Maaß *et al. EURASIP Journal on Advances in Signal Processing* (2015) 2015:14
DOI 10.1186/s13634-015-0203-4

EURASIP Journal on Advances in Signal Processing
a SpringerOpen Journal

RESEARCH Open Access

Data processing of high-rate low-voltage distribution grid recordings for smart grid monitoring and analysis

Heiko Maaß*, Hüseyin Kemal Cakmak, Felix Bach, Ralf Mikut, Aymen Harrabi, Wolfgang Süß, Wilfried Jakob, Karl-Uwe Stucky, Uwe G Kühnapfel and Veit Hagenmeyer

Abstract

Power networks will change from a rigid hierarchic architecture to dynamic interconnected smart grids. In traditional power grids, the frequency is the controlled quantity to maintain supply and load power balance. Thereby, high rotating mass inertia ensures for stability. In the future, system stability will have to rely more on real-time measurements and sophisticated control, especially when integrating fluctuating renewable power sources or high-load consumers like electrical vehicles to the low-voltage distribution grid.

In the present contribution, we describe a data processing network for the in-house developed low-voltage, high-rate measurement devices called electrical data recorder (EDR). These capture units are capable of sending the full high-rate acquisition data for permanent storage in a large-scale database. The EDR network is specifically designed to serve for reliable and secured transport of large data, live performance monitoring, and deep data mining. We integrate dedicated different interfaces for statistical evaluation, big data queries, comparative analysis, and data integrity tests in order to provide a wide range of useful post-processing methods for smart grid analysis.

We implemented the developed EDR network architecture for high-rate measurement data processing and management at different locations in the power grid of our Institute. The system runs stable and successfully collects data since several years. The results of the implemented evaluation functionalities show the feasibility of the implemented methods for signal processing, in view of enhanced smart grid operation.

Keywords: Power system analysis; Smart grids; Supply grid monitoring; Low-voltage distribution network recordings; Big data analysis; Energy data visualization

1 Introduction

The electrical supply network is changing into more locally controlled smart grids with interconnected measurements and advanced management methods [1]. By integrating more and more fluctuating energy sources as solar or wind power into the network at the end-user level, the power flow direction is affected, which influences stability and voltage quality in the local distribution grid [2]. Strong power consuming processes like electrical vehicle charging will result in feedback effects on the supply circuits in smart grids [3]. Additionally, the increased use of nonlinear loads and control means causes higher perturbations in the electrical power grid, which have negative effects on the connected consuming devices [4].

Therefore, the control in low-voltage distribution grids is currently gaining more importance [5]. In order to ensure quality and reliability of the supply, the awareness of the state of the system components is necessary. Distributed and synchronized voltage measurements will be a prerequisite for advanced control possibilities [6-8]. However, if these measurements reflect the energy consumption or production billing information, they are subject to privacy protection. For this reason, these data must be treated securely and confidentially [9].

In order to detect and counteract grid instabilities in the medium- and high-voltage net, power control stations use supervisory control and data acquisition systems (SCADA) together with energy management

* Correspondence: heiko.maass@kit.edu
Karlsruhe Institute of Technology, Kaiserstr. 12, 76131 Karlsruhe, Germany

systems (EMS). The update rate of SCADA and EMS is typically low, which does not meet the performance demands of a dynamic control of low-voltage distribution grids [10].

State-of-the-art monitoring systems for dynamic state estimation are currently encouraged by phasor measurement units (PMUs), which measure current, voltage, frequency, and phase at selected locations. PMUs are capable of synchronous acquisition and provide up to 50/60 datasets per second [11]. The calculated electrical characteristics are transferred to a Phasor Data Concentrator (PDC) for selection and further aggregation or even forwarded to grid protection and control units. Proprietary systems like ABB PSGuard PSG830/PSG850, Siemens Siguard Phasor Data Processor (PDP), Schweitzer Engineering Laboratory SEL-3373/SEL-3378, or Alstom Grid S800 substation Phasor Data Concentrator (sPDC) are used for medium- and high-voltage grid monitoring today [12]. However, network operators are not allowed to provide full data access for academic research due to privacy protection. Furthermore, deviations between specifications of various PMU types reduce the comparability between different manufacturers, especially at off-nominal frequencies [13]. The Open Source Phasor Data Concentrator (OpenPDC) [14] is an interesting license-free open-source software project for streaming typical PMU data time series in real-time. Unfortunately, this package is not intended for high-rate raw captures or for encrypted data transfer, yet.

There are promising approaches for measurement networks for low-voltage grids like the already implemented *ad hoc* phasor measurement network "WAMSTER," which was developed by Studio Elektronike Rijeka (STER) and is used by the Croatian Academic Research Wide Area Monitoring System (CARWAMS) [15]. The system provides a live data display, the comparison of historical and currently measured PMU data and even event-based triggering of other web-enabled devices. But there is no possibility to gain access to the raw data, which is discarded before transmission. A wireless sensor network for low-voltage grids is proposed from the University of Berkeley for voltages and currents [16]. The authors developed a Scalable Power Observation Tool (SPOT) which is capable of high-rate low-power monitoring and which conducts the transfer via wireless network. However, the system provides only power values as output, which are calculated from current and voltage readings integrated over time.

Hence, the currently available PMUs and other low-voltage measurement devices only provide aggregated information, mostly according to the specifications of the standard for electromagnetic compatibility (EMC) IEC 61000-4-30 [17]. In contrast to this, we intend to make the full high-rate captured transient data permanently available for research and evaluation. We assume valuable information in the transient waveform in addition to the extracted electrical characteristic values [18]. High-frequency disturbances like transient oscillations or switching processes are only visible in the full waveform. Furthermore, considering fault propagation evaluation, the timing analysis of the waveform distortion is essential. Interval aggregation is useful for data reduction and comparison, but in our vision, deep data analysis should be enabled considering full observation possibilities. As a consequence, existing frameworks for data management are not applicable. Therefore, we develop new means for acquisition, processing, transport, storage, retrieval, and evaluation of large time series. We separate the acquisition data transport paths from monitoring and data evaluation in order to prevent the capturing system from congestion by the massive measurement data flow. In Figure 1, this concept is shown schematically.

EDR acquisition system consists of the electrical data recorders (EDR) as the measurement data producing units, which especially enable for high-rate time series acquisition and full raw data transmission, as well as the EDR supervision network and appropriate methods for data storage [19]. The EDR evaluation system comprises methods and tools for data retrieval and analysis and is independent on the acquisition part. The complete EDR system is currently deployed at the Karlsruhe Institute of Technology (KIT) with three EDR capturing devices. Two of them are located in the island-like electrical distribution grid of the KIT Campus North, and one is located approximately 12 km away at KIT Campus South, which is close to the urban supply grid of the city of Karlsruhe.

The first of the following two main chapters describes our special acquisition approach and places focus on the processing of high-rate data. The EDR evaluation system is divided into five different methods for accessing and analyzing the large EDR data amount from the storage, which we explain in main chapter 3.

2 EDR acquisition system

The acquisition system is intended to reliably handle the measured data, preferably close to real-time. The EDR capturing device provides both, the full acquired waveform data and the extracted electrical characteristics like frequency, phase, effective values, and harmonics [20]. A short overview is given in sub-chapter 2.1 together with a description of the data processing. In order to allow for full flexibility concerning protocol and service design as well as for extended control possibilities, we implemented the specialized EDR-Broker-EDR-Customer network as a multiservice supervision system, which we present in sub-chapter 2.2. We develop the EDR-Netpipe software (see sub-chapter 2.3), which acts as a

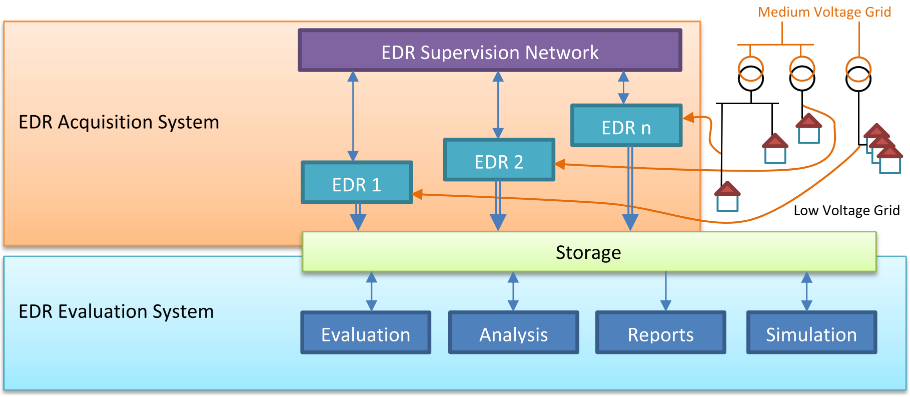


Figure 1 Simplified schematic drawing of the data flow concept. Legend 1: The double lined arrows show the massive measurement data transfer towards the storage. The single-lined arrows indicate minor data flow together with the transport direction. The curved horizontal lines to the EDR devices are the measurement connections from the power grid. This illustration is a simplified representation. The single parts are shown in the respective sub-chapters in more detail.

file-based transport connector from the EDR capturing device to the large-scale database via a dedicated web service. This web service is running on multiple servers in order to ensure for connectivity and scalability. We permanently keep the whole data in the database for long-term and deep analysis as well as for the comparison of information extraction methods or estimation algorithms. The so-called generic data services (GDS) are one part of the storage system and are designed as a file system-independent interface for high-performance data access and are used as an interface to the large-scale data storage and for data retrieval. The storage methods are outlined in sub-chapter 2.4. Security considerations are addressed in sub-chapter 2.5, and results of the acquisition system part are given in sub-chapter 2.6.

In Figure 2, a more detailed view of the EDR acquisition system is shown illustrating the software interaction and data flow. The following sub-chapters refer to this drawing.

2.1 EDR capturing device

Existing high-rate capturing devices like network analyzers or data loggers do not provide appropriate interfaces to transfer the high-rate transient captures to an external storage continuously without interruptions. In order to gain comprehensive access to the measured data, we developed the EDR capturing device for synchronized low-voltage time series at high rate. This recorder unit is a hardware device and currently consists of an acquisition interface to the power grid, a notebook

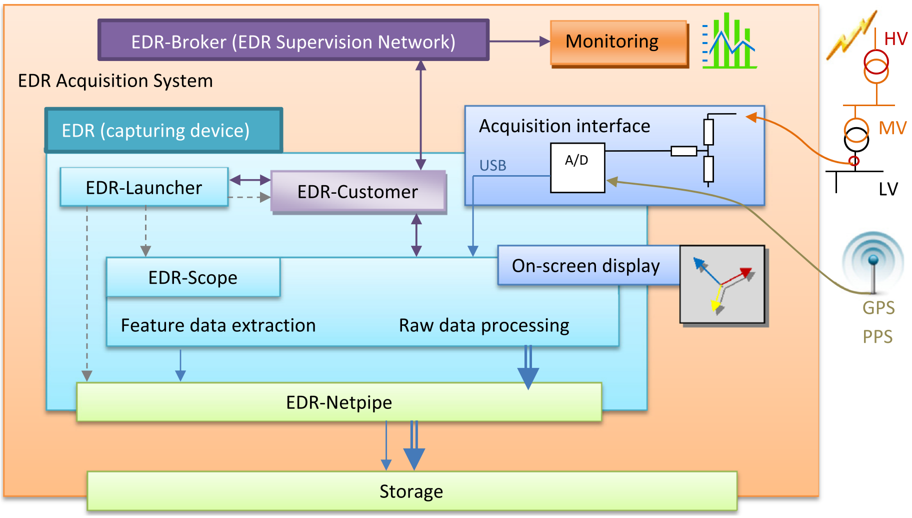


Figure 2 Schematic drawing of the EDR acquisition system.

as the computing hardware and a software package for data preprocessing. The low-voltage (LV) energy supply grid is connected by single-phase or three-phase plugs for voltage measurement and by Rogowski coils placed around the conductors for current measurement. In the last years, we intensively tested the sensing system and proposed methods for improved accuracy [21]. The hardware assembly and the entire source code of all EDR applications were developed at KIT. In this paper, we focus on the data processing aspects for the EDR capturing device.

2.1.1 Computing hardware

At the present time, we use low-cost notebooks as the local processing units. Using standard notebooks is advantageous in this development stage, because we have local display and manual interaction possibilities, local storage, and a power reserve by the rechargeable battery in the device. The actual units have Intel Pentium DualCore 2.2-GHz processors, 500-GB hard disk drive, 4-GiB main memory, USB 2.0, and wired internet connections. In a fully productive application, the processor usage periodically varies between 5% and 35% with a mean of 10%, which is by far an acceptable continuous computing load. It would be possible to reduce hardware cost by using highly integrated devices without display and less computing power, but it is more convenient to provide an on-site display and sufficient computation reserve in the development stage.

2.1.2 Acquisition interface

Precise time stamp detection is a very important prerequisite for the conversion from captured data to characteristic information and for the later comparison of values measured at different locations. Thus, like other wide area monitoring device manufactures, we rely on the highly accurate pulse per second (PPS) signal (±1 μs), which is derived from a GPS receiver. We utilize a commercially available A/D converter, which can acquire up to eight channels simultaneously at 16-bit resolution up to 25-k samples per second. One input channel is connected to the PPS signal from the receiver to link each captured sample to the absolute time when it was measured. In a full observing configuration, the seven remaining channels are connected to three phase voltages and three phase currents plus neutral conductor. Using a typical rate of 12.8 kHz, approximately 256 samples of each channel are acquired per period of a 50-Hz input signal. We connect the A/D converter directly to the voltage phases via a voltage divider. We do not use a transducer for voltage measurements which could modify the waveform in high or low frequencies. For current conversion, we use Rogowski coils from Fluke [22], that allow a −3 dB measuring bandwidth from 10 Hz to 10 kHz.

2.1.3 EDR software

As illustrated in Figure 2, the EDR software package includes the main software EDR-Scope for the acquisition process, the feature extraction, the display for on-site monitoring, and the preparation of raw and feature data files. The software EDR-Customer is the client part of the supervision system, described in sub-chapter 2.2. EDR-Launcher is responsible for the automatic startup and observation of the vividness of the running software parts EDR-Netpipe, EDR-Scope, and EDR-Customer (dashed arrows in Figure 2). If the execution halts unexpectedly, the process is killed and restarted. Also, updates and the automatic startup process are managed with this software. EDR-Netpipe is the data transfer management software (see sub-chapter 2.3).

We are developing all software elements in-house, and therefore, full source code in C/C++ is available. Currently, EDRs are set up with Windows 7 64-bit operating systems, but the EDR software is designed for cross-platform use and could be ported to other system environments.

2.1.4 Raw data processing

The EDR raw data processing starts with the continuous bulk transfer of all channels from the A/D converter via the USB connection. The full data stream is then divided into single channel series and into second-data blocks between two PPS channel rise events. All 1-s channel blocks are written to an XML-formatted file and are coded as a base64 16-bit value stream without any further pre-calculation and without any compression. Acquisition metadata are stored together with the corresponding channel raw time series in the same XML section. These metadata consist of the acquisition date and time, the calibration values, some general properties, and GPS coordinates of the installed EDR capturing device. We combine the 60-s sections of 1 min to one XML file and name it using the description of the EDR capturing device and the respective minute timestamp.

The amount of produced data mainly depends on the sample rate and the number of captured channels. Using the Equation 1, the acquisition word length $w = 2$ bytes (16 bit) at $p = 7$ channels, the base64 coding factor $c = 4/3$, and $m = 1{,}500$ bytes of additional metadata per second, we obtain $b \approx 240{,}433$ bytes per second at a sample rate $r = 12.8$ kHz.

$$b = w * p * c * r + m \tag{1}$$

This amount sums up to 19.35 GiB per EDR and day when multiplied by 86,400 s/day.

We decided for file-based data packages because in the case of transmission failure, the long-term buffering is greatly simplified by storing the files on the local hard disk, and because the data post-management and evaluation is made much easier by just considering the needed files of a certain time period. We currently do not apply any data reduction technique to keep the storage performance high and to keep the data access complexity low.

2.1.5 Feature data extraction

In order to provide electrical characteristics for evaluation already at the EDR capturing unit and to display values that are directly comparable to other monitoring systems, we perform feature extraction methods together with the raw data processing at the same time within the EDR capturing unit. We conduct the pre-calculation of the frequency, the phase, and the power quality characteristics defined by EN 50160 [23] for each voltage or current channel. The recurrent calculation of the features is performed without any further processing step-like outlier removal or feature sectioning. The resulting feature time series are used as the input database for the evaluation methods that are presented in chapter 3. We applied the assessment methods of the standard EN 61000-4-30. The device was successfully tested and complies with class A for frequency estimation and uncertainties, and it was found to meet class S conditions for the voltage range and harmonics. The determination of the harmonics could reach the requirements of class A when the typical sampling rate is increased to 25 kHz [21].

A complete list of extracted features together with the mathematical description is presented in Table 1. In the table, we use $x(t)$ as the generalized representation of the instantaneous input value of current $I(t)$ or voltage $V(t)$ at the time interval t, respectively. The beginning time of the evaluation interval is t_0. For example, the root mean square of the current I_{RMS} is calculated by the formula in the row root mean square (RMS) inserting the instantaneous current values $I(t)$ as the input values $x(t)$, and the root mean square of the voltage V_{RMS} is calculated with the voltage values $V(t)$ as the input. T is the evaluation period length, which is ten times the single waveform length for the evaluation according to the standard EN 61000-4-30 for 50-Hz networks. The harmonic proportions $H(i)$ are derived according to EN61000-4-7 [24] from a discrete Fourier transform (DFT) with a window length of 1,024 values, which were time scale interpolated from the evaluation interval, e.g., from approximately 2,560 samples of 10 periods when using the 12.8-kHz sampling rate. We use the integral sign in the formulas instead of the sigma sign to indicate the sample fraction consideration at the evaluation borders (T is not an integer multiple of a single sample duration, and t_0 is independent on single sample begin times). Period length T and the period borders are calculated from two ascending voltage zero crossings of a low-pass filtered signal. The finite response (FIR) low-pass filter has a cutoff frequency of 100 Hz.

Table 1 Feature extraction formulas used for the EDR

Value	Formula
Frequency	$f = \frac{1}{T}$
Average rectified value (ARV)	$X_{\mathrm{ARV}} = \frac{1}{T}\int_{t_0}^{t_0+T} \lvert x(t)\rvert dt$
Root mean square (RMS)	$X_{\mathrm{RMS}} = \sqrt{\frac{1}{T}\int_{t_0}^{t_0+T} x(t)^2 dt}$
Offset	$X_{\mathrm{offset}} = \frac{1}{T}\int_{t_0}^{t_0+T} x(t)dt$
Maximum value	$X_{\max} = \max_{\{t_0\,..t_0+T\}}(x(t))$
Minimum value	$X_{\min} = \min_{\{t_0\,..t_0+T\}}(x(t))$
Amplitude	$\hat{X} = \frac{(X_{\max}-X_{\min})}{2}$
Crest factor	$C = \frac{X_{\max}}{X_{\mathrm{RMS}}}$
Total harmonic distortion	$\mathrm{THD} = \frac{\sqrt{(H_2)^2+(H_3)^2+(H_4)^2+\ldots+(H_n)^2}}{H_1}$
Phase	$\varphi = \frac{360}{T} * (t_{\max}-t_0)$
Harmonics 0 to 50	$H_i = \mathrm{DFT}_{8192}\rvert_{i=0..50}$
Active power*	$P = \frac{1}{T}\int_{t_0}^{t_0+T} V(t) * I(t)dt$
Apparent power*	$S = V_{\mathrm{RMS}} * I_{\mathrm{RMS}}$
Reactive power*	$Q = \sqrt{S^2-P^2}$

Values marked with an asterisk are only used for combined voltage and current configurations.

The voltage input range is limited by an overvoltage protection circuit to a maximum of 280 V_{RMS}. This is a restriction compared to the class A voltage range definition in the standard EN 61000-4-30, which requires 10% to 150% of the supply voltage. Therefore, EDR complies with class S as the input range allows 10% to 121% of the supply voltage. Depending on the expected range of the current measurement, the Rogowski-coil preamplifier switch can be set to 30 A, 300 A, or 3,000 A. The standard EN 61000-4-30 does not apply for current measurements.

We determined the voltage estimation errors by using the ACS-800-PS voltage reference source from HBS [25] and a calibrated Tektronix AFG 3022B 14-bit arbitrary signal generator. We found that the frequency determination error is lower than ±50 μHz, which is far below the required ±10-mHz limit. The voltage and current channel measurements show an error of ±0.05% in the processing unit, but due to using of Rogowski coils as the sensors, currents are captured with a higher uncertainty of 1%. Using the DFT with a window length of 8,192, we could prove that the EDR is capable of determining harmonics up to the 30th with a maximum deviation of 5% and up to the 50th with a maximum deviation of 15%. When increasing the sampling rate from 12.8 to 25 kHz experimentally, we could even reach a 5% maximum deviation up to the 50th harmonics, which would comply with the class A definition of the standard EN 61000-4-30, but we do not intend to produce the double amount of data permanently. We could also show the effect of deviation reduction by higher sampling rate in a MATLAB simulation of the feature extraction process. We consider the A/D conversion finite step height and conversion inaccuracies to and from the DFT as responsible for this inconsistency with the Shannon sampling theorem.

2.1.6 On-screen display

In an experimental setup, the visual inspection for on-site plausibility checks is a valuable feature. Therefore, the EDR software optionally provides different types for visualization and can show the just captured raw waveforms as well as the history of the characteristics as graphical curve plots. In Figures 3 and 4, a selection of visual data representations at the EDR capturing device is presented. Figure 3 shows the typical three-phase phasor diagram including harmonics and phase-shift information. Using this display, a system operator gathers visual information on the actual phasor and on the change of the phasor together with the history by the curve plot of the last 100 phasor endpoints in phase color (red, yellow, blue). The harmonics bar charts in the upper right quadrant give an overview on the actual waveform distortion and the proportion to the EN 50160 limits, which are indicated by the gray blocks in the chart background, in the same display. The current-to-voltage phase shifts and the phase-to-phase shifts are visualized in a tachometer-like illustration in the upper left quadrant for each phase. The deviation between the measured frequency and the reference frequency is represented by the vertical position of a green frame while the height of the frame shows the actual frequency measurement uncertainty. Figure 4 shows combined plots of current and voltage raw data. Both display types from Figure 4 can be used for the characterization and the identification of connected loads.

All visualizations are refreshed each second at the capturing device for local monitoring purposes. During productive use of the EDR, the graphical display is not necessary at the capturing device and the visualizations are switched off. All presented data are recorded from our Institute at KIT.

In the standard EN 61000-4-30, ten periods are considered as the base evaluation interval for 50-Hz supply grids. From these results, real-time clock synchronized 10-min average values have to be calculated, by accepting overlapping base intervals. We consider this method

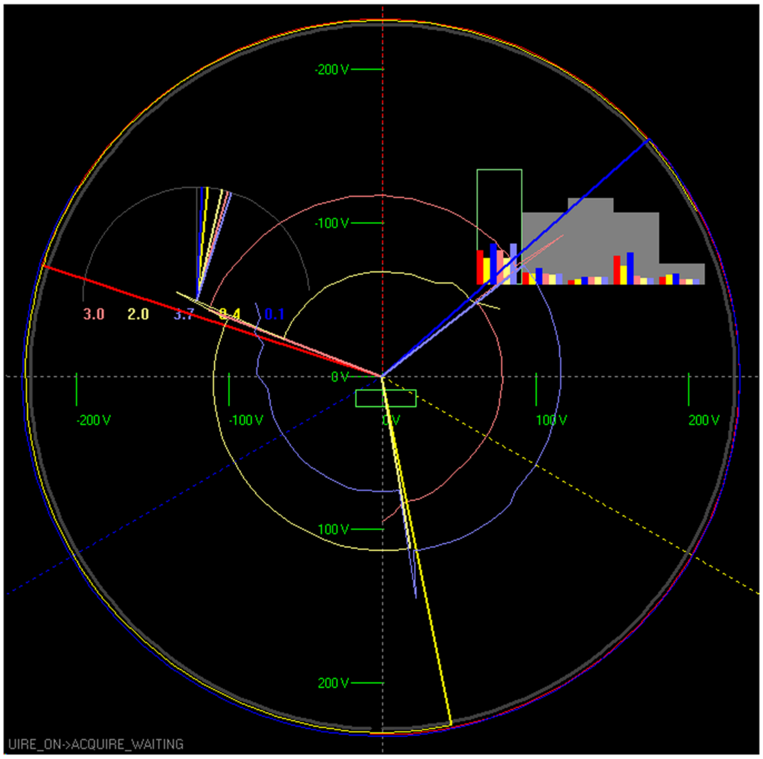


Figure 3 On-site phasor visualization at the EDR capturing device.

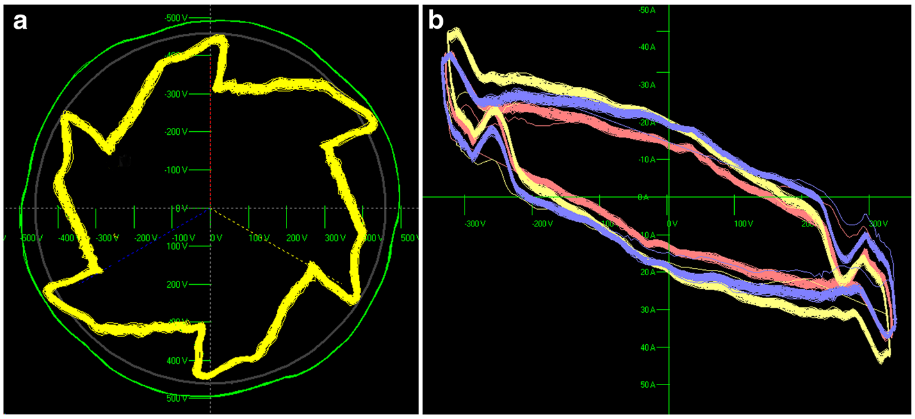


Figure 4 EDR-Scope on-site current characteristics display. Legend 4: The left image depicts the 1-s space phasor representation of currents (in yellow) and voltages (in green) of all phases. The phase directions are indicated by dotted lines in phase color. The right image shows the current-voltage diagram of 1 s.

important for inter-manufacturer result comparability because it is the standard. However, this rule considers measured samples twice at the 10-min border by synchronizing the first base interval to each border. There is no information, to what extent the base interval values were derived from overlapped samples at the 10-min border and which samples are relevant for both averaging procedures. The error is small, but at each 10-min border, discontinuities may occur. For a biunique evaluation, we provide fixed-time interval results (actually 1 s) where all full periods are taken into account which are either completely inside the interval or intersecting its begin time. With this method, all samples are uniquely assigned to one result and vice versa without overlapping. For real-time applications, the fixed-time duration and the update rate can be scaled towards sub-second intervals.

For simple and universal further processing, each feature set of the calculated electrical characteristics is stored as one row of double values in an easily readable tabular text format. The rows are time stamp indexed and saved in one character separated value (csv) formatted file each day. One csv file is created per captured channel. Each reaches a size of approximately 23 MB in 24 h. The overall data processing and local storage including raw data handling takes less than 15 ms on the actual computing hardware, which is short when compared to an update interval of 1 s.

2.2 EDR supervision network

For the purpose of long-term evaluation, interruptions of registered time series should be avoided. The data transport is performed via LAN or WLAN, which can be congested or broken on the one hand. On the other hand, we enable paths over public internet lines and have to protect the system from intrusion attacks, see sub-chapter 2.5. The EDR capturing devices are designed to run continuously but due to the internet connection, they have to be protected by operating system and virus

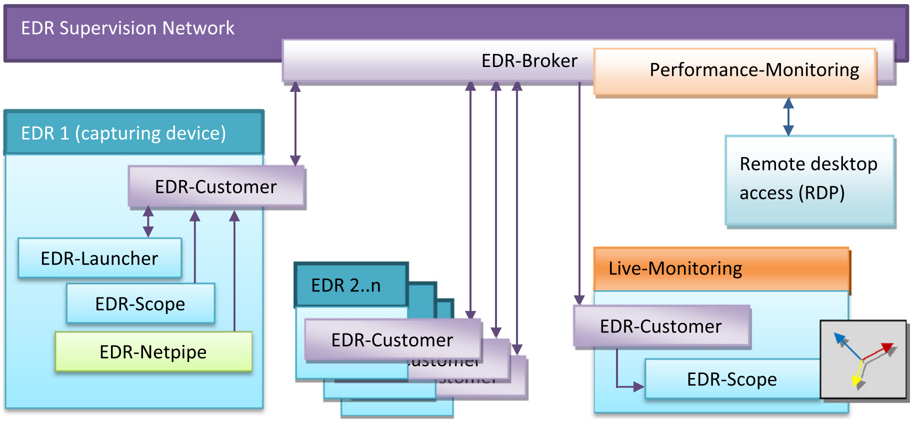


Figure 5 Components and data flow in the EDR supervision network.

scan updates regularly, which can cause shutdown breaks. We try to reduce the impact of the interruptions by only allowing for controlled updates at known scheduled times.

The EDR supervision network is primarily intended for ensuring continuous operation of the EDR capturing devices. Like Phasor Data Concentrators, we gain awareness of the acquisition system status and provide device managing methods. Due to the separate handling of the captured data transfer (see sub-chapter 2.3), this supervision system is mainly responsible for observing the EDR performance.

We provide multiservice interconnectivity possibilities by grid middleware using a broker and customer architecture and an in-house developed protocol based on TCP/IP, which is already used for other KIT projects [26]. By applying this technology, we enable communication capabilities for different protocols and services like client-to-server and client-to-client functionalities.

The detail view of the EDR supervision network in Figure 5 shows the interacting components and the data flow.

Each client runs an instance of the multipurpose EDR-Customer software. The software logs on to the in-house EDR-Broker server, which is accessible by operators over the internet. All EDR capturing devices run an EDR-Customer with a preregistered name and password and offer device and transport monitoring to the supervising broker. The EDR-Broker can be accessed via Remote Desktop Protocol (RDP) in the current configuration. In the on-screen user interface, the registered devices are displayed in a list and can be controlled by user interaction.

The displayed list of registered EDR capturing devices contains the information fields:

- name and IP address,
- CPU, memory, and network usage,
- Windows operating system update status,
- EDR-Netpipe status: number and size of buffered files, transfer rate (see sub-chapter 2.3),
- EDR capturing device status: sample rate, GPS-status, current time,
- last occurred errors at EDR, and
- remaining local hard disk capacity.

The EDR-Broker has control of the EDR capturing devices on:

- software updates and full start-stop control (EDR-Scope, EDR-Netpipe, EDR-Customer),
- comprehensive Windows Updates control, and
- GDS-Web Service adapter configuration (see sub-chapter 2.4).

Additional interconnection services between EDR-customers are:

- VNC-Tunneling between EDR-Customers and
- live measurement data monitoring.

The live measurement data monitoring is comparable to data concentration possibilities, which are available with PDCs. Characteristics datasets from EDRs can be transferred to a central monitoring device running another EDR-Customer client and an instance of the EDR-Scope software, handling the network stream as the measured input data and thus enabling for the same remote display possibilities as on the EDR capturing device.

2.3 EDR-Netpipe

The EDR device continuously produces files from the acquired data (see sub-chapter 2.1.4). In the case of regular network condition, the finished files should be transferred to the data storage immediately. In the case of network congestion or receiving service failure, local buffering and subsequent transport must be ensured as soon as possible after the delay. After confirmation of successful transfer to the storage, the files have to be deleted from the local disk because of limited storage capacity. If 20 GB are produced a day, a disk capacity of 500 GB would last approximately 3 to 4 weeks for intermediate buffering. For the local management of data files and the reliable transport to the storage, we develop the multipurpose file transport interface EDR-Netpipe, which is running on each EDR capturing device.

The EDR capturing device copies entirely written files to a selected output folder, which is observed by the EDR-Netpipe software. If a new file is detected, the software selects an IP address from a list of available web servers acting as GDS front end of the storage (see Figure 6) and establishes a socket connection. The EDR file is sent to the server as a base64 encoded data block together with a message-digest algorithm 5 (MD5) [27] check hash by using the Simple Object Access Protocol (SOAP) communication protocol [28]. If the server does not confirm a successful storage within 2 min, the same transmission procedure is tried on another web server. The packet retransmission on failure continues up to 100 times and reports a permanent transmission error afterwards. The file remains on the capturing device. A new retransmission process is restarted when connected to the EDR supervision network. If a single server fails more than ten times, the server is skipped for 2 h. EDR-Netpipe allows up to four transmissions to one server at the same time. By applying these simple rules, automated load balancing is accomplished in the case of many files waiting for transport or single server

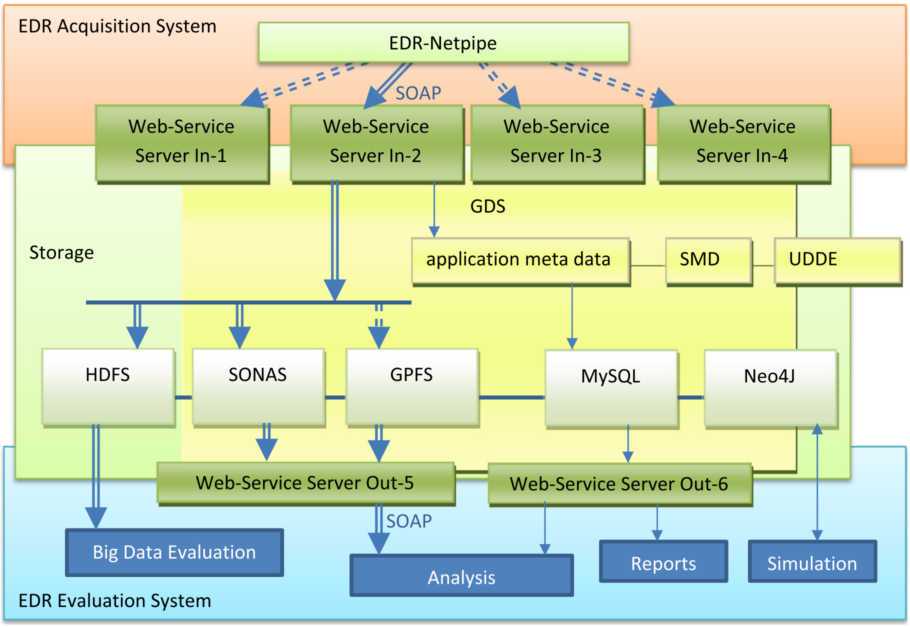


Figure 6 Illustration of the storage access. Legend 6: The dashed lines from EDR-Netpipe to the Web Service interfaces refer to the load balancing, which has been described in sub-chapter 2.3.

unavailability. Highly performant server connections will be used more often than servers with a low accepting data rate. All transmission errors are logged for any later failure search.

Depending on the producing EDR configuration, the minimum required data transport rate is 0.8 Mbit/s for voltage only recordings and 1.83 Mbit/s for voltage and current data captures, calculated from the produced amount per day. In the installation at KIT, we observe an average overall transport rate of 28 Mbit/s from the EDR to the GDS, calculated by EDR-Netpipe output amount per time of transfer completion.

2.4 GDS storage and retrieval

The EDR acquisition system is connected to GDS, which are data management modules based on object- and service-oriented programming models. Arbitrary data formats are handled with minimal configuration effort. First issues on GDS have been introduced in [18] and [19]. In Figure 6, the data flow in the GDS storage access is drawn schematically.

Application data elements are converted to data objects with well-defined properties according to our internal programming rules that support a uniform description of respective classes by structural metadata (SMD) and promote language-independent implementations. The approach comprises relevant object-oriented concepts like classes, class properties and their visibility, generalization, or object relations. The SMD describe the classes from which GDS data objects are instantiated. Applications that apply for GDS data management have to declare the format of their data elements as SMD by means of a Universal Data Description editor (UDDE). Data access is realized by referring to so-called application metadata (AMD), which identify data elements respectively data objects in GDS. AMD are analogous to key attributes for database entries. GDS also creates a universal object identifier, which is the main GDS-internal handle for data objects besides AMD.

At present, there exist special services for transferring data to and from the GDS for EDR measurement data. Currently, different serialization solutions are implemented based on JavaScript Object Notation (JSON), Extensible Markup Language (XML), and a self-developed interface based on SMD that all will allow an application-independent method for delivering and retrieving data, e.g., for further EDR data processing components. The overall access to GDS is done by addressing Web Services via SOAP (see also sub-chapter 2.3 EDR-Netpipe).

GDS is also designed to be independent of the used storage system. Concerning EDR, data are stored in the Large Scale Data Facility (LSDF) at KIT [29] which contains a Hadoop File System (HDFS) for big data analyses. General Parallel File System (GPFS) and a Scale Out Network Attached Storage (SONAS) for band-secured long-term archiving. GDS performs direct staging to HDFS for the most recent data to enable distributed data analyses in Hadoop. Therefore, we permanently keep the data of the last 12 months synchronized in an HDFS folder.

Up to now, two database systems are used for metadata and model data storage: MySQL and Neo4j. The latter was chosen for model data storage, as it is a graph-based database system which is likely to hold the topological graph structure of power grid models. MySQL is selected as the database for all GDS internal metadata storage, i.e., SMD, AMD, and object-IDs, as well as the so-called organizational metadata (OMD), which comprise localization and security data, see also sub-chapter 2.5. Data objects with identical security requirements and access rights are grouped to object sets. There is a GDS-specific user management based on these sets. Access rights and ownerships are related to the sets only and not to individual data objects. Every set belongs to exactly one user. Beside the access rights of owners, there may be rights of user groups in addition. Details can be found in [30].

GDS is completely implemented in Java, which offers hardware independence, and all of its classes and services shall be obeying to regulations of OPM, which defines AMD and the structure of SMD. The objects which perform the data management, the metadata, and the user data objects that are managed are seamlessly integrated into the GDS system according to our programming rules. Thus, future components with different user data structures can easily be adapted to GDS and benefit from a growing portfolio of data storing and access modules.

Ongoing research deals with the further development of the GDS storage interface, like for example staging mechanisms together with comfortable user interfaces. To address the challenge of accessing vast amounts of data, it is planned to transform GDS to an ontology-based data access system (OBDA) to create a flexible information platform around the future energy system.

2.5 Security and privacy considerations

As far as the measured data are related to single person directly, to office rooms belonging to certain employees, or to households, they are subject of privacy protection. To illustrate the potential of misuse, we refer for example to energy consumption profiles in the hand of burglars, who plan their next housebreaking. Furthermore, detailed data on the grid status can be used for both, maintaining its stability or attacking it with the aim of a blackout. The motivation for the latter may be blackmail, terrorist attacks, or of military nature. In addition to the necessity of the protection of the data against unauthorized reading, its integrity and authenticity must be ensured so that the information is of worth for service providers, who are, e.g., in charge of the stability of the grid. For the comparable case of smart metering, a detailed requirement document was prepared in Germany by the Federal Office for Information Security [31], which serves as a basis for further developed concepts [32].

The security demands of our EDR net differ in so far as the number of types of participants is limited, and there is no need for a concept open to future additional service providers with differing access needs and rights as outlined in [31,32]. In [30], we defined a set of requirements for a large-scale metering system and introduced a concept based on a virtual private network (VPN) for the EDR-Netpipe part of the metering system in conjunction with the GDS and its storage backend. Its first implementation based on Cisco hardware was presented in the same paper. Alternatively, we are also studying a solution based on OpenVPN [33] and a concept based on secured data transport using TLS [34]. Regardless of which approach is to be implemented in the end, the communication of EDR supervision network will be encrypted as an additional level of security to the proprietary protocol mentioned in sub-chapter 2.2.

2.6 Acquisition system results

The acquisition system is steadily running since February 2012. One EDR capturing device is recording three-phase voltage measurements at KIT Campus North (CN) from the beginning. Up to five devices were temporarily installed at different nodes on the CN in the meantime. Currently, three EDR capturing devices are recording continuously since February 2014: one voltage-only EDR and one voltage and current EDR at a supply node of an institute building on KIT CN plus one voltage-only EDR on KIT Campus South (CS). The produced amounts of data are 8.4 GiB per day for three-phase voltage-only measurements and 19.35 GiB for three-phase voltage and four channel current capturing devices. Until today, we collected about 30-TB high-rate voltage and current measurement data. Due to short-term network connection failures, about 5% of the packets are retransmitted, which results in an actual average additional network load of 0.83 Mibit/s per voltage-only EDR device and 1.92 Mibit/s for a voltage and current EDR device towards the database.

At present, the integrity of the transmitted data is protected by the already mentioned MD5 check sum and the privacy of the measured data by the use of pseudonyms for the exact measurement points within the grid. On the other hand, the saved GPS coordinates reveal the location of a measurement, although the exact position is not provided in buildings with more than one floor, because the altitude information from the GPS receiver is not saved. Additionally, the precision of the two remaining GPS coordinates is 5 m at best. To avoid any devaluation of the pseudonymization, we will omit all location coordinates from the data files in the future. Privacy concerns depend strongly on what can be derived from the data in question. At present, it is hard or even impossible to regain consumer behavior from pure voltage measurements, e.g., if somebody is in a house or not and which devices are switched on. For this, current or consumption measurements are required today. So, at present, the measured voltage values are no critical personal data. If the planned and ongoing research is successful, this may change in the future and the operation of single consumers could become identifiable by precise voltage captures only. Therefore, a reliable pseudonymization is required also for pure voltage measurements in the near future.

Currently, all data managed by GDS are transferred via a 10-Gbit connection to the LSDF hosted by KIT, where all the mentioned file systems are available. Because of temporary unavailabilities of the LSDF, we provided a GDS-controlled buffer system, which uses a 4-TB SAS-RAID storage system. Furthermore, there is temporal buffer capacity directly attached to the EDR devices, so that they can manage a loss of connectivity to the GDS for up to 2 weeks.

3 EDR evaluation system

One new aspect of our approach is to search for additional valuable information from preserved long-term recordings of high-rate time series. Because of the vast amount of stored values in the database, standard computing architecture is not able to handle these large data. Therefore, we developed interfaces for data retrieval of selected intervals concerning particular different evaluation purposes. We demonstrate the practicability of the interfaces by showing exemplary evaluation results in this chapter. In the following, we present five different methods that we prepared for data exploration, processing, and analysis.

– Optimized Hadoop-based *big data queries* on the raw and feature data using "*S*calable *Ti*me *Se*ries *A*nalysis *Q*ueries" (STiSeAQ): Large data search and processing requests are formulated using the Pig Latin [35] language together with custom analysis code implemented in Java. Processing steps are automatically optimized to Map-Reduce code, which is distributed to all Hadoop worker nodes. The code is executed at the nodes where the target data is stored. The results of all nodes are collected and stored as metadata (see sub-chapter 3.1).

– Sophisticated time series *exploration* with statistics evaluation, visual *interpretation*, and structural analysis by using our interactive software "Visual Analysis of Time Series" (*ViAT*): Predefined, adjustable data interpretation schemes are applied on the feature data. The results are visualized graphically (sub-chapter 3.2).

– *e*ASiMoV (*e*lectrical grid *A*nalysis, *Si*mulation, *Mo*deling, and *V*isualization) *Data integrity tests* with interactive visual feedback using *e*TSAnalyzer and *e*MetaVis in the BReSoC framework (sub-chapter 3.3): overview on existing and missing data in the database and visual representation of value ranges.

– *Visualization*, *Simulation*, *Modeling* tool (*e*ASiMoV): graphical interface to third party simulation software and simulation-package-independent graphical modeling and integration of EDR measurement data in simulation processes (see [36]).

– File-based *MATLAB* import interface and using toolbox *Gait-CAD* [37] for feature data analysis and comparison (sub-chapter 3.3): Any MATLAB calculation can be performed on the imported measurement data, and time series analysis is supported by using the Gait-CAD toolbox.

The illustration in Figure 7 shows the data processing possibilities from the storage via the respective data retrieval interface to the possible evaluation application. The methods in the list above are depicted in the figure from left to right.

3.1 Big data queries

One important information retrieval task is the search for patterns in the huge collection of time series data. This procedure is very I/O intensive, because we need to read the full respective data files for computing occurrences or similarities to the given pattern. Finding the most similar time series, matching a search pattern in millions of large files could take unacceptable duration

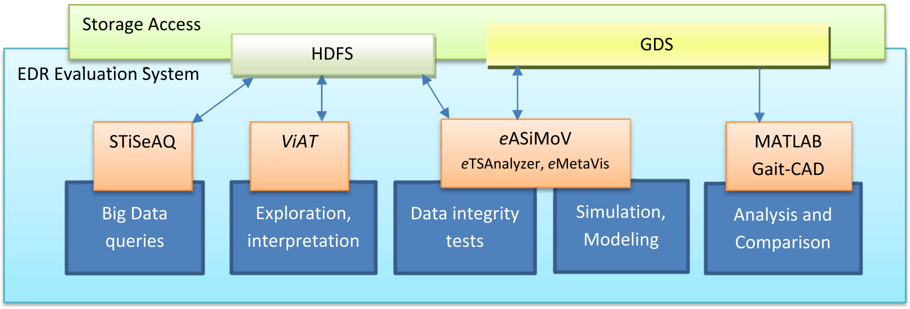


Figure 7 **Schematic representation of the data processing methods.**

to complete. Processing huge time series data with a standard PC architecture has many limitations. One is the decrease of processing speed whenever data to be processed does not fit in local RAM, which has a typical size of about 16 GB in current personal computers. Then, the next slower storage layer must be used, which is the hard disk. Again, as soon as the hard drive's storage limit is reached, data must be fetched from archive servers over the Ethernet, which is again slower. Data I/O becomes the main bottleneck and not - like in other fields - computing effort.

One data-intensive computing approach to lower the complexity of the task is to reduce the relation of data transfer to computing power by distributing the data to multiple computing nodes and to collect the results in a second step. We implemented this method and present it in section 3.1.1. Another principal technique of search task complexity reduction is to preprocess the data once and to extract a smaller indexed database that can be searched much faster afterwards. We keep the preprocessed data together with the original time series and attach the reduced dataset as searchable metadata. This approach is described in section 3.1.2.

3.1.1 Data-intensive computing with MapReduce

In order to reduce computing and data transfer load for search queries and analyses, we developed special methods for data-intensive computing utilizing the MapReduce [38] programming paradigm for distributed, data-locality aware computations. The software is implemented in the Java programming language and bundled under the name STiSeAQ. Using Apache Pig [35,39], we are able to transfer custom analysis code - so-called user defined functions (UDFs) that are written in Java - to the Hadoop cluster, where the query process is automatically optimized and compiled to MapReduce code. Resulting code is then run in a distributed manner on the computers where our target data (feature data or raw data) already persists. By bringing the code to the data and not vice versa, we avoid the bottleneck of slow Ethernet-transfers and disc I/O. This, combined with the distribution of computations, speeds up the processing and enables fast analysis of large datasets. The approach is scalable to any data size, as long as enough HDFS space and computing nodes are provided. This is simple and efficient, since commodity hardware may be used and additional computing nodes can be added as necessary. The speedup depends on data size and is illustrated in Figure 8, showing processing times for computing statistics over EDR voltage time series of different sizes (logarithmic scale), compared to two classical multi-core parallel processing architectures. A detailed description of the survey can be found in [40].

3.1.2 Data reduction using discretization

Although the characteristics of the data-intensive computing platform were known to scale well with data size, we additionally apply more advanced analysis methods to largely improve computing performance. We decided to use a symbolic representation for value and time range discrete aggregation in order to enable the application of machine learning algorithms from the fields of text mining and bioinformatics such as text indexing, Markov models, decision trees, suffix trees, grammar inference, etc., afterwards. These well-experienced methods are defined exclusively for discrete datasets. The main payload of data files will be separated from rather small descriptive, structural, or administrative information, which we keep in relation with the original data as a concept of metadata. Instead of having to read big real-valued data, analysis algorithms are operating on these relatively small but easily extensible metadata. This

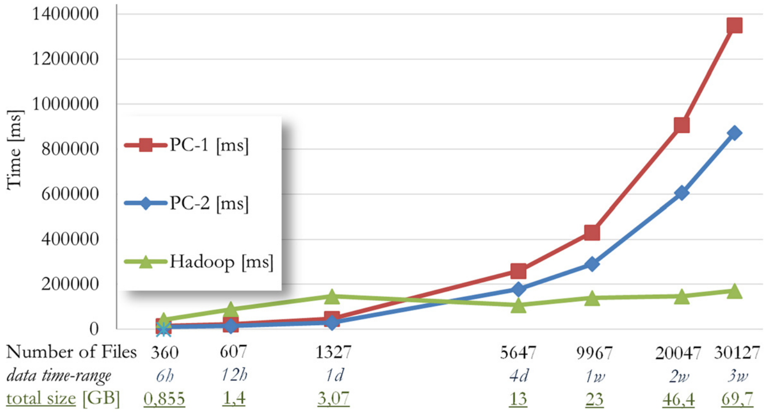


Figure 8 Runtime comparison between Hadoop and multi-core environments. Legend 8: Green triangles: Hadoop environment, red boxes: (PC-1) four core parallel computing environment, blue diamonds: (PC-2) eight core parallel computing environment, *x*-axis in logarithmic scale.

leads to more efficiency and scalability with growing data sizes.

We transformed the time series data to the symbolic aggregate approximation (SAX) [41]. SAX allows for time series distance measures, which lower bound the corresponding distance measures of the original real-valued series. This means that an approximate search in the SAX representation produces correct answers: false positive results may occur due to the reduced accuracy but never false negatives [42]. This property in combination with the dimensionality reduction, the speed of indexing, and the simplicity of SAX makes it an efficient time series representation, especially valuable for indexing and search tasks. Besides the dimensional reduction by temporal piecewise aggregate approximation (PAA), SAX additionally discretizes the measurement value range to symbols. To make time series with different value range comparable, each series is normalized to zero mean and unit of variance (z-normalization) in a preprocessing step.

The symbolic representation is computed on three different detail levels: aggregates for 1 h and for 1 min with an alphabet size of 4 and aggregates for 1 s using an alphabet size of 16. The derived SAX-Strings are saved as XML metadata files with file names that correspond to the original files. For additional search possibilities, basic statistic results like mean, standard deviation, median, minimum, and maximum are calculated from the same time intervals and are added to the metadata for each original feature.

Based on this metadata concept, a more semantic view on the raw data is enabled. We can query directly against the metadata, so the queries are comparatively fast and indexes fit in the RAM or at least on the local disk of a standard PC. Currently, we are able to search for known and partially unknown, uncommon, and frequent patterns (discords and motifs) [43] and for series exceeding defined thresholds in the statistic features. These tasks are integrated in the interactive visualization suite *ViAT* (see sub-chapter 3.2). At the moment, we are developing advanced methods for structural analysis based on the metadata.

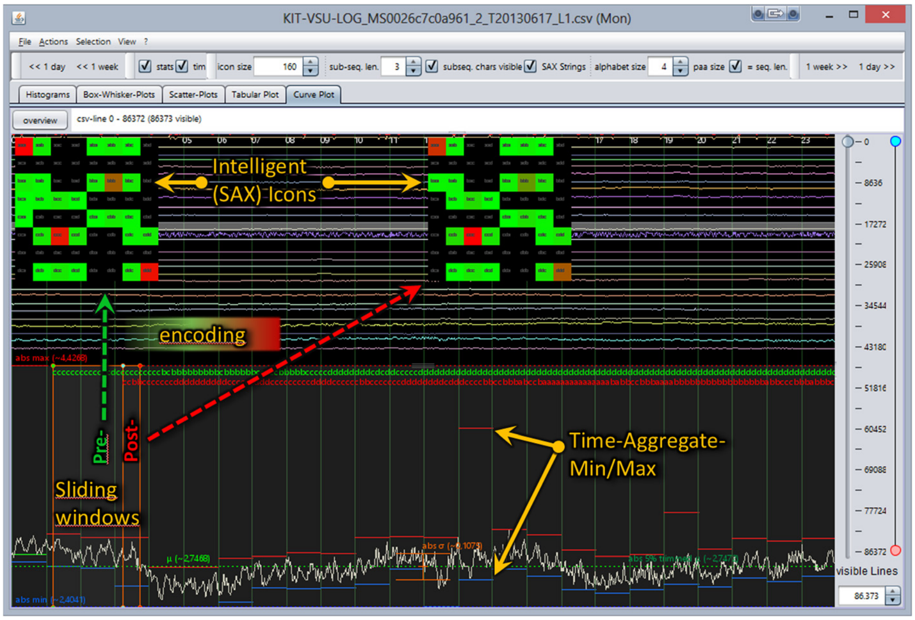
One future goal is to detect the relevance for additional feature extraction methods from the raw data and to decide which measurement rates are useful and which data could be discarded. For example, the data amount could be reduced by lossless compression techniques, but the deflating could be time-consuming and complex for retrieval. We will compare our methods to find an effective combination of consumed space, access speed, and computing power for the needs of future smart grids. Even if the large data are not derived from full waveform captures in the future, the developed methods for the management and evaluation of large data will be needed, e.g., for the handling of a high number of measurement locations, since existing methods are not able to scale with rapid data growth.

3.2 Exploration and interpretation

For fast and interactive explorative analysis of raw data (see sub-chapter 2.1.4), feature data (see sub-chapter 2.1.5), and metadata (see sub-chapter 3.1.2), the software suite *ViAT* was developed. The main goal of the software is to provide a fast overview of the characteristics of data within arbitrary time ranges, with the ability to navigate and zoom interactively and fast while conserving the context within huge multivariate time series, like the EDR measurements.

Since the number of horizontal pixels of computer screens is always limited, visual representations of a number of data points exceeding the number of available pixels always involve a loss of presented information, i.e., some data points are not visible. However, crucial events may occur within the invisible time range. So, one important new concept in ViAT is to additionally compute and visualize metadata like time aggregate statistics for, e.g., seconds, minutes, hours, and days. Aggregate statistic computation can be done either while the data are loaded (online) or in advance as an offline computing step using data-intensive computing (DIC) methodology (see sub-chapter 3.1). The hierarchic aggregate statistic integration enables the visual and automatic detection of outliers and discords [43], while the according curve detail is not visible, by representing it in the next higher time aggregate.

In the main curve plot in Figure 9, the time grid, which is represented by vertical lines, is adaptive to interactive zooming, so the level of wrapped details always fits the visible data range. The zoom is centered at the cursor's position, allowing the user to navigate and zoom in one step towards the cursor position by using the scroll wheel of the mouse. Figure 9 shows a screenshot of the curve plot with additional description of the different parts. As an explorative example, a sliding window was placed interactively by a user. The calculated SAX sequence is presented as lines of symbols for both parts in the middle of Figure 9. From these SAX sequences, so-called 'intelligent icons' [44] are generated and graphically displayed as color-coded matrices at the upper part of the figure. Intelligent icons indicate the probability of the occurrence of a symbol pattern. In Figure 9, all different possibilities of four symbols (a, b, c, d) in a three-symbol pattern (aaa, aab… ddd) are shown. The colors range from green (rare) to red (frequent), whereas black indicates non-occurring patterns. As the visualized analysis result, differences between the pre- and post-window icon illustrate the change in the time series pattern characteristics. In



KIT-VSU-LOG_MS0026c7c0a961_2_T20130617_L1.csv (Mon)

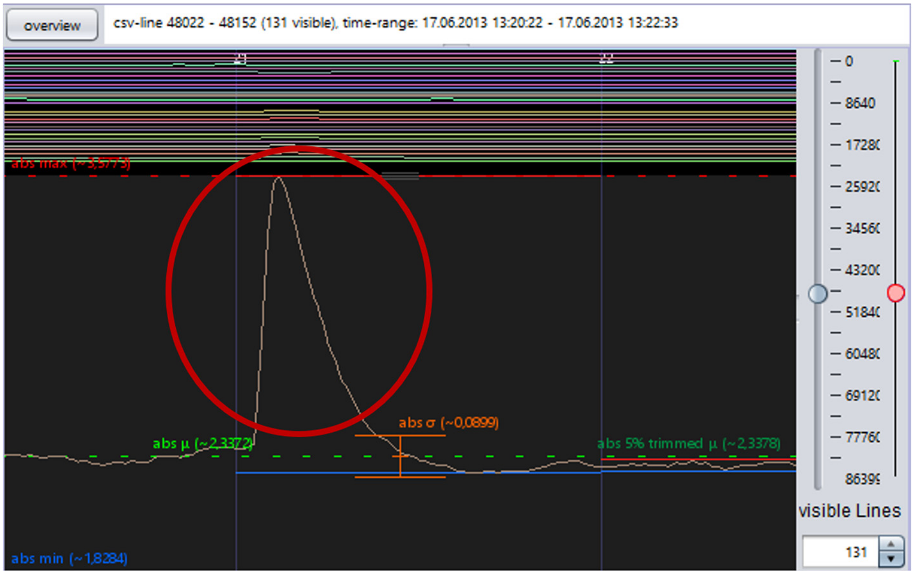
☑ stats ☑ tim | icon size 160 | sub-seq. len. 3 | ☑ subseq. chars visible ☑ SAX Strings | alphabet size 4 | paa size ☑ = seq. len.

csv-line 0 - 86372 (86373 visible)

Figure 9 ***ViAT*** **curve plot display.** Legend 9: User interface of ViAT showing multiple features as curves, a selected curve with graphical aggregate display in the bottom line and a graphical representation of two-dimensional SAX icons of the sliding pre- and post-windows as overlay.

Figure 9, the icons look similar, since the pattern occurrences may be considered more or less equal for both windows. The SAX encoding parameters like symbol width, height, and number as well as pre- and post-window sizes can be arbitrarily defined in order to improve pattern change detection sensitivity. For example, if a longer time span is selected for the pre-window while a shorter interval is defined for the post-window, the respective icons represent a look-ahead-model which could help to identify changes of unknown kind.

Additional visualizations are available, where each has special benefits in visualizing special data features, including histograms, box-whisker-plots, scatter-plots, dendrograms for a SAX-based clustering, and a tabular plot that encodes the curve's values as colors, which would exceed the scope of this paper.



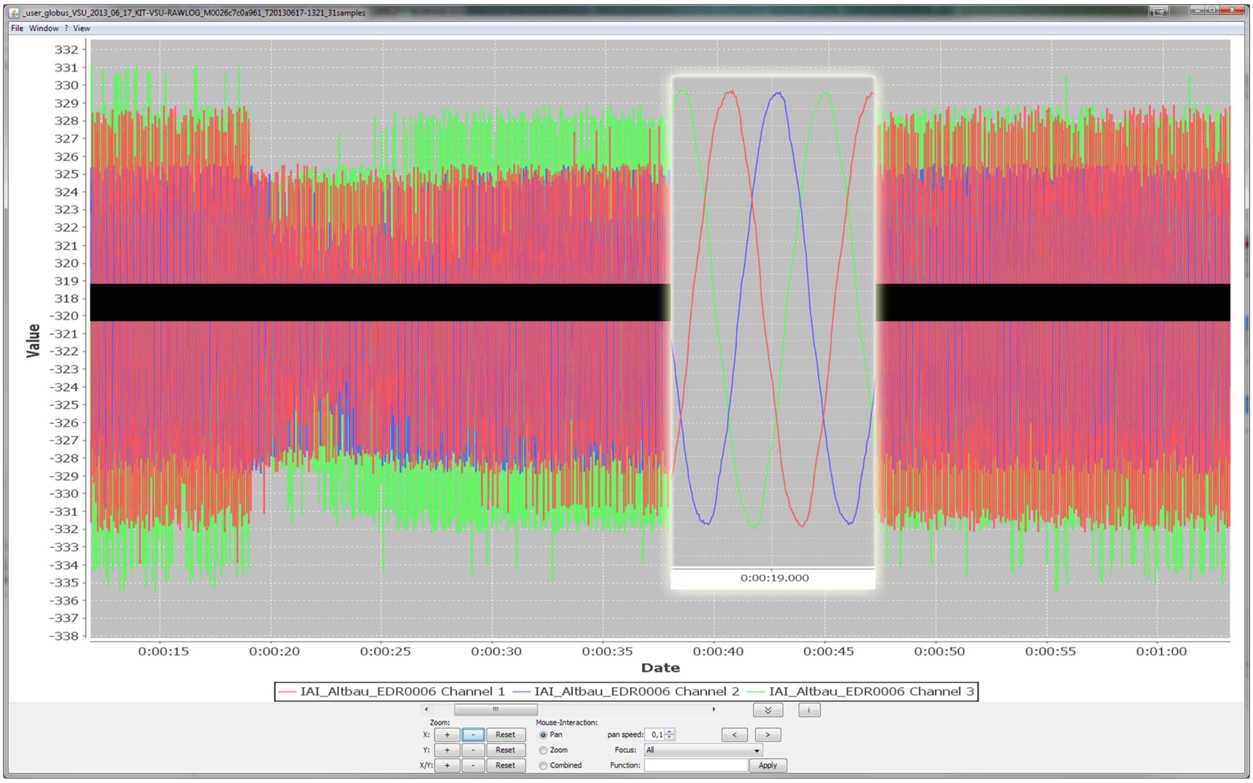
csv-line 48022 - 48152 (131 visible), time-range: 17.06.2013 13:20:22 - 17.06.2013 13:22:33

Figure 10 ***ViAT*** **detail view.** Legend 10: *ViAT* detail view of curve plot showing the detected high peak (encircled in red).

With this tool, we are able to detect important events in the energy grid by examining our measured data in a special way that is much faster than manual exploration on the detail level. One typical workflow for detecting uncommon artifacts is to explore regions where outstanding maxima, minima, or discords were automatically detected, like the exceptionally high maximum in the total harmonic distortion (THD) feature as shown in the center of Figure 9 (an arrow points to the time aggregate maximum). A peak in the THD curve often indicates a transient drop of voltage. Note that the displayed curve itself does not reveal any peak in the according cell for that hour because it is invisible due to aliasing, but the 1-h aggregates, which is depicted by the red horizontal line in the image, indicates that at least one value inside that time range must be extremely high.

When we zoom in to that region, the display automatically changes to 1-min aggregates that guide us to the exact location of the outlier in a fast manner, until we are able to fully see the curve reaching the high level (see Figure 10). We see that the curve has a clear peak (encircled in red) with a long tail at that position, not just an isolated high value, which would indicate an outlier that could just be an artifact of a measurement error. A look at the raw voltage data is possible from within the same view to verify the voltage drop and to find out the exact time of the event on a sub-millisecond scale. In a normal view, the voltage drop is not directly visible at the found position. However, by cropping the vertical value range and displaying only the extreme values, the voltage reduction of about 4 V becomes visible (see Figure 11). We are currently developing methods that enable an automatic detection of these kinds of events directly in the raw data. By comparing occurrence times between measurements from different stations, we may also gain information about how the voltage drop is apparent within the topology of the supply net.

We also developed the IntelSAXSeqEditor at KIT, which is intended for interactive pattern mining based on precomputed metadata and which is also part of the ViAT suite. Using this editor, the operator defines a query sample step-by-step. The software calculates the occurrences of the query pattern in a predefined dataset of SAX sequences (e.g., long-term measurements) after each step and gives color-coded probability information of the remaining possibilities to the operator. By considering this relevance feedback, the user can continue with the next step by defining the next part of the query sample and can thereby identify a highly frequent pattern, which was initially not known.



_user_globus_VSU_2013_06_17_KIT-VSU-RAWLOG_M0026c7c0a961_T20130617-1321_31samples

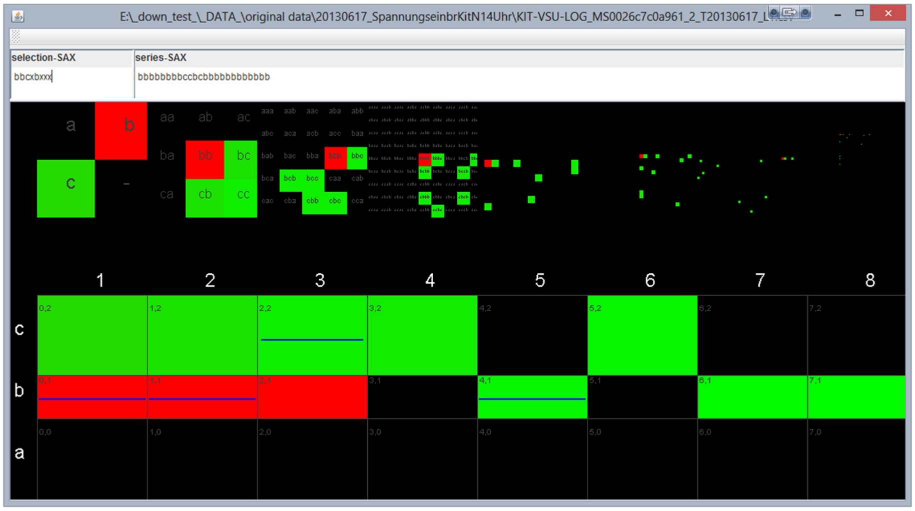
Mouse-Interaction: ● Pan ○ Zoom ○ Combined

pan speed: 0,1

Focus: All

Function:

Figure 11 Voltage drop in raw voltage data. Legend 11: Display of the detected voltage drop in the raw voltage data. The vertical value range is cropped between −320 and 318 V to make the change more visible in this view.



E:_down_test__DATA_\original data\20130617_SpannungseinbrKitN14Uhr\KIT-VSU-LOG_MS0026c7c0a961_2_T20130617_L

selection-SAX: bbcxbxxx

series-SAX: bbbbbbbbccbcbbbbbbbbbbbbb

Figure 12 IntelSAXSeqEditor display. Legend 12: Editor for interactive pattern mining based on a symbolic representation. Table in the lower half: user selected pattern (selection is indicated by blue horizontal lines) and occurring symbol combination colored in red (often) and green (sparse). Table in the upper half: intelligent icons display.

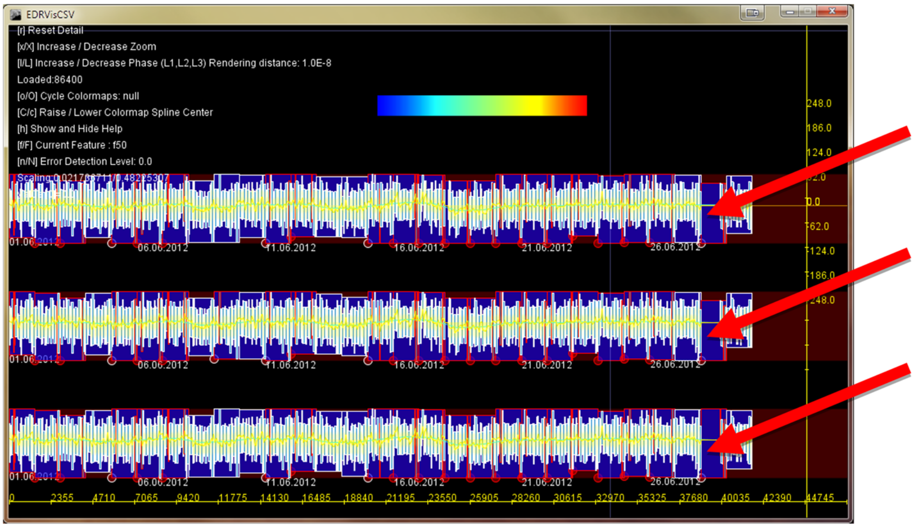
In Figure 12, we see a partially selected (clicked) pattern in the lower half of the editor. The probability color code is red if the pattern occurred very often, green if the pattern appears only sporadically, and black if the pattern is not present. Note that for the current selection (blue horizontal lines in columns 1, 2, 3, 5), we see that there is only one possible occurring pattern for the remaining query sample columns (4, 6, 7, 8). Figure 12 is just a very simple demonstration example. The length of the pattern is adjustable, and the size of the symbol alphabet can be chosen between 2 and 20.

The software IntelSAXSeqEditor was inspired by a similar interactive pattern mining software, which is also based on SAX and called VizTree [45], authored by Jessica Lin. VizTree provides an interactive visual representation of the sequence's suffix tree (called SAX trie), where the thickness of the branches encodes occurrence frequencies. However, the visual representation of such a tree becomes very large and complex for long subseries sizes. Therefore, for IntelSAXSeqEditor, the table design with color encoding was used, that allows mining for much longer patterns.

3.3 Data integrity tests

In-depth analysis of EDR time series characteristics requires data integrity tests, which check the EDR feature data for completeness and correctness. As part of the *e*ASiMoV framework (*e*lectrical grid *A*nalysis, *Si*mulation, *Mo*deling, and *V*isualization), we developed the Java-based software *e*TSAnalyzer (*e*nergy *T*ime *S*eries *Analyzer*), which analyzes EDR data and generates statistical hierarchical and aggregated metadata based on basic statistical data characteristics (minima, maxima, average, deviation, and the median) in the time domain [36]. With a direct access to the Hadoop cluster, *e*TSAnalyzer first checks EDR data files for missing or duplicate files. In the latter case, the most complete and correct data file is further processed. In the second analysis stage, each EDR data file content is analyzed. Since one characteristics dataset per second is expected and each dataset contains one timestamp, the analysis process is straightforward. Missing and multiple datasets as a consequence of possible problems during data transfer or storage are identified, and appropriate metadata together with aggregated statistical data in the time domain are generated. Based on this hierarchical metadata, the interactive Java-based visualization tool *e*MetaVis (*e*nergy *Meta*data *Vis*ualizer) creates a compact and informative graphical representation, which gives a quick overview of huge datasets. Based on statistical analysis, the expected EDR data value ranges are calculated. Interactive visual outlier detection is enabled via data range limit control. The analysis and visualization software are embedded into the BReSoC (*B*roker-based *Re*mote *So*ftware *C*ontrol) framework, which enables remote control of the related software even on Android-based mobile devices. The introduced software frameworks and the tools are at advanced stage of development and currently not yet available as open source.

A data integrity test was applied to early recorded EDR characteristics data in the time period 01 June 2012 to 29 June 2012 with 1.81 GiB stored in 87 csv data files. The data transfer from the Hadoop cluster to the local workstation was done in 27.62 s. The total processing time for the analysis was about 114.4 s on an Intel i5 CPU with 3.3 GHz in single core mode, the EDR file



EDRVisCSV

[r] Reset Detail

[x/X] Increase / Decrease Zoom

[l/L] Increase / Decrease Phase (L1,L2,L3) Rendering distance: 1.0E-8

Loaded:86400

[o/O] Cycle Colormaps: null

[C/c] Raise / Lower Colormap Spline Center

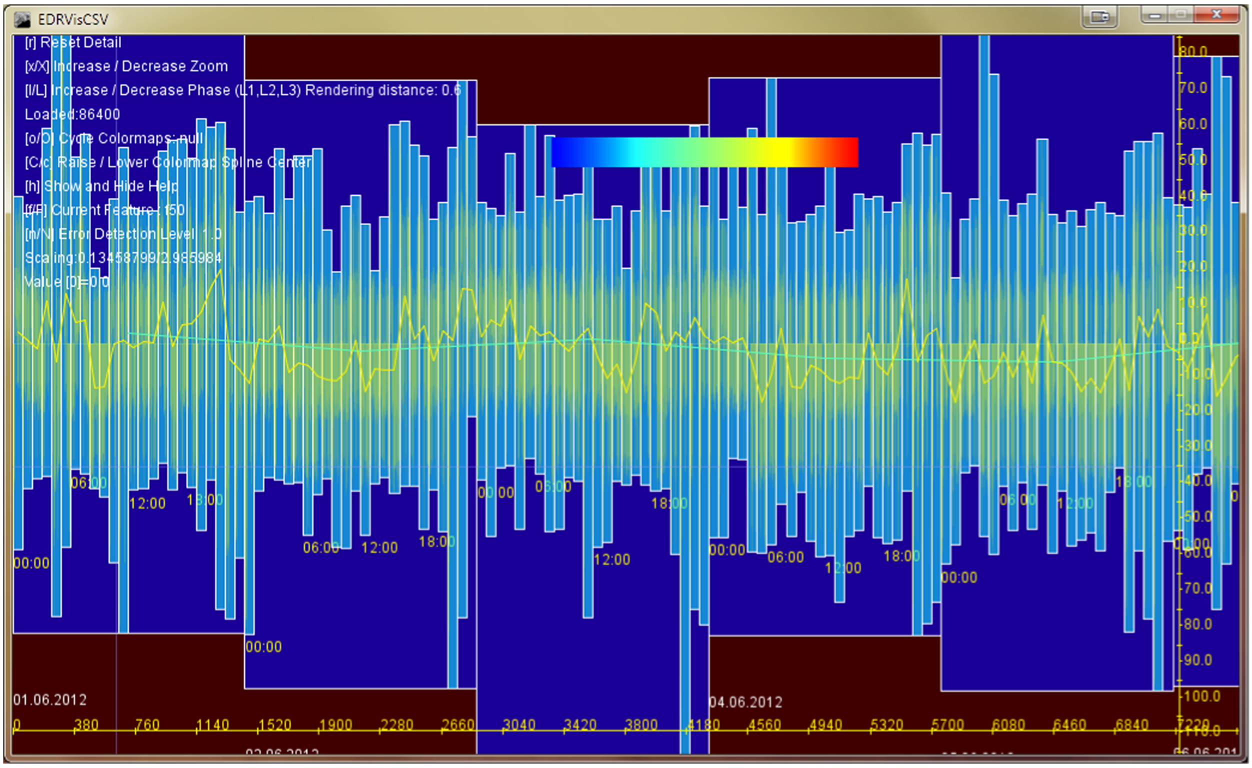
[h] Show and Hide Help

[f/F] Current Feature : f50

[n/N] Error Detection Level: 0.0

Scaling:0.021788711/0.48225307

Figure 13 Data integrity test results. Legend 13: Interactive visualization of integrity test results with *e*MetaVis. Displayed are three-phase EDR frequency deviation data for 29 days. Missing EDR data files can be quickly identified (pointed out by red arrows). Each frame represents 1 day, and outliers are highlighted by color (red frames) and shape (red dots). The interactive variation of the data validity range plotted as dark red horizontal strap enables to detect significant outliers.



EDRVisCSV

[r] Reset Detail

[x/X] Increase / Decrease Zoom

[l/L] Increase / Decrease Phase (L1,L2,L3) Rendering distance: 0.6

Loaded:86400

[o/O] Cycle Colormaps: null

[C/c] Raise / Lower Colormap Spline Center

[h] Show and Hide Help

[f/F] Current Feature : f50

[n/N] Error Detection Level: 1.0

Scaling:0.13458799/2.985984

Value [D]=0.0

Figure 14 Detail view in *e*MetaVis. Legend 14: A detail view of the hierarchical visualization of statistical metadata in the time domain: per day (dark blue), per hour (light blue), and per minute (yellow) as well as the daily and hourly averages.

import and parsing was about 49.1 s, and the statistical calculation and file output took about 65.3 s. The error rate for the total dataset was 3.16%. The inspection of the log files indicated technical problems with the data acquisition for 3 days in this recording period as indicated by red arrows in the three-phase frequency deviation plot of the EDR data in Figure 13. After excluding the corrupt data files from the analysis, the error rate was 42.3 ppm. As shown in Figure 13, the EDR data validity range processing (dark red horizontal strap) *a priori* detects too many anomalies and outliers (red dots and red frames) for a predefined rather strict classification; however, the validity range can be relaxed to detect significant outliers. As shown in Figure 14, the hierarchical visualization of statistical metadata enables to zoom to the highest available temporal data resolution for a detail view.

Missing EDR data files can be quickly identified (red arrows). Each frame represents 1 day, and outliers are highlighted by color (red frames) and shape (red dots). The interactive variation of the data validity range plotted as dark red horizontal strap enables to detect significant outliers.

We found out that duplicate files resulted from the retransmission of the packets. As already mentioned, we fixed this by introducing MD5 hash checks in order to verify the transfer completion of a file. Missing files came from the overrun of the local storage buffer from long-term interruptions of the communication connection to the GDS and from several unexpected halts of the former version of the acquisition software. Irregular time stamp losses are far more complicated to identify. We still observe capturing pauses of 1 to 4 s around five times a week, which we consider as recalibration processes of the acquisition A/D converter. The manufacturer of this converter could not give satisfying information for this issue.

Another data integrity analysis for a more recent EDR dataset from the time period 10 October 2013 to 27 December 2013 showed an error rate of 8.5 ppm, and no corrupt EDR data files were found any more. Thus, continuous integrity tests with *e*TSAnalyzer and *e*MetaVis enabled us to improve the quality of EDR data acquisition software by the factor 5 in the last 2 years.

3.4 MATLAB and Gait-Cad

The EDR feature files from the database can be imported into the MATLAB toolbox Gait-CAD for advanced visualization and analysis tasks [37]. Gait-CAD is an open-source data mining toolbox. It can be downloaded from http://sourceforge.net/projects/gait-cad/.

In a first step, all extracted time series were aggregated to different time scales based on mean, median, maximum, and minimum operators (e.g., for 10-min sample times). In a second step, time series of different buildings, phases, and days are collected into one project for visualization and analysis with data mining methods. It includes analysis with correlation methods to search for occurrences of different effects, cluster methods to group similar day patterns resp. event detections, e.g., to localize unusual situations automatically in large datasets. In addition, various visualization techniques can be applied to enable an interactive analysis.

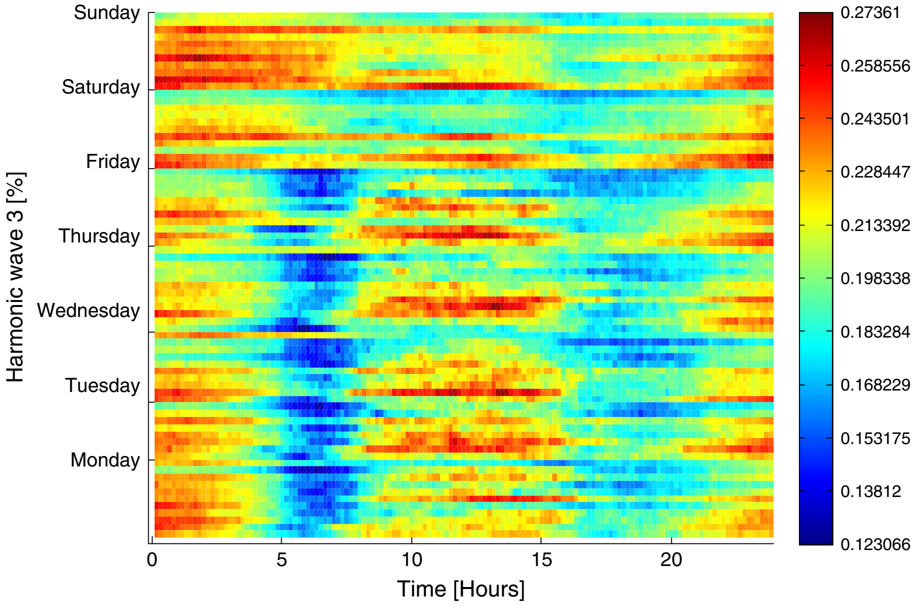


Figure 15 Pattern of the third harmonic wave. Legend 15: Third harmonic in one building for one phase depending on weekdays and based on 10-min median values, over 3-month observation time.

As an example, the third harmonic wave in one building for one phase depending on weekdays is shown in Figure 15. The third harmonic is an important indicator of nonlinear loads and has parasitic effects on the network, especially on the unbalanced currents in the neutral conductor. Further, it is not possible to compensate the third harmonic in the network by reset circuits like for the fifth and the seventh harmonic. A standard PC produces up to 4 A/kW third harmonics current [46]. We captured the voltages at our Institute at KIT over 11 weeks (October to December 2013) and display the third harmonics of one phase, aggregated to 10-min values over the course of the day in color code. In order to identify weekly similarities visually, the lines of the days are grouped by weekdays in the figure.

The image shows typical working days and weekend patterns, which are different for other phases and other buildings. The data are sorted by weekdays, within a weekday, October data are on the bottom, and December data are on the top. In this building, most working days and weekends have the same patterns. As an example, weekdays have low values in the morning (5 to 8 a.m.) and evening hours (4 to 9 p.m., shown in blue). This low morning region is less prominent on weekends. The most prominent changes are due to public holidays (e.g., Christmas in the top data lines for each weekday).

3.5 Evaluation system results

The collected data can be interactively analyzed and explored by advanced visualization methods (as shown in Figures 11, 13, and 15). Using Gait-CAD (see sub-chapter 3.3), the stored data can be automatically analyzed using data mining methods, e.g., by cluster methods to find typical patterns like for tariff-dependent power patterns in the Olympic Peninsula dataset as shown in [47]. We are able to detect and to identify voltage anomalies up to the transient waveform by using data-intensive computing methods as presented in sub-chapter 3.2, e.g., to relate a peak in the THD voltage feature to a voltage drop in the high-rate voltage measurement data. Additionally, we could improve the reliability of the EDR acquisition system by using data integrity tests with *e*ASiMoV as shown in sub-chapter 3.3.

Using the EDR evaluation system, we searched for correlations between characteristics at different locations in the KIT supply grid. We compared locations in the same and in different rings, as well as between office buildings at the same or at different substations. We could not find comprehensive significant dependencies in-between characteristics, between characteristics at five different locations or between the characteristics and the estimated load, which was calculated from 10-min values, measured by standard smart meters in the buildings. A possible explanation is the heterogeneous usage of the electrical installation in the buildings, e.g., different daily consumer configurations affect the harmonic patterns independent on the measured load. Additionally, we did not measure directly at the output of the respective substation transformer, so that we considered the perturbation and the load view of a sub-network only.

In the current development stage, the introduced methods are providing useful information and allow deep insight to the measured data. However, the search of correlations and dependencies is still challenging and lacks in tools that are easy to apply on combinations of data of different sources.

4 Conclusions

In the present contribution, new methods for power grid data acquisition, for smart grid monitoring, for large-scale data exploration, and for deep data analysis are developed. Successfully tested components are installed at the KIT campuses and are collecting high-rate distribution grid data since 2 years. We present the interconnection of the capturing devices as well as the data processing and transport. In contrast to the existing literature, we propose to store the high-rate waveform data for subsequent full data analysis, for which we developed specialized software packages. The feasibility of the whole system is shown by the evaluation results. Special large-scale analysis methods are proposed that utilize data-intensive computing concepts like Apache Hadoop's distributed and data-locality aware computing. The focus is on the creation of a reliable, secure, and accurate recording system and on sophisticated methods, which are needed for large data retrieval and evaluation. We address security measures since parts of the collected data are subject to privacy issues and because information on the status of a distribution grid can be misused to attack its stability with the aim of severe disturbances or blackouts.

The amount of stored data is large in the proposed method, especially when saving the data permanently. However, in combination with distributed storage and analysis methods, this enables the analysis of data on a large scale, i.e., years of measurements, on the one hand and in depth, i.e., on a millisecond scale, on the other hand. Using this full-capture recording at selected locations together with permanent storage will allow for scientific exploration of the dependencies and the creation of valid simulation models on the transient level.

However, the electrical supply of the island-like KIT Campus North is stable and did not exhibit any major marginal conditions since we are recording. In order to obtain more insight in the dependencies between grid status and the measurements, we are going to install the system at less stable grids and will include the load and other state conditions in the evaluation.

As the next main step, we will configure an experimental setup in well-known and configurable smart grids with full-rate waveform captures of voltage and current in the near future. Thereby, we will extract a transient model to estimate the status of this grid far more precisely. Then, we intend to extend the findings to more general networks and provide devices that use measurement and simulation results to predict smart grid behavior and to provide information for sophisticated control.

Competing interests

The authors declare that they have no competing interests.

Received: 30 October 2014 Accepted: 4 February 2015
Published online: 24 February 2015

References

1. CH Hauser, DE Bakken, A Bose, A failure to communicate: next generation communication requirements, technologies, and architecture for the electric power grid. IEEE Power Energ. Mag. **3**(2), 47–55 (2005)
2. S Dierkes, F Bennewitz, M Maercks, L Verheggen, A Moser, Impact of distributed reactive power control of renewable energy sources in smart grids on voltage stability of the power system. Paper presented at the Electric Power Quality and Supply Reliability Conference (11–13 June 2014), pp. 119–126; doi:10.1109/PQ.2014.6866795
3. S Shengnan, M Pipattanasomporn, S Rahman, Grid integration of electric vehicles and demand response with customer choice. IEEE Trans. Smart Grid. **3**(1), 543–550 (2012)
4. A Baggini, *Handbook of power quality* (John Wiley & Sons, Ltd., 2008)
5. D Boroyevich, I Cvetkovic, D Dong, R Burgos, F Wang, F Lee, Future electronic power distribution systems a contemplative view. Paper presented at 12th International Conference on Optimization of Electrical and Electronic Equipment (OPTIM), 20–22 May 2010, pp.1369 –1380; doi:10.1109/OPTIM.2010.5510477
6. V Kirincic, S Skok, I Pavic, Power system state estimation based on PMU measurements vs SCADA measurements. Int. Rev. Model. Simulat (1974–9821) **5**(5), 311–318 (2012)
7. DE Bakken, A Bose, CH Hauser, DE Whitehead, GC Zweigle, Smart generation and transmission with coherent, real-time data. in. Proc. IEEE **99**(6), 928–951 (2011)
8. I Dobson, Voltages across an area of a network. IEEE Trans. Power Syst. **27**(2), 993–1002 (2012)
9. KC Budka, JG Deshpande, M Thottan, *Communication Networks for Smart Grids* (Springer, London, 2014)
10. AG Phadke, JS Thorp, History and applications of phasor measurements. In Proc. IEEE Power Syst. Conf. Expo. Oct./Nov. 2006, pp. 331–335, doi:10.1109/PSCE.2006.296328
11. J Depablos, V Centeno, AG Phadke, M Ingram, Comparative testing of synchronized phasor measurement units. IEEE Power Eng. Soc. Gen. Meet. **1**, 948–954 (2004)
12. M Hurtgen, JC Maun, *Applications of PMU Measurements in the Belgian Electrical Grid. Published as Technical Report*, 2012, pp. 1–75
13. N Gellerman, P Ranganathan, R Vallakati, A Mukherjee, User interface for situational awareness of openPDC. Published as lecture, pp 1–6; doi:10.1109/NAPS.2014.6965418
14. The open source phasor data concentrator 2014, http://openpdc.codeplex.com. Accessed 29 Oct 2014
15. S Skok, D Brnobic, V Kirincic, Croatian academic research wide area monitoring system - CARWAMS. Int. J. Commun. Antenna Propagation - IRECAP **1**(4), 72–78 (2014)
16. X Jiang, P Dutta, D Culler, I Stoica, Micro power meter for energy monitoring of wireless sensor networks at scale. ACM/IEEE IPSN (2007)
17. Electromagnetic compatibility (EMC) – Part 4–30, Power quality measurement methods (IEC 61000-4-30:2008). German version EN **61000**, 4–30 (2009)
18. H Maaß, HK Çakmak, W Süß, A Quinte, W Jakob, E Müller, K Boehm, KU. Stucky, UG Kühnapfel, Introducing a new voltage time series approach for electrical power grid analysis. Paper presented at IEEE EnergyCon2012, 10.09.-13.09.2012, pp. 953–958; doi:10.1109/EnergyCon.2012.6348277
19. H Maaß, HK Çakmak, W Süß, A Quinte, W Jakob, KU Stucky, UG Kuehnapfel, First evaluation results using the new electrical data recorder for power grid analysis. IEEE Trans. Instrum. Meas. **62**(9), 2384–2390 (2013)
20. H Maaß, HK Çakmak, F Bach, UG Kühnapfel, One year high rate low voltage recording - devices, methods and results. Paper presented at IEEE AMPS International Workshop on Applied Measurements in Power Systems (2013), pp. 68–72; doi:10.1109/AMPS.2013.6656228
21. H Maaß, HK Çakmak, F Bach, UG Kühnapfel, Preparing the electrical data recorder for comparative power network measurements. Energy Conference ENERGYCON (2014), pp. 759–765, doi:10.1109/ENERGYCON.2014.6850511
22. Fluke i3000s Flex-24 AC current clamp, 610 mm, http://en-us.fluke.com/products/all-accessories/fluke-i3000s-flex-24.html. Accessed 28. Oct 2014
23. Voltage characteristics of electricity supplied by public distribution networks, German version EN 50160:2010 + Cor. 2010, pp. 1–39
24. Electromagnetic compatibility (EMC) - part 4-7, testing and measurement techniques - general guide on harmonics and interharmonics measurements and instrumentation, for power supply systems and equipment connected thereto (IEC 61000-4-7:2002 + A1:2008); German version EN 61000-4-7:2002 + A1:2009, pp 1–43
25. M Hönig, H Franke, Fact sheet of HBS ACS-0800-PS (2011), http://hbs-electronic.de. Accessed 29. Oct 2014
26. H Maaß, HK Cakmak, UG Kühnapfel, N Ritter, KisGrid - a new network for surgery training. Int. J. Comput. Assist. Radiol. Surg. (2008) **3**(1), 127 (2008)
27. R Rivest, RFC 1321 - The MD5 message-digest algorithm, April 1992, https://www.ietf.org/rfc/rfc1321.txt. Accessed 29. Oct 2014
28. SOAP (Simple Object Access Protocol) Version 1.2, industrial standard of the W3C consortium, http://www.w3.org/TR/soap W3C Recommendation (Second Edition) 27 April 2007. Accessed 28. Oct 2014
29. A García, S Bourov, A Hammad, J van Wezel, B Neumair, A Streit, V Hartmann, T Jejkal, P Neuberger, R Stotzka, The large scale data facility: data intensive computing for scientific experiments. In: 25th IEEE International Symposium on Parallel and Distributed Processing Workshops and Phd Forum (IPDPSW) (2011), pp. 1467–1474; doi:10.1109/IPDPS.2011.286
30. A Kramer, W Jakob, H Maaß, W Süß, *Security in large-scale data management and distributed data acquisition. Paper presented at the 3rd international conference on data management technologies and applications (DATA 2014), Vienna, 29–31 August 2014*, pp. 125–132; doi:10.5220/0005095901250132
31. Bundesamt für Sicherheit in der Informationstechnik (BSI) (Federal Office for Information Security, Germany), Protection Profile for the Gateway of a Smart Metering System (Smart Meter Gateway PP), https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Zertifizierung/ReportePP/pp0073b_pdf, accessed 29. Oct 2014
32. GridPriv, M Stegelmann, D Kesdogan, *A Smart Metering Architecture Offering k-anonymity. Paper Presented at the 11th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom 2012), liverpool, 25–27* (IEEE, Piscataway, 2012), pp. 419–426
33. M Feilner, Open VPN, *Open VPN. Building and Integrating Virtual Private Networks: Learn How to Build Secure VPNs Using this Powerful Open Source Application* (Packt Pub, Birmingham, U.K, 2006)
34. R Oppliger, SSL and TLS, *Theory and Practice* (Artech House, Boston, 2009)
35. Apache Pig platform for analyzing large data sets. http://pig.apache.org, Accessed 28. Oct 2014
36. HK Çakmak, H Maaß, F Bach, UG Kühnapfel, *A new framework for the analysis of large scale multi-rate power data. KIT scientific working papers 21*, ISSN: 2194-1629 (KIT, Karlsruhe, 2014)
37. R Mikut, O Burmeister, S Braun, M Reischl, *The open source matlab toolbox gait-CAD and its application to bioelectric signal processing* (In Proc., DGBMT-Workshop Biosignalverarbeitung, Potsdam, 2008), pp. 109–111
38. J Dean, S Ghemawat, MapReduce: simplified data processing on large clusters. Commun. ACM **51**, 107–113 (2008)
39. A Gates, Programming Pig. (O'Reilly Media, 2011)
40. F Bach, HK Çakmak, H Maaß, UG Kühnapfel, *Power grid time series data analysis with Pig on a hadoop cluster compared to multi core systems. In proceedings of the 21st euromicro international conference on parallel, distributed, and network-based processing, PDP* (Belfast, Ireland, 2013), pp. 208–212
41. J Lin, E Keogh, S Lonardi, B Chiu, A symbolic representation of time series, with implications for streaming algorithms. In Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery (2003), pp. 2–11, doi:10.1145/882082.882086

42. J Lin, E Keogh, L Wei, S Lonardi, Experiencing SAX: a novel symbolic representation of time series. Data Min. Knowl. Discov. **15**(2), 107–144 (2007)
43. E Keogh, J Lin, A Fu, Hot sax: Efficiently Finding the Most Unusual Time Series Subsequence, in *Fifth IEEE International Conference on Data Mining*, 2005, pp. 1–8. doi:10.1109/ICDM.2005.79
44. L Wei, E Keogh, X Xi, S Lonardi, Integrating lite-weight but ubiquitous data mining into GUI operating systems. J. Univers. Comput. Sci. **11**(11), 1820–1834 (2005)
45. J Lin, E Keogh, S Lonardi, JP Lankford, DM Nystrom, Visually Mining and Monitoring Massive Time Series, in *Proceedings of the tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2004, pp. 460–469. doi:10.1145/1014052.1014104
46. ABB Control, Handbuch der dritten Oberwelle. THF 80 DE 99–09, (ABB Schalt- und Steuerungstechnik GmbH, Heidelberg) (in German); pp 1–37
47. S Waczowicz, S Klaiber, P Bretschneider, I Konotop, D Westermann, M Reischl, R Mikut, Data mining to analyse the effects of price signals on household electricity customers. In at - Automatisierungstechnik **62**, 740–752 (2014) (in German)