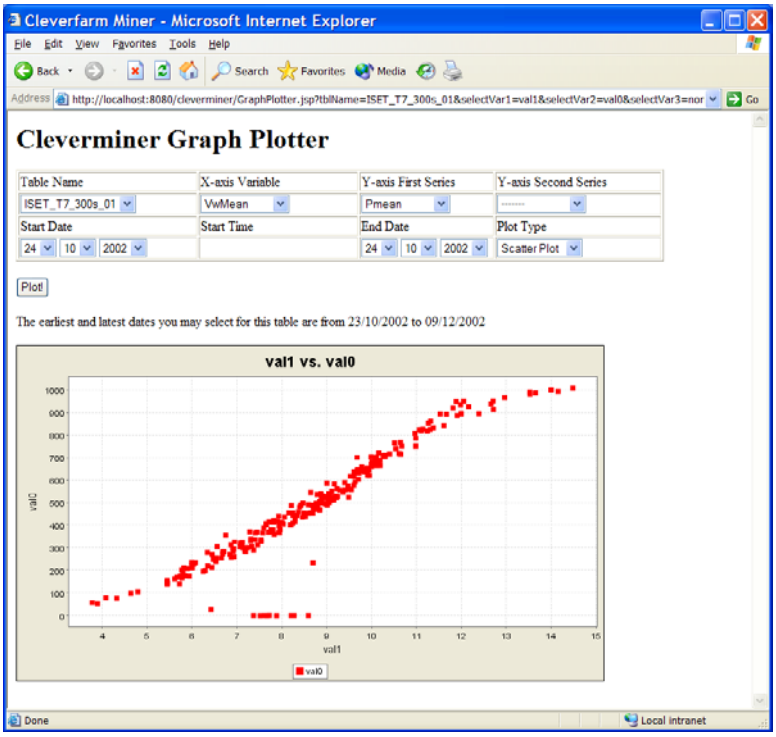


Figure 26: Scatter Plot of 300s mean power vs. mean wind speed for Turbine 7 on the 24th October.

4 The installations

4.1 Nøjsomheds Odde

In two installation sessions on February 20, 2001, and on May 15, 2001, Gram&Juhl and ISET installed their vibration monitoring systems in SEAS' offshore test bed in Nøjsomheds Odde.

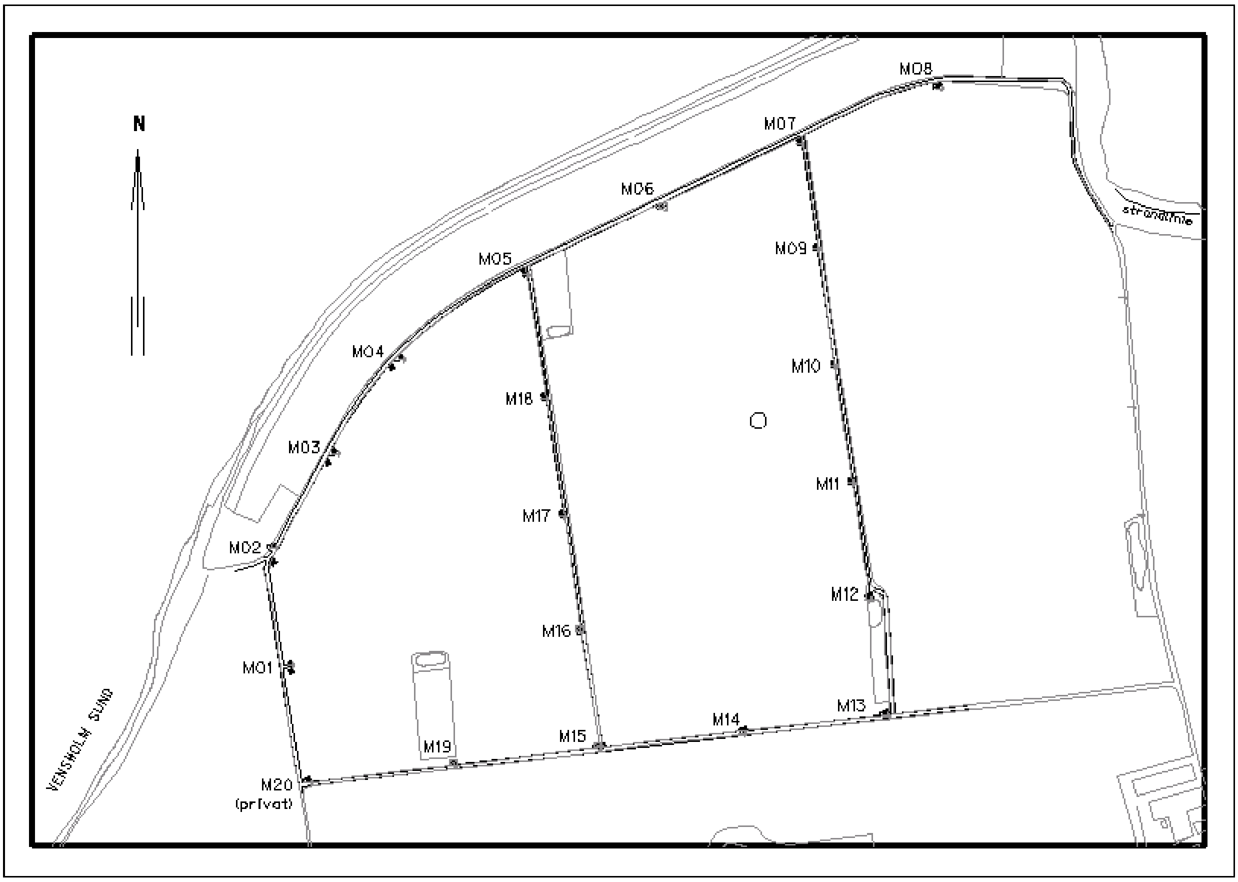


Figure 27: A map of the Nøjsomheds Odde wind farm on the Danish island of Lolland.

Turbines 7 and 8 were instrumented by both systems. Additionally, Overspeed put an industrial video camera on top of turbine 7, and a smaller webcam inside of its nacelle.

Figure 28: The wind farm at Nøjsomheds Odde on a particularly sunny day.

Figure 29: Turbine 7 with the video camera on top, as seen from the ground.

Figure 30: The Overspeed video camera, with full pan-tilt-zoom functionality. The long metal rod on the side is the lightning protection for the camera.

The camera consists of a heavy-duty outdoor pan/tilt head and the high quality video camera itself (zoom and auto focus) in a housing. The camera and the head are controlled via a specially developed RS232/485 remote interface from the server that is situated in the tower base of the turbine. Standard web cams are not suitable for that purpose because they in general could not provide uncompressed pictures, which are important for the investigation of details like blade damages. Therefore, the video pictures are transferred to the video server by an analog video line and digitised there in the requested quality.

Figure 31: The inside webcam to monitor the nacelle of turbine 7.

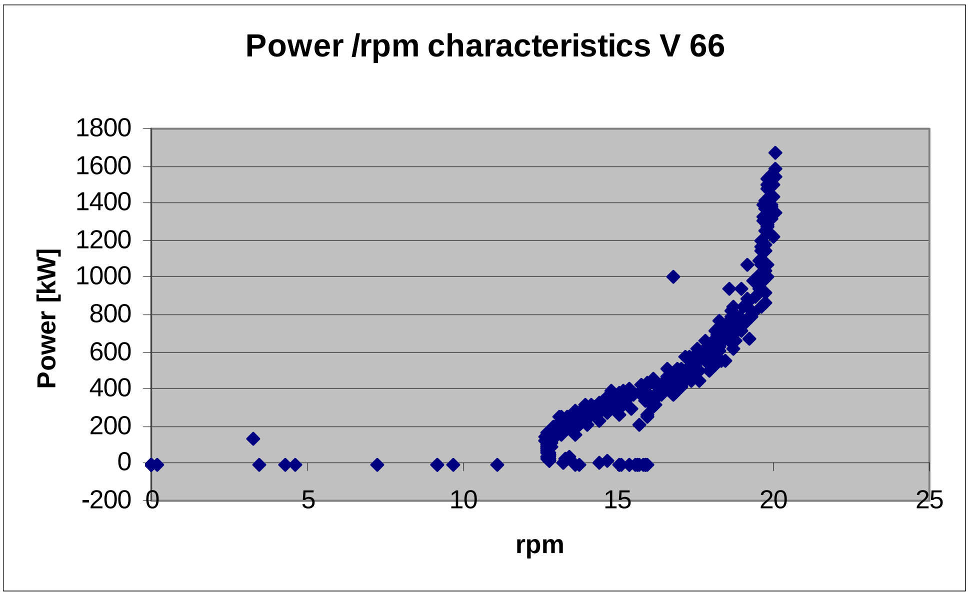
One problem with the inside webcam was the light for it. To use it in the dark nacelle, you need to be able to remotely turn on the lights. However, until the end of the project we never managed to get it working.

4.2 Grüppenbühren

The wind farm Grüppenbühren was chosen due to two reasons: first, the farm (6 Vestas V66) is equipped with the Oltec farm control, which is one of the standard wind farm SCADAs in Germany (all Vestas and Bonus turbines there are equipped with it). Additionally, the farm has a permanent DSL online internet connection, so it easily could be accessed from everywhere. On top of Oltec, the Overspeed OptiFarm system is running, which itself interfaces to CleverFarm again (see section 3.3.6).

Figure 32: The layout of the Grüppenbühren wind farm. The motorway seen is the A1 west of Bremen.

As a part of CleverFarm, one turbine in the park was equipped with the Gram&Juhl Condition monitoring system. This farm is an ideal test case for this because the Vestas turbines are operated at variable rotational speed. This makes the process of handling the vibration data much more complicated than on a turbine with fixed rotational speed like from Bonus. In addition, rotational speed and power output is needed in the CMS for classifying the measured spectra and so on. Through CleverFarm, it is possible to feed in this information into the G&J system. In practice, this attempt did not work because the time resolution of the data from the wind farm SCADA via OptiFarm or CleverFarm (typically 10 seconds to a minute). Due to the very steep power/rotational speed characteristics (see Figure 33), this is definitely too slow for use in the CMS. From our practical experience, it seems that the parameters needed for condition monitoring must definitely be measured by the CMS itself. It is very difficult to get the according information from the turbine controller in a reliable manner with a given, high temporal resolution. On the other hand side, the CMS-data is completely useless if the operational data of the turbine is not reliable.



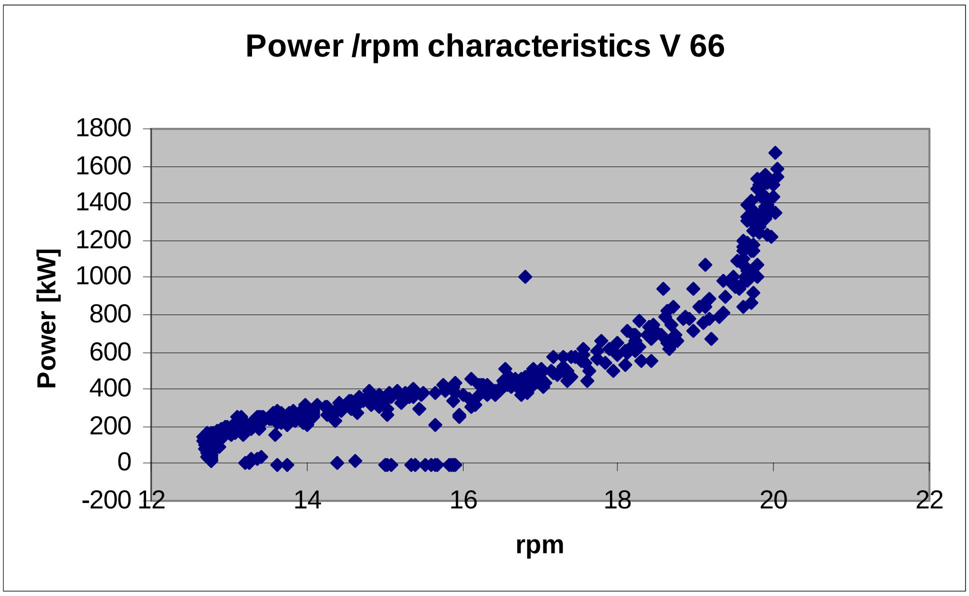


Figure 33: Power/ rotational speed characteristics for the wind turbine type V66 installed in the wind farm Grüppenbühren. Above 800 kW, the characteristics are very steep. Therefore, average data of rotational speed could not be used for classification of power bins.

Figure 34: The wind farm at Grüppenbühren.

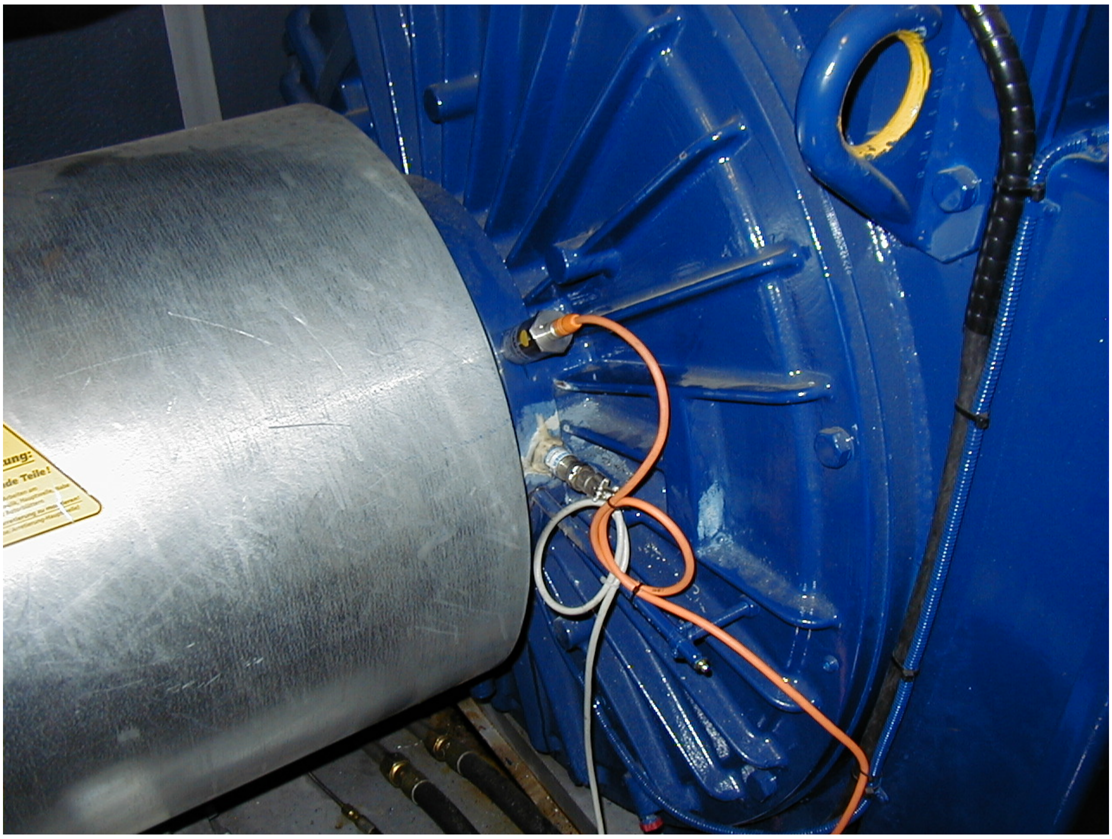


Figure 35: The DAM sensors in the turbine in Grüppenbühren.

5 Clever algorithms

5.1 A Gram&Juhl success story

None of the monitored turbines failed during the project, leaving us therefore with relatively unspectacular data. However, here is one example of how Gram&Juhl's vibration monitoring system could detect a fault in the planetary stage bearings of a similar gearbox already **one year in advance** in a 1-MW turbine in Denmark. This was not within the CleverFarm project, but used the same technology.

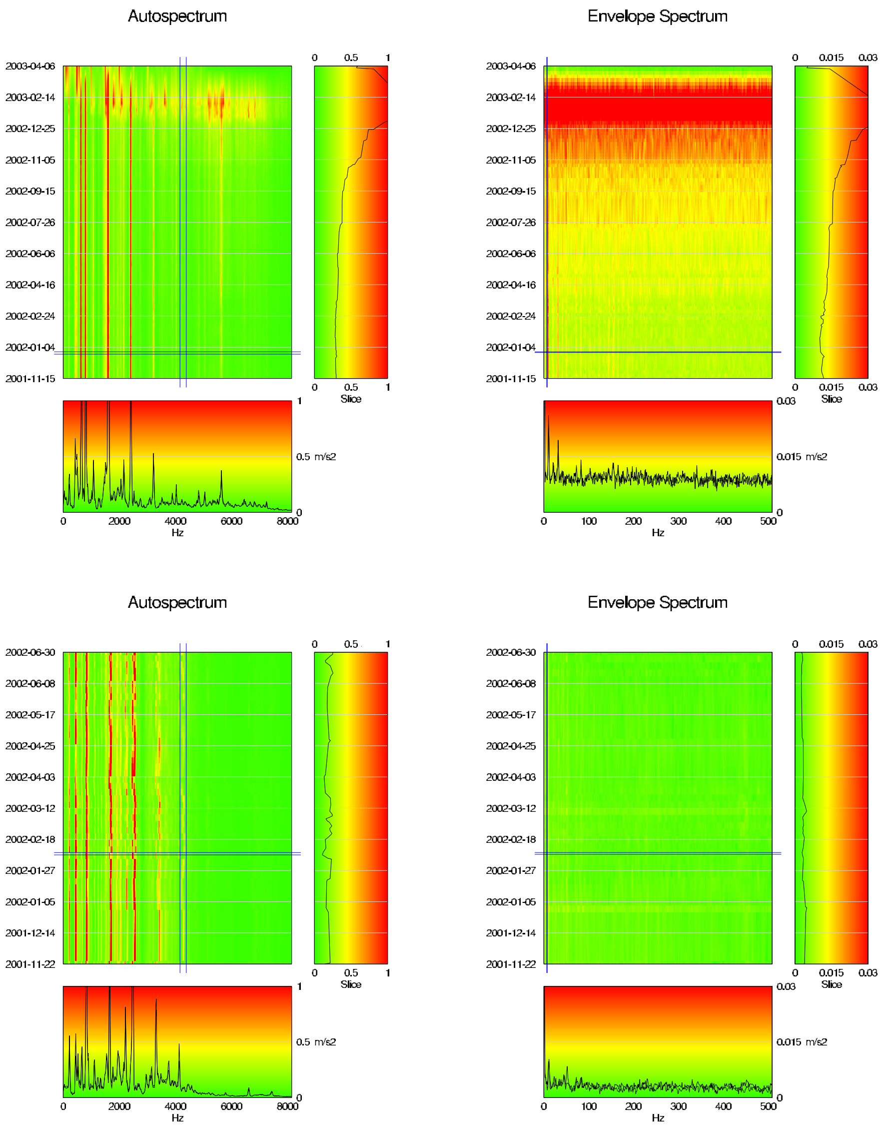


Figure 36: Two spectra of gearbox vibration measurements. Top row, a broken gearbox. Bottom, the comparison.

The measurements from the 1 MW wind turbine shown here are the Broadband Auto Spectrum and the Envelope Frequency Spectrum. The measurement plots below in Figure 36 shows the evolution of a gearbox damage compared to the same measurements from an identical healthy gearbox in the same park. These result where automatically generated, but all historical data are online available for further expert analysis.

How to read the measurement plots: The plots consist of a contour plot showing the vibration level over time (y-axis with dates). To the right a RMS slice in time is shown. The slice selection in frequency is marked with two vertical blue lines. Below there is an overlay plot, *ie* all measurements in between the dates marked with two horizontal blue lines.

5.2 Results of the data mining

Evaluation of VIBRO-IC data

The following section gives some examples of the data evaluation methods, which are used to document the actual condition of the test wind energy converters (turbine 7 and 8) in the Nøjsomheds Odde wind farm.

Overall operational characteristics

To obtain diagnosis information about any performance degradation of a wind energy converter, resulting for example from increased rotor blade roughness, icing or faults of the generator system, continuous supervision of the power characteristic is performed. For this purpose wind speed measurements from the nacelle are very well suited because the evaluation is only based on relative changes in the characteristic and not on comparisons with the plant's nominal power curve. By using the built in anemometer, wind speed can be measured without any additional sensor installation. For a detailed discussion of power characteristic monitoring see [6, 7, 8].

Figure 37 shows the learned power characteristic of turbine 8. The small blue dots result from pairs of measured 5 min. mean values of power P and wind speed v_W. The red circles are the classified mean values. Class width is 0.5 m/s. The outer red lines are absolute alarm limits determined by the CMS application programmer. The inner red lines are learned alarm limits. The learning process is done by classification of the measured $P(v_W)$-values. The bandwidth of the learned alarm limits is calculated out of the class mean values combined with the standard deviation of the respective class.

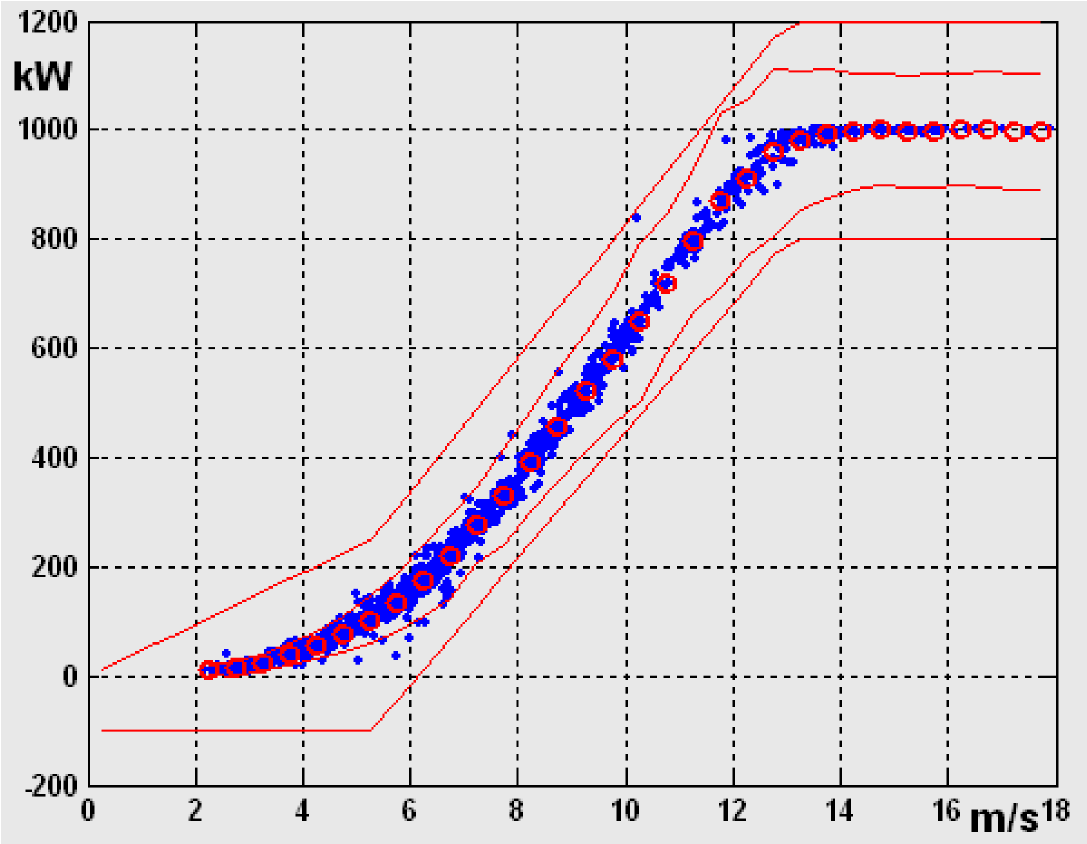


Figure 37: Power characteristic with learned alarm limits.

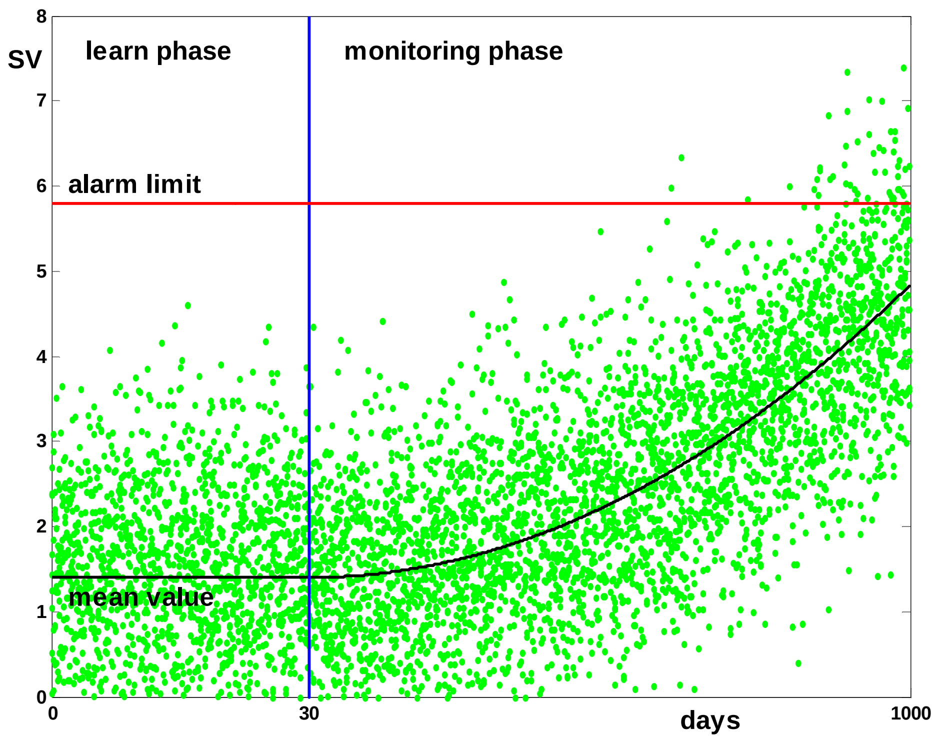


Figure 38: Learning process for scalar values

Gearbox and bearings

Monitoring and fault prediction in gearbox and bearings is performed by measurement of broad band characteristic values like bearing condition unit (BCU) and Crest (which is the peak value divided by the RMS value in a time signal). These characteristic values give information about the overall condition of gearbox and bearings. If a rising trend for these values is detected, additional diagnosis for bearings can be derived from more detailed evaluation algorithms, like envelope curve analysis which uses modulations of high frequency resonances to evaluate specific fault frequencies. Figure 38 shows the principle of the procedure for monitoring of a scalar characteristic value, like the above mentioned BCU or Crest values. After performing a learning phase (30 days in this example), the alarm limit is calculated out of mean value and standard deviation of the measured characteristic value. In the monitoring, every incoming measured value is checked against the alarm limit. If more than 5 subsequent values exceed the limit, an alarm is triggered.

Figure 39 shows the trend plot of the BCU values of the generator bearing in turbine 7. The BCU signal depends on the rpm of the rotor/generator. Therefore, the values are classified by the generator rpm, which is 1000 rpm in the low generator stage (shown as red dots) and 1500 rpm in the high generator stage (blue dots). There is not rising or falling trend visible in the data, so this points to an intact bearing.

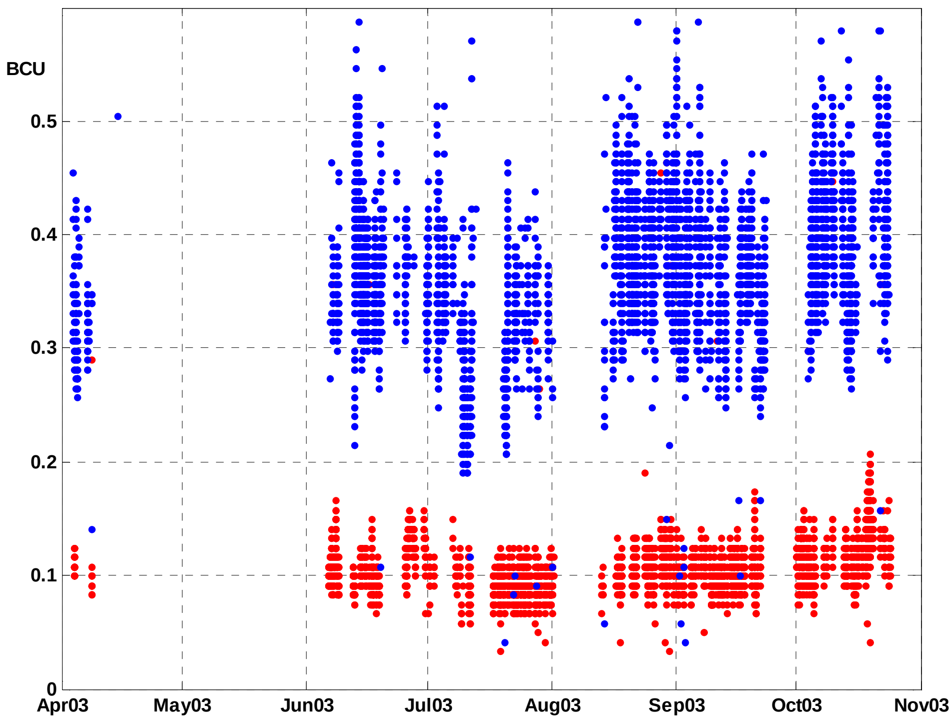


Figure 39: Trend plot of the BCU value on generator bearing. Red dots stem from the low generator stage, blue dots from the high stage.

A trend plot of the Crest Factor values on the gearbox is given in Figure 40. As can be seen, there is no obvious trend in the time history of the data. The data points show a quite high variance. The reason for this is not so clear at the moment and will require further investigations with the data. The Crest Factor does not depend on the actual generator stage. Therefore, only the data for the high generator status are shown. The Crest factor does depend on the actual load condition of the wind energy converter. So these values have to be classified according to the power output to yield reasonable trend analysis.

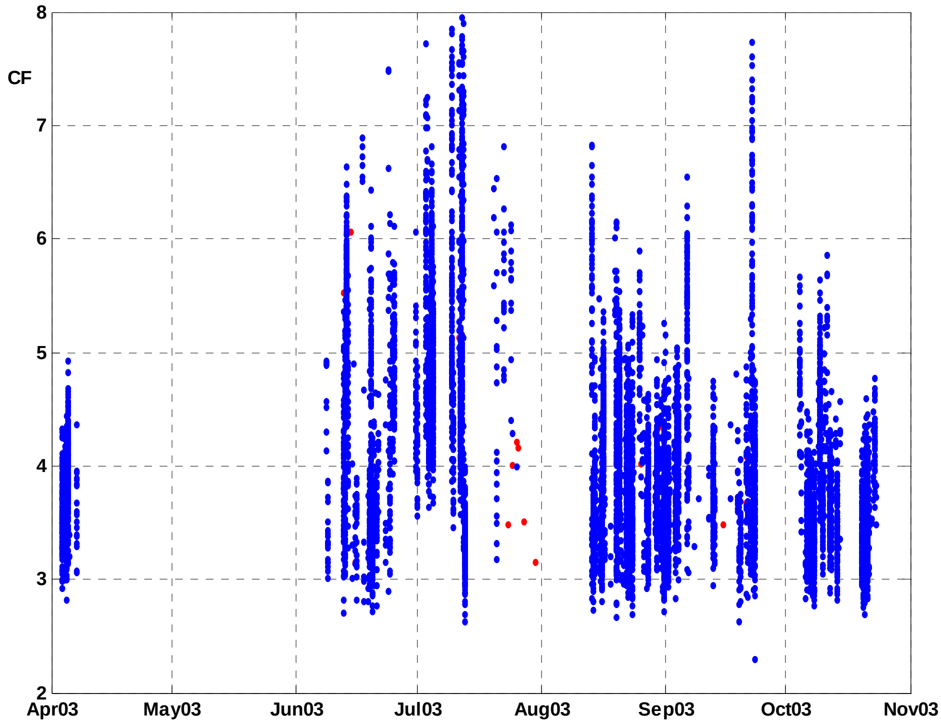


Figure 40: Trend plot of the Crest factor value of the gearbox.

Structure (Tower)

To monitor the structural health of the wind energy converter, the frequency of the first tower Eigen bending mode is measured. If the tower has fatigue effects it can be expected that this frequency will decrease, because of the decreasing stiffness of the material. Figure 41 shows the trend plot of the frequency measurements of the first tower Eigen bending mode at turbine 8. The mean value is 0.49 Hz. In Figure 42, the correlation between power output and Eigen frequency is shown. As can be expected, there is no correlation between the two measures.

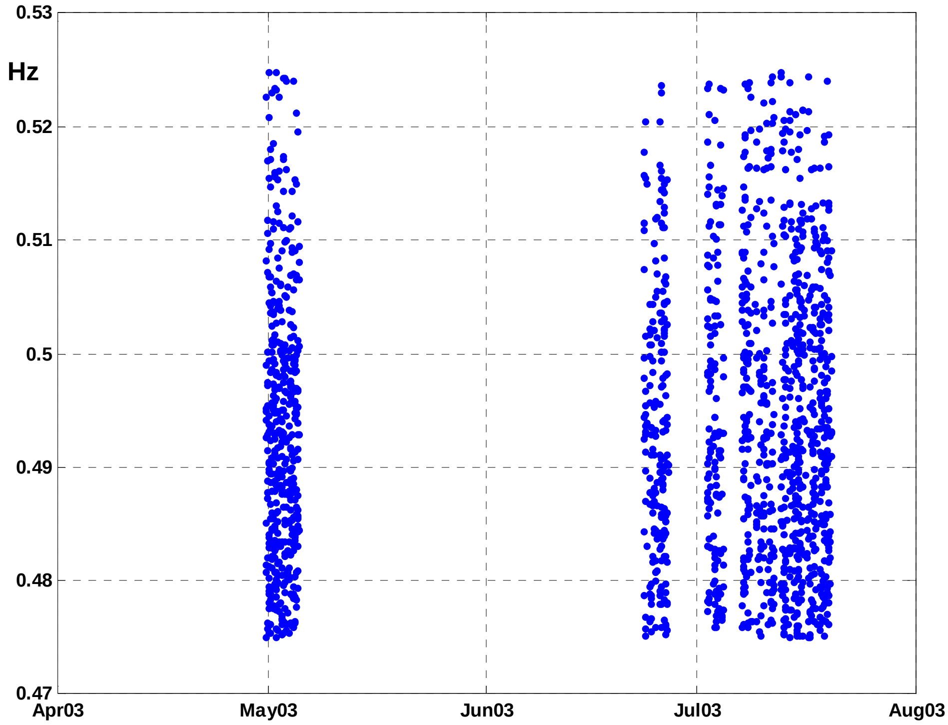


Figure 41: Trend plot of first tower Eigen frequency.

Figure 42: First tower Eigen frequency versus power output.

Rotor faults

Methods to detect rotor asymmetries (*eg* mass imbalance, aero-dynamic asymmetry, yaw misalignment) are based on spectral analysis of low frequency nacelle oscillation. Analysis of these nacelle oscillations induced by the rotor has proven to be very useful to detect and distinguish rotor fault conditions. Characteristic values resulting from spectral analysis algorithms appear as complex values with real part and imaginary part. These complex values can be interpreted as vectors. Both amplitude and phase of these vectors are significant to certain fault conditions.

Figure 43 shows the principle functionality of the algorithm for monitoring of vector characteristic values. To monitor amplitude and phase simultaneously the alarm limit is represented by a circle around the tip of the mean value vector X_0. Exceeding of alarm limit is checked by calculating the differential vector to the actual measurement (X_1, X_2 resp.). With this circle limit changes in phase with constant amplitude (differential vector dX_1) will be detected as well as amplitude deviation without significant phase shift (dX_2).

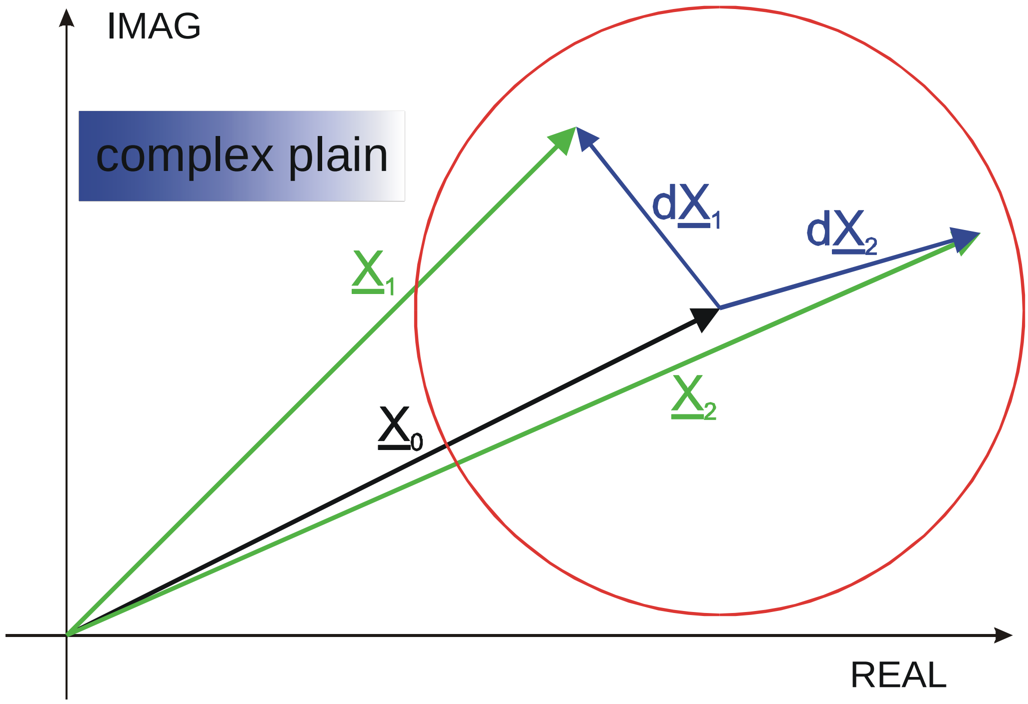


Figure 43: Principle for monitoring of vector characteristic values

Measurements of low frequency nacelle oscillation are described in section 2.2 of this report. As an example, Figure 44 shows a polar plot of the 1p amplitude and phase of the transverse nacelle oscillation of turbine 7. The 1p frequency is the spectral component at rotor shaft frequency. The transverse direction is excited, when the rotor has a mass imbalance. That means that the absolute mass or the mass deviation of one of the blades is different from the others. This causes the centre of gravity of the rotor to move away from the rotor shaft. There is then a resulting mass, which rotates with the rotor frequency. Each dot in the polar plot represents a measured amplitude and phase data set, resulting from a spectral analysis window of 90 sec. and the calculation of the power density spectra, performed with an improved FFT algorithm.

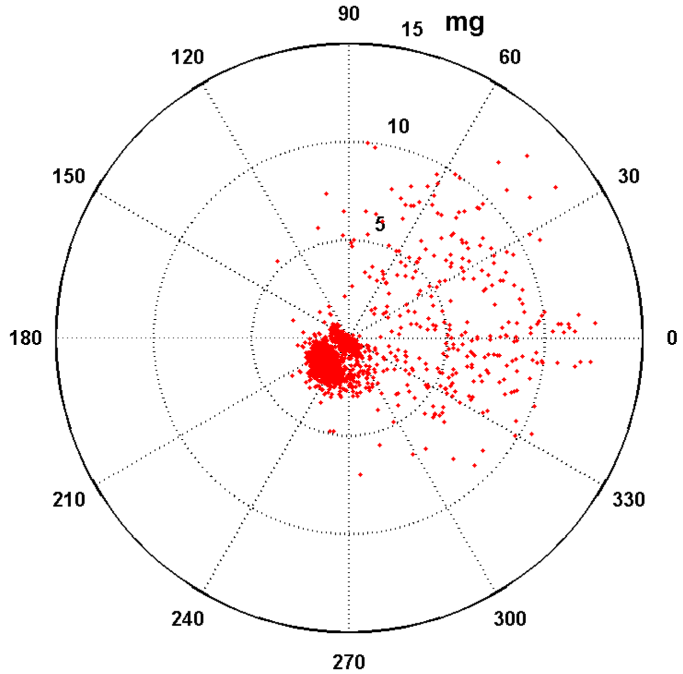


Figure 44: Polar plot of 1p transverse nacelle oscillation

As can be seen, most of the measurement points are sited in a clear limited area at amplitudes of 0 to 3 mg and a phase angle of app. 220 degree. This points to a minor mass imbalance, which is likely to be within the production tolerances of the blades. The data points in the right half circle (90 – 0 – 270 degree) with higher amplitudes result from measurements taken at higher power output of the turbine, when the active stall power regulation is in operation. This can be proven by plotting the amplitude versus the power output, which is given in Figure 45. The amplitudes are constant in a range of 250 kW to 750 kW. In this case, one will decide to use data points only from this power range to monitor rotor mass imbalance.

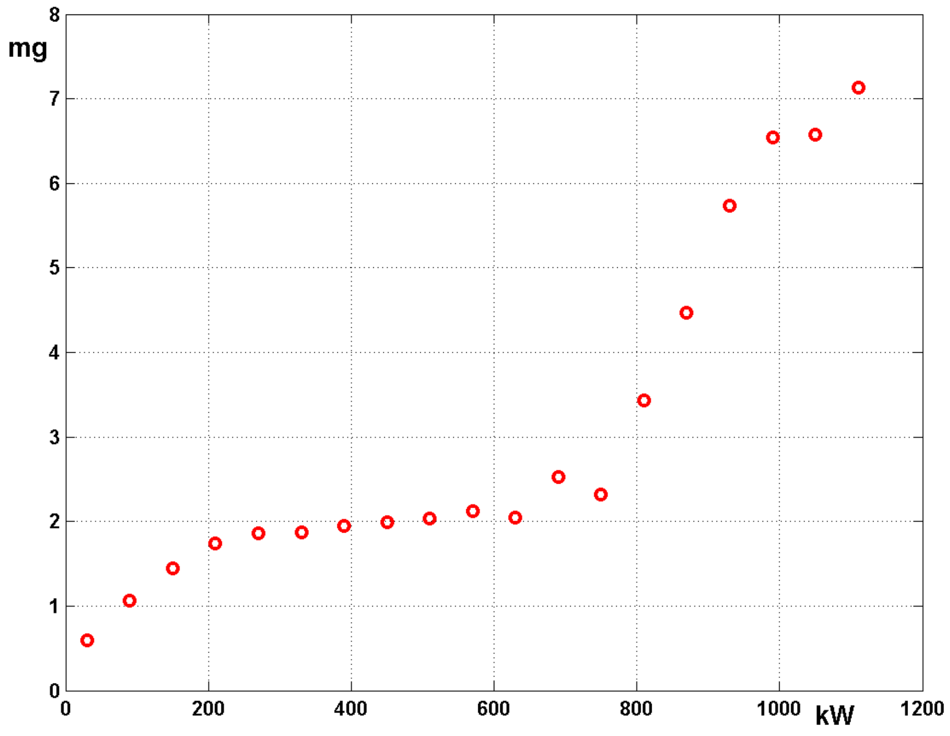


Figure 45: 1p transverse nacelle oscillation versus power output

Another possible fault of the rotor can be an aerodynamic asymmetry, *eg* if one blade angle differs from the others due to mechanical/electrical problems with the pitch drive train. In case of an aerodynamic asymmetry, the torsional 1p oscillation around the tower vertical axis and in axial direction will increase. Figure 46 and Figure 47 show the respective polar plots. The torsional oscillation has no significant amplitude and phase, so this points to an acceptable rotor condition due to aerodynamic asymmetries. The axial oscillation shows higher amplitudes and a quite constant phase. This could possibly be caused by excitations of the 3p-frequncy (the so called blade passing frequency), when the rotor blades pass the turbine’s tower. Further investigations about this effect have to be done.

Figure 48 and Figure 49 show the power correlation of torsional and axial nacelle oscillation respectively. For the torsional amplitudes the same range of power output (250 kW to 750 kW) is suitable for valid data points. The axial amplitudes are much less depending on the power output, so the range for taking data points can be extended, *eg* from 200 kW to almost the rated power of 1000 kW.

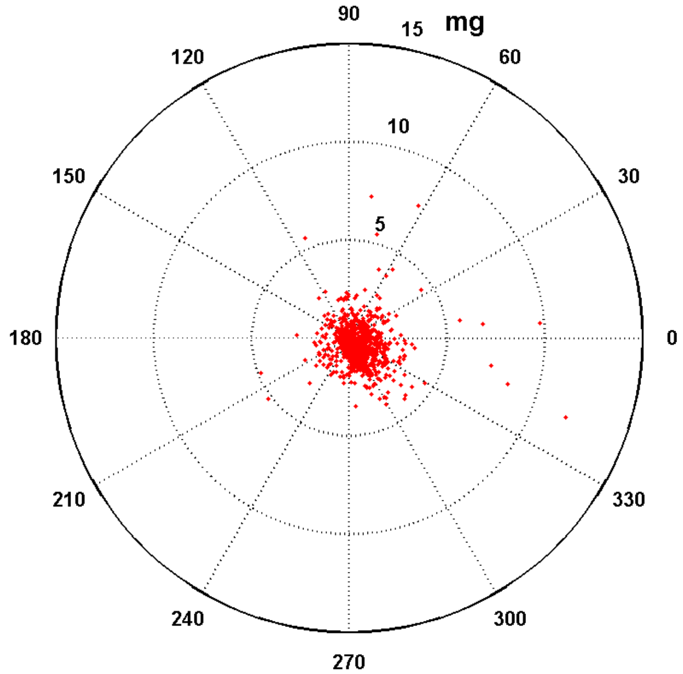


Figure 46: Polar plot of 1p torsional nacelle oscillation

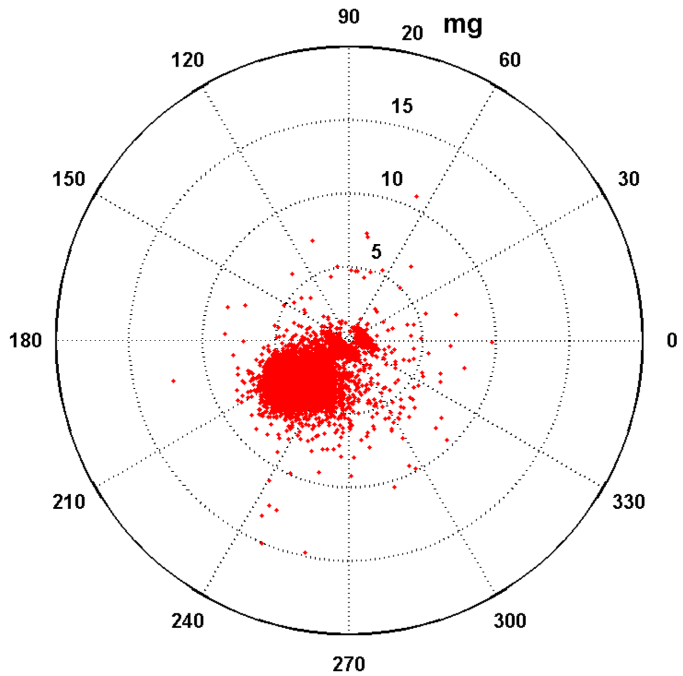


Figure 47: Polar plot of 1p axial nacelle oscillation

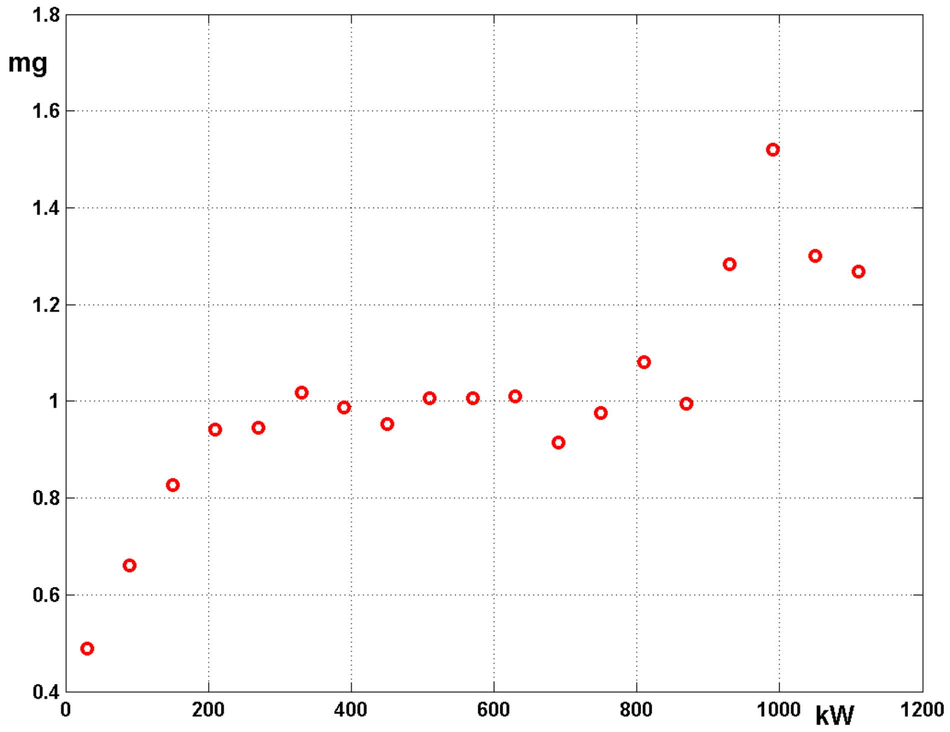


Figure 48: 1p torsional nacelle oscillation versus power output

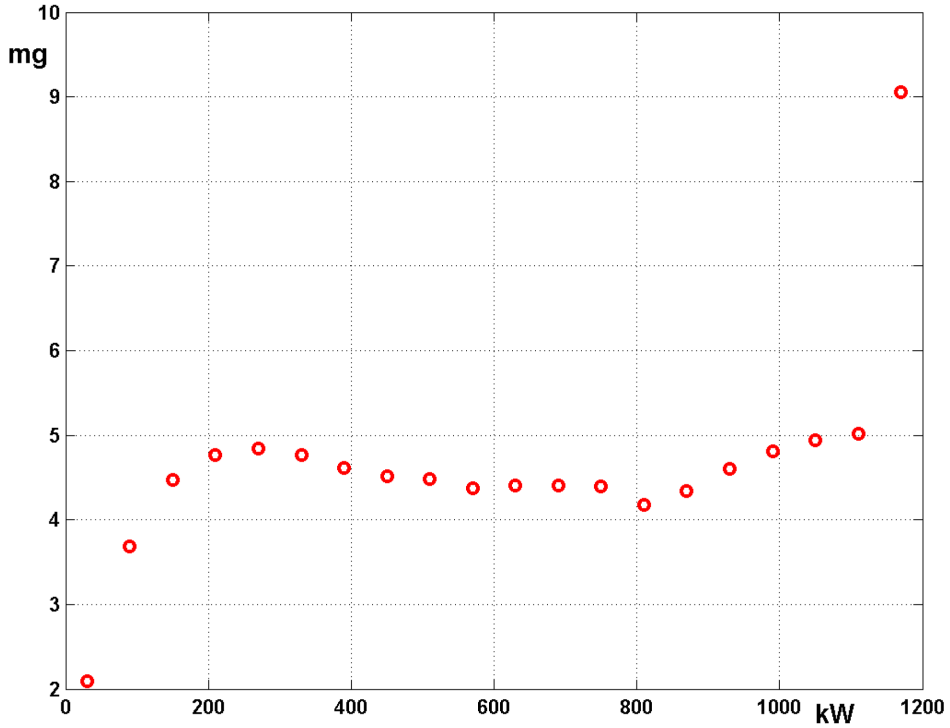


Figure 49: 1p axial nacelle oscillation versus power output

6 Concluding remarks

During the three-and-a-half years of the project, three wind turbines in two wind farms in Denmark and Germany were instrumented with CMS systems and video cameras. Software was written and tested to combine all of the different systems under one hood. A database of high-resolution wind turbine data exists now, and still is extended. The data is relatively unspectacular, since no turbine broke down yet. Algorithms and software for data mining has seen the light, too. The CleverFarm system is thus a kind of extensible SuperSCADA, that can unify the previously diverse user and data interfaces to turbine data, short-term prediction, video streams and condition monitoring. The software could be beneficial for large wind power developers or utilities, having to run many hundred diverse turbines. CleverFarm can also be used for single cases, especially with the data mining options and the short-term forecasting module that positions it uniquely in the market. The market itself has become much more aware of the possibilities of CMS systems since the project started, which correlates nicely with the expertise and experience built up by the partners during the project.

However, at the state the software is now, and with the resources every partner can give to the further development, CleverFarm is mainly going to remain a research platform. To make the tool available to others, we decided to open the platform and make it open source, by making the CleverFarm core system and some example beans downloadable from the CleverFarm homepage. Additionally, this report and the Bean Developers Kit will be public, too.

Seeing what has been done with the resources given, we are largely content with the outcome of the project. We also are glad that the results of the project will not go to waste, since most of it is software that can be handed on for further development. The CleverFarm software is also used and further developed in two follow-up projects, and is used by the partners in-house for additional tasks, so it is not going to die any time soon.

Glossary

CMS	Condition Monitoring
DAM	Dynamic Analysis Module
DP	Data Provider
GUI	Graphical User Interface
IPC	Industrial PC
ISDN	Integrated Services Digital Network
JSP	Java Server Pages
NWP	Numerical Weather Prediction
RAID	Redundant Array of Inexpensive Disks
RPM	Rotations Per Minute
SCADA	Supervisory Control And Data Acquisition
SVM	Structural Vibration Monitoring
VPN	Virtual Private Network
WPS	Wind Power Monitoring System
WTC	Wind Turbine Controller

References

1 Giebel, G., R. Brownsword, G. Kariniotakis: *The State-of-the-Art in Short-Term Prediction of Wind Power – A Literature Overview*. Project report for the ANEMOS project, published on anemos.cma.fr. See also these proceedings.

2 Gellermann, Thomas and Walter, Georg: *Anforderungen an Condition Monitoring Systeme für Windenergieanlagen.* Internal Report No.: 03.01.068. Contact Person: Manfred Jelting, Allianz Versicherungs-AG, Firmen Technische Versicherung, 80802 Munich, Germany

3 *Grundsätze für die Prüfung zur zustandsorientierten Instandhaltung von Windenergieanlagen*. Internal Paper of Gothaer Allgemeine Versicherung AG, Germany

4 Waldl, H.-P., U. de Witt: *Online-Fernüberwachung von Windturbinen mittels Internet und Video*. Poster on the DEWEK Wilhelmshaven, 7-8 June 2000.

5 See www.prediktor.dk or www.risoe.dk/zephyr and the references therein

6 P. Caselitz, J. Giebhardt, M. Mevenkamp: *On-line Fault Detection and Prediction in Wind Energy Converters*. Proceedings of the EWEC '94, Thessaloniki, pp. 623 - 627

7 P. Caselitz, J. Giebhardt, M. Mevenkamp: *Development of a Fault Detection System for Wind Energy Converters*. Proceedings of the EUWEC '96, Göteborg, pp. 1004 - 1007

8 P. Caselitz, J. Giebhardt, M. Mevenkamp, M. Reichardt: *Application of condition monitoring systems in wind energy converter*. Proceedings of the EWEC '97, Dublin, pp. 579 - 582

Mission

To promote an innovative and environmentally sustainable technological development within the areas of energy, industrial technology and bioproduction through research, innovation and advisory services.

Vision

Risø's research **shall extend the boundaries** for the understanding of nature's processes and interactions right down to the molecular nanoscale.

The results obtained shall **set new trends** for the development of sustainable technologies within the fields of energy, industrial technology and biotechnology.

The efforts made **shall benefit** Danish society and lead to the development of new multi-billion industries.