ORIGINAL ARTICLE

Goals and measures for analyzing power consumption data in manufacturing enterprises

Sören Henning[1] · Wilhelm Hasselbring[1] · Heinz Burmester[2] · Armin Möbius[3] · Maik Wojcieszak[4]

Abstract

The Internet of Things adoption in the manufacturing industry allows enterprises to monitor their electrical power consumption in real time and at machine level. In this paper, we follow up on such emerging opportunities for data acquisition and show that analyzing power consumption in manufacturing enterprises can serve a variety of purposes. In two industrial pilot cases, we discuss how analyzing power consumption data can serve the goals reporting, optimization, fault detection, and predictive maintenance. Accompanied by a literature review, we propose to implement the measures real-time data processing, multi-level monitoring, temporal aggregation, correlation, anomaly detection, forecasting, visualization, and alerting in software to tackle these goals. In a pilot implementation of a power consumption analytics platform, we show how our proposed measures can be implemented with a microservice-based architecture, stream processing techniques, and the fog computing paradigm. We provide the implementations as open source as well as a public show case allowing to reproduce and extend our research.

1 Introduction

The immense electrical power consumption of the manufacturing industry (International Energy Agency 2019) is a considerable cost factor for manufacturing enterprises and a serious problem for environment and society. Corporate values, public relations, energy-related costs, and legal requirements are therefore leading to an increasing energy awareness in enterprises (Shrouf et al. 2017). At the same time, trends toward the Industrial Internet of Things, Industry 4.0, smart manufacturing, and cyber-physical production systems allow to collect energy data in real time and at machine level, from smart meters or machine-integrated sensors (Shrouf et al. 2014; Mohamed et al. 2019). Furthermore, research on big data provides methods and technologies to analyze data of huge volume and high velocity, as it is the case with power consumption data (Sequeira et al. 2014; Zhang et al. 2018). However, even though research suggests a variety of goals and measures for analyzing power consumption data, the full potential of available data is rarely exploited (Shrouf and Miragliotta 2015; Bunse et al. 2011; Cooremans and Schönenberger 2019).

In our Industrial DevOps research project Titan (Hasselbring et al. 2019), we work on methods and tools for integrating and analyzing big data from Internet of Things devices in industrial manufacturing. Analyzing power consumption data in two enterprises of the manufacturing industry serves as a case study. Both enterprises are project

✉ Sören Henning
soeren.henning@email.uni-kiel.de

Wilhelm Hasselbring
hasselbring@email.uni-kiel.de

Heinz Burmester
heinz.burmester@kieler-nachrichten.de

Armin Möbius
a.moebius@ibak.de

Maik Wojcieszak
mw@wobe-systems.com

[1] Software Engineering Group, Kiel University, 24098 Kiel, Germany

[2] Kieler Zeitung GmbH, Co. Offsetdruck KG, Radewisch 2, 24145 Kiel, Germany

[3] IBAK Helmut Hunger GmbH, Co. KG, Wehdenweg 122, 24148 Kiel, Germany

[4] wobe-systems GmbH, Edisonstraße 3, 24145 Kiel, Germany

partners of wobe-systems and Kiel University in the Titan project.

In this paper, we present the goals our studied enterprises aim at to achieve by analyzing power consumption data and propose a set of software-based measures that serve these goals. As start, we conduct a literature review to identify similar goals in related work, suggested measures for our goals, and reported implementations in the literature. Our proposed measures are compiled as part of the Titan project from knowledge of domain experts within the studied pilot cases as well as from our literature review. In a pilot implementation, we show how our proposed measures can be implemented in an Industrial DevOps analytics platform, the Titan Control Center (Henning and Hasselbring 2021). To summarize our contributions:

1. We identify the goals reporting, optimization, fault detection, and predictive maintenance in our pilot cases. Our literature review shows that the goals reporting and optimization are subject of research in various disciplines, whereas fault detection and predictive maintenance based on industrial power consumption are still in an early stage.
2. Based on suggestions from the domain experts in our Titan project and results of our literature review, we propose the measures real-time data processing, multi-level monitoring, temporal aggregation, correlation, anomaly detection, forecasting, visualization, and alerting. Furthermore, we provide a mapping of goals and measures by rating the impact of measures on goals.
3. With our Titan Control Center (Henning and Hasselbring 2021) analytics platform, we show how our proposed measures can be implemented with a microservice-based architecture, stream processing techniques, and the fog computing paradigm. We provide the implementations as open source as well as a public show case allowing to reproduce and extend our research.

The remainder of this paper is structured as follows. Section 2 summarizes the results of our literature review. Section 3 briefly describes the current state of energy monitoring in our studied pilot cases. Section 4 presents the goals for analyzing power consumption data identified in our pilot cases, followed by our proposed measures for tackling these goals in Section 5. Section 6 shows how our proposed measures can be implemented in an analytics platform. Finally, Section 7 concludes this paper.

2 Literature review

Analyzing industrial energy data is an emerging field of research. In this section, we highlight the findings of our literature review regarding goals and measures for analyzing power consumption data as well as related work on implementing such measures.

2.1 Goals for analyzing power consumption data

A lot of research exists, in particular, on how energy data analysis can contribute to reducing the energy usage in manufacturing. For example, Vikhorev et al. (2013) point out that making energy data available for production operators promotes energy awareness. Cagno et al. (2013) show that a lack of energy consumption information prevents implementation of energy-saving measures. Detailed information is especially required at process and machine level for optimizing energy consumption, as highlighted by Thollander et al. (2015). For systematic monitoring and optimizing energy consumption, enterprises are moving towards establishing an energy management (Cooremans and Schönenberger 2019). Implementing an energy management requires to reveal all energy consumptions within the enterprise (Fiedler and Mircea 2012). Schulze et al. (2016) identify organizational measures for implementing an energy management in industry.

Increasing availability of smart meters and Internet of Things (IoT) adoption in the manufacturing industry (Industry 4.0) enable enterprises to collect energy data in great detail (Shrouf and Miragliotta 2015; Miragliotta and Shrouf 2013; Mohamed et al. 2019). This includes commercial metering systems as well as prototypical low-cost systems as proposed by Jadhav et al. (2021). Shrouf and Miragliotta (2015) highlight several benefits of IoT adoption for energy data obtained from reviewing literature and information published by European manufacturing enterprises. Tesch da Silva et al. (2020) present a systematic literature review on energy management in Industry 4.0.

Implementing an energy management and analyzing monitored energy data assist an enterprise in understanding its energy consumption (Miragliotta and Shrouf 2013; Vikhorev et al. 2013; Shrouf et al. 2017). It provides insights into which devices, machines, and enterprise departments use how much power and during which times this power is consumed. Combined with information about the production processes, reports can thus be used to identify which processes consume how much power (Herrmann and Thiede 2009). In this way, measures for optimizing energy consumption can be evaluated and saving potentials can be identified (Bunse et al. 2011).

Literature focuses particularly on optimizing power consumption for economical and ecological reasons (Bunse et al. 2011; Miragliotta and Shrouf 2013; Shrouf et al. 2014; Schulze et al. 2016; Shrouf et al. 2017). Mohamed et al. (2019) report on opportunities provided by IoT energy data for improving energy efficiency and reducing energy costs.

Shrouf and Miragliotta (2015) focus on optimizing energy usage to reduce costs and improve reputation, for example, by reducing energy wastage and improving production scheduling. In their systematic literature review on energy management in Industry 4.0, Tesch da Silva et al. (2020) outline methods for improving energy efficiency and point out current limitations for their implementation.

In order to optimize the overall power consumption, it can be expedient to optimize the operation of machines in production individually. For example, Vijayaraghavan and Dornfeld (2010) optimize the power consumption of machine tools to reduce the power consumption of an entire manufacturing system. Shrouf et al. (2014) optimize the production scheduling of a single machine for minimizing overall energy consumption costs.

A special optimization aspect is the reduction of peak loads (Herrmann and Thiede 2009; Vikhorev et al. 2013; Shrouf and Miragliotta 2015). In addition to the basic price, which is fixed per month, and the price per kilowatt hour, large-scale power consumers such as manufacturing enterprises often have to pay a demand rate. The demand rate depends on the maximum demand that occurs within a billing period. In this way, grid operators expect to have a load as uniformly as possible in the electricity grid (Albadi and El-Saadany 2008). Demand peaks are therefore disproportionately more expensive for the customer. Thus, an optimization should aim to achieve a power consumption as constant as possible, i.e., to distribute the demand evenly over time (peak shaving). In order to achieve this, it is necessary to identify periods during which relatively much power is demanded. Likewise, it is important to discover which consumers are responsible for the demand and to what extent (Herrmann and Thiede 2009).

Other goals besides reporting and optimization can only rarely be found in literature. Quiroz et al. (2018) report how power consumption, which deviates from its normal behavior, can be an indicator for a fault such as a mechanical defect or faulty operation. Analyzing the power consumption of machines can therefore be used to automatically detect such faults and to react accordingly (Vijayaraghavan and Dornfeld 2010; Mohamed et al. 2019). Further, analyzing power consumption data may allow to predict future faults such that necessary maintenance actions can be taken (Shrouf et al. 2014; Mohamed et al. 2019).

2.2 Measures for analyzing power consumption data

Many studies consider near real-time processing of energy data to be necessary (Vijayaraghavan and Dornfeld 2010; Vikhorev et al. 2013; Sequeira et al. 2014; Herman et al. 2018; Liu and Nielsen 2018). Proposed implementations are therefore often using stream processing techniques and

tools (see also the following Section 2.3). Several studies point out that many types of power consumption analysis require consumption data at different levels (Vikhorev et al. 2013; Shrouf and Miragliotta 2015; Kanchiralla et al. 2020). Whereas, for example, the effect of overall power consumption optimizations can be evaluated with data of the overall power consumption, detecting defects in machines requires to acquire data at machine-level. Moreover, different stakeholders are often interested in power consumption reports of different granularity (Shrouf et al. 2017). In addition to aggregating the power consumption of multiple consumers to larger groups, it is often required to aggregate multiple measurements of the same consumer over time (Shrouf and Miragliotta 2015). Analyzing energy data often yields significantly better results if, in addition to recorded power consumption, further information is included such as operational and planning data from the production as well as business data (Vijayaraghavan and Dornfeld 2010; Shrouf et al. 2017).

Most approaches for energy analytics platforms and energy management systems include data visualizations (Fiedler and Mircea 2012; Vikhorev et al. 2013; Sequeira et al. 2014; Zhang et al. 2018). Visualizations are often realized as information dashboards, which contains multiple components providing different types of visualization. Individual components show, for example:

- the current status of power consumption as numeric values or gauges (Rist and Masoodian 2019; Vikhorev et al. 2013)
- the evolution of consumption over time in line charts (Vikhorev et al. 2013; Sequeira et al. 2014; Fiedler and Mircea 2012)
- the distribution among subconsumers and categories (also in the course of time) (Vikhorev et al. 2013; Masoodian et al. 2015)
- correlations of individual power consumer (Sequeira et al. 2014; Masoodian et al. 2017)
- particular important values such as the peak load (Vikhorev et al. 2013)
- detected anomalies (Chou et al. 2017)
- forecasted power consumption (Singh and Yassine 2018)

Research also exists on forecasting power consumption or detecting anomalies in power consumption data. Both practices are closely related. Methods for forecasting and anomaly detection create models of the past power consumption, explicitly or implicitly, and project it into the future (forecasting (Martínez-Álvarez et al. 2015)) or compare the actual power consumption with it (anomaly detection (Chou and Telaga 2014; Liu and Nielsen 2018)). Common approaches use statistical methods such as ARIMA (Chujai et al. 2013) or kernel density estimation

(Arora and Taylor 2016), machine learning methods such as artificial neural networks (Din and Marnerides 2017; Zheng et al. 2017), or a combination of both (Chou and Telaga 2014). Whereas much forecasting and anomaly detection research exists on energy consumption of households (Chujai et al. 2013; Liu and Nielsen 2018), buildings (Arora and Taylor 2016; Chou and Telaga 2014), and electricity grids (Din and Marnerides 2017; Zheng et al. 2017), approaches regarding power consumption of industrial production environments are rare due to their irregular nature (Bischof et al. 2018). Liu and Nielsen (2018) show how alerts could be triggered when anomalies are detected.

2.3 Implementation of measures

Software systems for implementing such measures are presented, for example, by Sequeira et al. (2014) and Rackow et al. (2015). Yang et al. (2020) propose such a system for accessing power consumption at a university campus. However, these systems only focus on a subset of measures proposed in this paper.

A couple of software architectures for implementing energy data analysis are suggested. Several architectures (Sequeira et al. 2014; Shrouf et al. 2017; Herman et al. 2018; Liu and Nielsen 2018) follow the Lambda architecture pattern (Marz and Warren 2015). Such architectures deploy a speed layer for fast online processing and a batch layer for correct offline processing of data. In our pilot implementation (see Section 6), we pursue a more recent architectural style of processing data exclusively online (also referred to as Kappa architecture) (Kreps 2014) by utilizing Apache Kafka's capabilities for reprocessing distributed, replicated logs (Wang et al. 2015). Additionally, we combine this with the microservice architecture pattern and design dedicated, encapsulated microservices per analytics task. Benefits of using microservices and, in particular, the associated concept of polyglot persistence for analyzing industrial energy usage are highlighted by Herman et al. (2018) and Henning et al. (2019).

Big data analytics of energy consumption heavily relies on cloud computing (Shrouf et al. 2014; Herman et al. 2018; Sequeira et al. 2014; Mohamed et al. 2018; Yang et al. 2020). Sequeira et al. (2014) propose *cloud connector* software components for integrating data from energy meters. Recent studies suggest to apply fog computing for integrating production data in general (Qi and Tao 2019) and energy consumption data in particular (Mohamed et al. 2019). Our pilot implementation follows the suggestions of Pfandzelter and Bermbach (2019) to deploy data analytics using stream processing in the cloud and data preprocessing and event processing in the fog. Szydlo et al. (2017) present how data transformation at fog computing nodes can be implemented using flow-based programming and graphical dataflow modeling.

3 Studied pilot cases

In this section, we give a brief overview of our two studied pilot cases. Both pilot cases are enterprises of the manufacturing industry.

The first studied enterprise is a newspaper printing company. It is characterized by high requirements on production speed and the fact that production downtimes are extremely critical. The company has to print and deliver daily newspapers for the next day within only a few hours during the night. If newspapers would be printed too late, they are not up to date anymore and could no longer be sold. Production failures would therefore be associated with significant economic damage. The characteristic production times, with peaks in the nights before working days, are reflected, for example, in the power consumption of the air compressors as depicted in Fig. 1. In addition to daily newspaper printing, the company prints advertising supplements, weekly newspapers, and customer magazines to utilize production capacity.

The second studied enterprise is a manufacturer of optical inspection systems for non-man-size pipelines and wells. This enterprise is characterized by a high vertical range of manufacturing. Thus, its production environment operates a wide range of machines, some of which are largely autonomous, others are primarily user-controlled. Furthermore, the manufacturer operates a rather large data center which runs software for its administration, development, and production. In this paper, we focus on power consumption of the production processes and not on the power consumption of inspection systems themselves.

Both enterprises already have the necessary physical infrastructure to record electrical power consumption in production and query it during operation. Electricity meters already capture the required data with great detail, that is, at machine level and with high frequency. We therefore do not include approaches and techniques for acquiring power consumption data in this paper. However, both companies do not yet exploit the full potential of the recorded and stored data. Currently, they analyze the data mainly by hand and only at certain times. Much of the information hidden in power consumption data is therefore not revealed yet. The reasons for this cannot be found in missing interest, but in a lack of applicable technologies. Currently, the production operators use software provided by metering device manufactures, for example, to visualize the stored data. However, this software does not meet all requirements. For example, the amount of visualized data is too large, making it hard to extract the really important

Fig. 1 Power consumed for generating compressed air in the newspaper printing company over a period of 2.5 weeks. The curve shows a weekly pattern and reflects the company's operating hours with constant low consumption at the weekend and peak loads at night
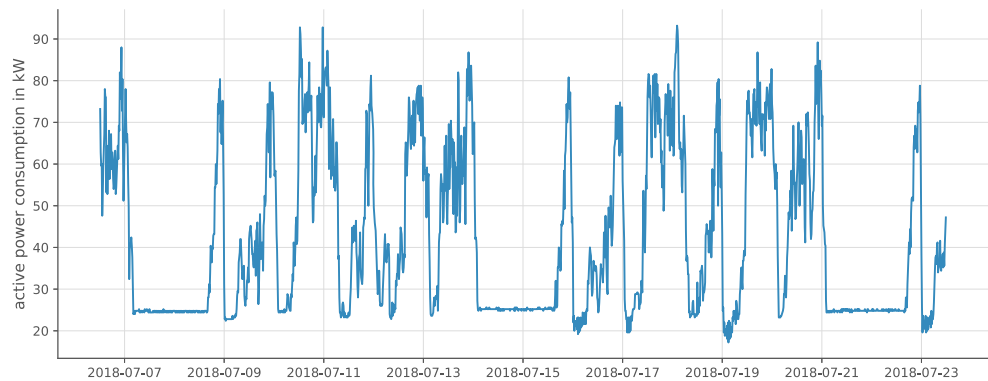
information. Another issue is the integration of different types of electricity meters. Although standardized protocols exist, many metering devices and systems do not apply them.

4 Goals for analyzing power consumption data

In this section, we identify motivations for analyzing power consumption data in our studied pilot cases. We classify these motivations into the four goal categories reporting, optimization, fault detection, and predictive maintenance. Our literature review (see Section 2) suggests that these goals also occur in other manufacturing enterprises as similar motivations can be found in related studies. In the following, we describe each goal category in detail.

4.1 Reporting

In both studied enterprises, comprehensive reporting is particularly required for an (ISO 50001 2018) certification. The ISO 50001 standard specifies requirements for organizations and businesses for establishing, implementing, and improving an energy management system. It describes a systematic approach to support organizations in continuously improving their energy efficiency. In order to be certified to use an energy management system in compliance with ISO 50001, enterprises commission accredited certification bodies to perform regular independent audits (Jovanović and Filipović 2016). These certifications are usually not required by law, but serve as evidence that a company is making efforts to save energy.

Both companies consider sustainability as an important pillar of their corporate philosophy. ISO 500001 certification allows them to demonstrate their efforts in saving energy to customers and other stakeholders. Moreover, in Germany, where both enterprises are located, ISO 50001 certification enables cost savings as such a certification is a prerequisite for manufacturing enterprises with high power

consumption to reduce regulatory charges (e.g., reducing the EEG surcharge (Bundesamt für Wirtschaft und Ausfuhrkontrolle (BAFA) 2020)). Certification is even essential for the manufacturer of optical inspection systems. Its customers are mainly public authorities, which often require ISO 50001 certification in their calls for tender.

Reports for ISO 50001 certification are required to justify irregular or increasing power consumption. This is in particular challenging for the newspaper printing company, where power consumption highly depends on the production utilization and external influences. Hence, this company requires to perform complex analyses for their reports, such as correlations with external data from the production and the environment. Moreover, the ISO 50001 standard requires that reports on the enterprise's energy consumption are available for customers, stakeholders, employees, and management.

4.2 Optimization

The ecological and economical motivations for optimizing energy consumption presented in our literature review (see Section 2) also apply to both our pilot cases. We identify the following types of optimizing power consumption in the studied enterprises.

Optimization of Overall Consumption For optimizing the overall power consumption, a first step is to identify energy-inefficient machines and devices. This knowledge can then be used to replace them with more energy-efficient ones or retrofitting them accordingly. Furthermore, time periods should be detected in which devices consume energy, although it would not be necessary. Typical examples of unnecessary energy consumption are keeping machines in standby mode or lighting workplaces outside of working hours, but also less apparent saving potential is expected to be discovered.

Optimization of Peak Loads Being large-scale power consumers, both studied pilot cases have to pay a demand rate

based on the maximum demand within a billing period. Thus, demand peaks are disproportionately more expensive. An optimization aim is therefore to achieve a power consumption as constant as possible, i.e., to distribute the demand evenly over time (peak shaving). In order to achieve this, it is necessary to identify periods during which relatively much power is demanded. Likewise, it is important to discover which power consumers are responsible for the demand and to what extent. This includes, on the one hand, the identification of large consumers in general but, on the other hand, also demand fluctuations of individual devices. Based on this information, production processes can be modified such that, for instance, multiple machines with a high inrush current are not started at the same time. Reducing the overall energy consumption is highly related to reducing peak loads. If measures are taken to replace devices, this has an effect on both optimization goals. For example, if devices that are unnecessarily operated standby during load peaks are turned off during these periods, not only demand peaks are reduced, but also the enterprise's power consumption in total.

Optimization on Machine-Level Similar to what we present in Section 2, it is reasonable to optimize the operation of machines or production processes to optimize the overall power consumption or peak loads. A potential power saving measure in the newspaper printing company exists in the printing process. The number of newspapers produced per unit time depends directly on the operating speed of the printing presses. To determine an optimal printing speed, several other factors are also taken into account, such as reliability, which decreases when increasing production speed. With monitoring and analyzing the printing presses' power consumption, the company can also include energy-related costs when determining the production speed.

4.3 Fault Detection

The studied enterprises report that a power consumption of machines, deviating from their normal behavior can be an indicator for a fault such as a mechanical defect or faulty operation. Analyzing the power consumption can therefore be used to automatically detect such faults and to react accordingly. A typical case of anomalous power consumption is a strong increase, for example, when a defect occurs suddenly. A decrease of power consumption can also be such an indicator as parts of a machine may no longer be operated due to a defect. Less noticeable is a slight deviation over a longer period of time, for example, if several minor defects occur over time. Detecting deviations or a long-term trend in regularly fluctuating power consumption is even more challenging.

The central compressed air supply in the newspaper printing company is an example for fault detection using power consumption data. An extensive pipe network supplies various areas of the production environment and finally individual machines with compressed air. The compressed air distribution network leaks regularly, causing air to escape. These leaks do not necessarily become apparent directly, but should still be repaired. As leaks result in higher power consumption of the air compressors, power consumption data can provide an instrument for leak detection. However, since power consumption of the air compressors is subject to strong, irregular fluctuations (see Fig. 1), an increase in power consumption does not immediately become apparent. This may be solved by considering the power consumption only in idle times, for example, during the weekend. An increase in power consumption over several weekends may thus be an indication of a leak. Figure 2 shows the average power consumption between Saturday 12:00 and Sunday 12:00 for each weekend in 2017 and 2018. The course shows a steady increase in 2017 due to leaks in the compressed air supply. In early 2018, the company repaired several leaks, causing a tangible reduction in power consumption.

4.4 Predictive maintenance

With regular, time-based maintenance intervals, machines and devices are often maintained even though there is no actual need for it. This means that components and operating materials are replaced since their expected operating time expires, although they are still functioning and could actually continue operating. Predictive maintenance is an approach that aims for performing maintenance actions only if it would otherwise results in defects or limitations in performance or quality (Yan et al. 2017). The difficulty is therefore to decide when maintenance is really necessary. For this purpose, sensor data of the machine and its environment are collected and automatically analyzed (Yan et al. 2017). Our literature review (see Section 2) suggests that power consumption can be such data.

We distinguish between predictive maintenance and fault detection as while fault detection aims to detect errors after they occurred, predictive maintenance refers to the detection of errors before they occur. Nevertheless, predictive maintenance is closely related to fault detection as occurring faults often cause further faults. Therefore, early fault detection may allow future faults to be detected and appropriate preventive measures to be taken.

An example for predictive maintenance using power consumption are cooling circuits as used in the studied enterprises. Such circulation systems typically include a filter through which coolant is pumped to remove impurities. These filters need to be replaced regularly. The

Fig. 2 Stand-by power consumption for generating compressed air in the newspaper printing company at weekends over a period of two years
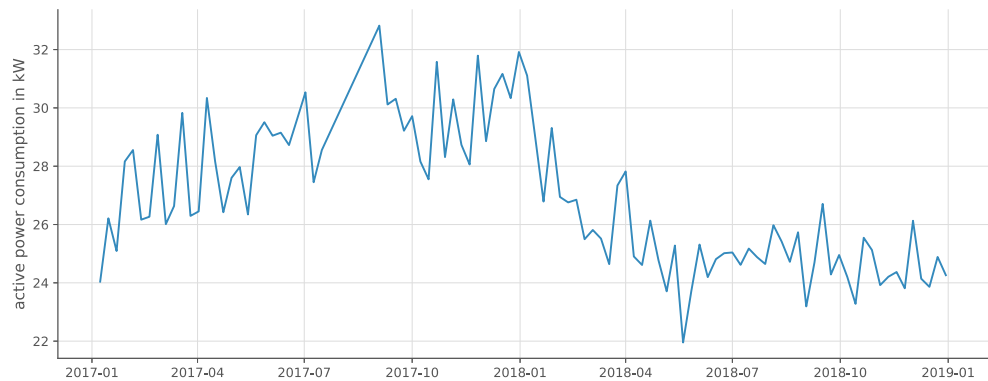
electrical power consumption of the pump indicates the resistance within the circulation system and, thus, how polluted the filter is. Increased power consumption can therefore serve for detecting an upcoming filter change. Lower power consumption can also provide information. It may indicate that not enough coolant is in the circuit (referred to as dry run) and, thus, coolant needs to be refilled.

5 Measures for analyzing power consumption data

In this section, we discuss software-based measures for analyzing power consumption data that support in achieving the goals defined in the previous section. Based on our literature review in Section 2 and knowledge from domain experts within our studied pilot cases, we suggest the following measures: real-time data processing, multi-level monitoring, temporal aggregation, correlation, anomaly detection, forecasting, visualization, and alerting. Different use cases weight goals differently and measures vary in their importance for the individual goals. We therefore rate the impact of each measure on each goal and visualize these impacts on radar charts shown in Fig. 3. In the following, we briefly describe each measure and characterize how each measure affects each goals.

5.1 Near real-time data processing

Near real-time (also referred to as online) data processing describes approaches, where data are immediately processed after their recording. It contrasts batch (also referred to as offline) processing, which first collects recorded data and then processes all the collected data only at certain times. Whereas near real-time data processing is usually more difficult to design and implement than batch processing, it yields immediate results and, thus, allows to react immediately on these results.

Data processing in near real-time supports primarily the goals optimization (see Fig. 3b), fault detection (see Fig. 3c), and predictive maintenance (see Fig. 3d) (Vijayaraghavan and Dornfeld 2010; Shrouf and Miragliotta 2015). Power consumption can be efficiently optimized if the effectiveness of energy-saving actions are evaluated immediately. The sooner a fault is detected and reported, the faster it can be reacted to the fault and, therefore, the more valuable its detection is. Predictive maintenance requires processing monitoring data in real time as otherwise the time for maintenance may be determined after the maintenance should have already been performed (Sahal et al. 2020). Although a real-time overview of the enterprise's energy usage at any time is not required for ISO 50001 audits, it assists in reporting (see Fig. 3a) the power consumption, for example, to the management (Miragliotta and Shrouf 2013).

5.2 Multi-level monitoring

We suggest to organize power consumers in a hierarchical model, where groups of devices and machines are further grouped into larger groups (Henning and Hasselbring 2020). Multiple such models have to be maintained in parallel. For example, it is reasonable to organize devices by their type (e.g., all air compressors), but also to organize them by their physical location (e.g., a certain shop floor).

Besides monitoring groups of consumers, for example, via sub-distribution units, data for groups can also be obtained by aggregating the consumption of all its partial consumers. In particular, this is necessary for devices which, for reasons of redundancy, have more than one power supply. Here, the overall machine's power consumption is usually more important than the power consumption of the individual power supplies. Comparing the power consumption monitored by sub-distribution units with aggregated data of all known sub-consumers may reveal consumptions, which were unknown so far.

Hierarchical models of power consumers particularly support reporting (see Fig. 3a) as they offers insights at

Fig. 3 Impact rating of the proposed measures for the four goals presented in Section 4. The larger its distance from the radar chart's center is, the higher a measure's impact was weighted on the corresponding goal

(a) Reporting

(b) Optimization

(c) Fault Detection

(d) Predictive Maintenance
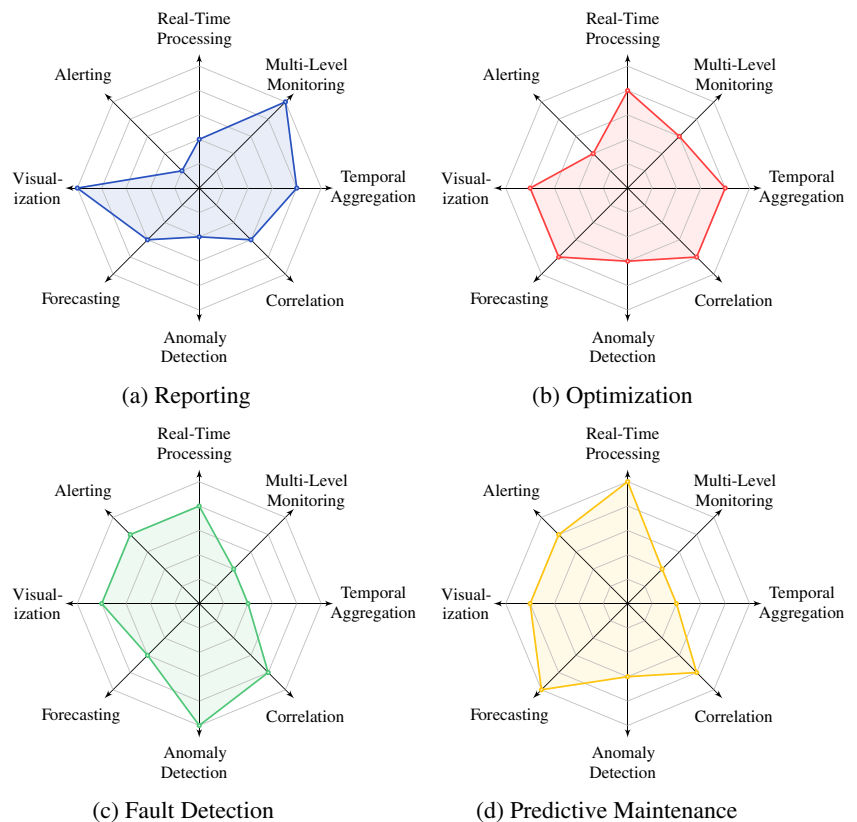
which times which consumers or groups consume how much power. Power consumption can be optimized (see Fig. 3b) on machine-level as well as on aggregated data (see Section 2). Furthermore, our literature review shows that also fault detection (see Fig. 3c) and predictive maintenance (see Fig. 3d) may be performed on different levels.

5.3 Temporal aggregation

Temporal aggregation refers to summarizing multiple measurements of the same consumer over time to one data point. It serves for: (1) reducing the number of data points for storage and (2) simplifying data analysis by providing a more abstract view on the data. Therefore, temporal aggregation supports humans in comprehending the monitored power consumption data and, thus, reporting (see Fig. 3a) as well as manual identifying optimization potentials. Also automatic data processing for optimization (see Fig. 3b), fault detection (see Fig. 3c), and predictive maintenance (see Fig. 3d) may benefit from aggregated data. We distinguish two different kinds of temporal aggregation as described in the following.

Aggregating Tumbling Windows The first kind is to collect and aggregate all measurements in consecutive, non-overlapping, fixed-sized time windows (tumbling windows (Carbone et al. 2019)). An appropriate size for such

windows is, for example, 5 minutes so that every 5 minutes a new aggregation result is computed representing the average, minimal, and maximal power consumption over the previous 5 minutes. The number of data points can thus be massively reduced, which is required for several forms of storing, analyzing, and visualizing data. We suggest to perform multiple such aggregations (e.g., for time windows of size 1 minute, 5 minutes, and 1 hour) and store their aggregation results for different durations. This allows to store more recent (and more interesting) data with more detail than data from the previous months or years.

Aggregating Temporal Attributes The second kind of temporal aggregation is to aggregate all data points having the same temporal attribute such as day of week or hour of day. The set of aggregated data points allows to model or identify seasonality. For example, aggregating all measurements recorded at the same day of week allows to show the average power consumption course over a week. Likewise, aggregating based on the hour of the day allows to obtain the average course of a day.

5.4 Correlation

Our literature review (see Section 2) shows how operational and planning data from the production as well as business data can be included in different types of power

consumption analysis. Furthermore, it is reasonable to correlate the power consumption of different consumers as their power consumption may depend on each other if their production processes are depended (Bischof et al. 2018).

Correlating power consumption data with production data supports reporting (see Fig. 3a) as it allows for better understanding the power consumption. Management levels might be interested in a correlation with business data as this allows to report about, for example, the energy costs per produced unit. In particular, correlation can serve as trigger for optimization (see Fig. 3b), fault detection (see Fig. 3c), and predictive maintenance (see Fig. 3d). If, for example, the power consumption of a machine increases rapidly while also the production speed increases, the increasing power consumption was most likely not caused by a fault. If, however, the production speed remains constant and no other production data justifies the increase, a fault detection could be triggered. Correlations of power consumption of different consumers are interesting for reporting, but also for optimizations, in particular for reducing load peaks.

5.5 Anomaly detection

Anomaly detection (also referred to as outlier detection) describes methods for automatically finding unexpected pattern in data (Chandola et al. 2009). We suggest to employ anomaly detection techniques to discover time periods, during which power consumption is unexpectedly high or low, like it is done for energy consumption of building or household in related work (see Section 2). This includes continuously computing anomaly scores for observed power consumption and comparing these anomaly scores with previously defined thresholds. We suggest to apply anomaly detection both on monitored and aggregated data (see Section 5.2).

Primarily, anomaly detection serves as a measure for the goal of fault detection (see Fig. 3c). Faults in devices, machines, or production processes are deviations from the desired behavior and, thus, anomalous behavior of power consumption may indicate an occurring fault. Detecting anomalies in power consumption exclusively in relation to time is often not sufficient. The consumption of many devices is subject to external influences such as temperature (Liu and Nielsen 2018) and, especially in production environments, the operating times of machines do not follow daily or weekly patterns (Bischof et al. 2018). Correlating power consumption with environmental, operational, and planing data (see Section 5.4) therefore assists in detecting anomalies.

Furthermore, anomaly detection allows to identify potential applications of optimization (see Fig. 3b) and predictive maintenance (see Fig. 3d) and supports in

explaining power consumption behavior in reporting (see Fig. 3a).

5.6 Forecasting

As highlighted in our literature review (see Section 2), analyzing power consumption data allows to make predictions about the future power consumption. Similar to anomaly detection, predicting power consumption in industrial production poses additional challenges in contrast to predicting power consumption of households, buildings, or electricity grids. To cope with the irregular nature of industrial power consumption, correlation with environmental, operational, and planning data (see Section 5.4) promises to create more accurate models.

Forecasting the power consumption of machines in addition to the overall production environment supports optimization (see Fig. 3b) as it allows to detect load peaks before they actually occur. Thus, production operators may take appropriate countermeasures such as replanning production processes. Making predictions about the future status of the production environment is required for predictive maintenance. Thus, forecasting power consumption enables predictive maintenance (see Fig. 3d) based on power consumption data. Fault detection (see Fig. 3c) based on anomaly detection often relies on forecasts by comparing the actual consumption with the expected (i.e., forecasted) one. Furthermore, forecasting can be used in reporting (see Fig. 3a) as it supports planing and decision making for business and production operation.

5.7 Visualization

According to our literature review in Section 2, we propose to visualize analyzed power consumption data in information dashboards. This way, visualizations integrate individual measures proposed in this chapter and serve as a link between data analysis and the users. Dashboards should be dynamic and interactive in the sense that they are updating their visualized data continuously and let users interact with them (Rist and Masoodian 2019). For example, dashboards may start with a rough outline of the overall production's power consumption but allow users to zoom in and show specific machines and time periods in detail. We summarize typical visualizations for an industrial power consumption dashboard in Section 2.2.

As state-of-the-art libraries and frameworks for data visualization are largely based on web technologies (Bostock et al. 2011), it is reasonable to implement dashboards as web applications. This has the additional advantage that the visualization is user-friendly accessible since it does not have further requirements on software or hardware infrastructure than a web browser.

First and foremost, dashboards enable reporting (see Fig. 3a) on power consumption. Appropriate visualization allows to understand how power consumption is composed, observe changes in power consumption over time, and compare the power consumption of different machines and production processes. Enterprises may provide different dashboards for different stakeholders to only show the information, which is relevant for the corresponding target audience (Shrouf and Miragliotta 2015). Visualizations assist in optimization (see Fig. 3b) as they allow to identify optimization potentials and enable operators to check whether optimization actions are effective. Furthermore, interactive visualizations can motivate, trigger, and enable energy saving actions (Rist and Masoodian 2019). A dashboard may also show information concerning fault detection (see Fig. 3c) and predictive maintenance (see Fig. 3d) and provide means to verify whether faults and maintenance actions are detected successfully (Shrouf and Miragliotta 2015).

5.8 Alerting

Industrial production becomes increasingly autonomous (Lasi et al. 2014). Permanently observing a dashboard (see Section 5.7) and waiting for faults or necessary maintenance to be detected can therefore be a tedious work. Instead, it would be convenient to automatically notify production operators when faults are detected or maintenance actions have to be taken (see Figs. 3c and 3d). Depending on their frequency and severity, such notifications and alerts may be sent via email or messenger. For reporting (see Fig. 3a) purposes, such notifications may additionally be displayed in a dashboard. Furthermore, operators may be notified if optimization potential is detected (see Fig. 3b), for example, by generating an alert if a load peak is about to occur.

6 Pilot implementation of the measures

In this section, we show how the measures proposed in Section 5 can be implemented in a software architecture that adopts the microservice architecture pattern, big data stream processing techniques, and fog computing. In our Titan project on Industrial DevOps (Hasselbring et al. 2019), we develop methods and techniques for integrating Industrial Internet of Things big data. A major emphasis of the project is to make produced data available to various stakeholders in order to facilitate a continuous improvement process. The Titan Control Center[1] is our open source pilot application for integrating, analyzing, and visualizing industrial big data

from various sources within industrial production (Henning and Hasselbring 2021).

The architecture of the Titan Control Center follows the microservice pattern (Newman 2015). It consists of loosely coupled components (microservices) that can be developed, deployed, and scaled independently of each other (Hasselbring and Steinacker 2017). Our architecture features different microservices for different types of data analysis. Individual microservices do not share any state, run in isolated containers (Bernstein 2014), and communicate only via the network. This allows each microservice to use an individual technology stack, for example, to choose the programming language or database system that fits the service's requirements best. In a previous publication (Henning et al. 2019), we show how these architecture decisions facilitate scalability, extensibility, and fault tolerance of the Titan Control Center.

Figure 4 shows the Titan Control Center architecture. It contains the microservices Aggregation, History, Statistics, Anomaly Detection, Forecasting, and Sensor Management. In addition to these microservices, our architecture comprises components for data integration, data visualization, and data exchange.

The Titan Control Center is deployed following the concepts of edge and fog computing (Garcia Lopez et al. 2015; Bonomi et al. 2012). In particular suited for Internet of Things (IoT) data streams, with edge and fog computing data is preprocessed at the edges of the network (i.e., physically close to the IoT devices), whereas complex data analytics are performed in the cloud (Pfandzelter and Bermbach 2019). In order to facilitate scalability and fault tolerance, the Titan Control Center microservices for data analysis and storage are deployed in a cloud environment. This can be a public, private, or hybrid cloud, which allows elastic increasing and decreasing of computing resources. On the other hand, software components for integrating power consumption data into the Titan Control Center are deployed within the production. This includes querying or subscribing to electricity meters, format and unit conversions, filtering, but also aggregations to reduce the amount of data points. We employ our Titan Flow Engine (Hasselbring et al. 2019) for this purpose. It allows graphical modeling of data flows in industrial production according to flow-based programming (Morrison 2010). With the Titan Flow Engine individual processing steps are implemented in so-called *bricks*, which are connected via a graphical user interface to *flows*. This enables production operators to reconfigure power consumption data flows, for example, to integrate new electricity meters, without having advanced programming skills.

All communication among microservices as well as between the data integration and microservices takes place asynchronously via a messaging system. We use Apache

---

[1] https://github.com/cau-se/titan-ccp
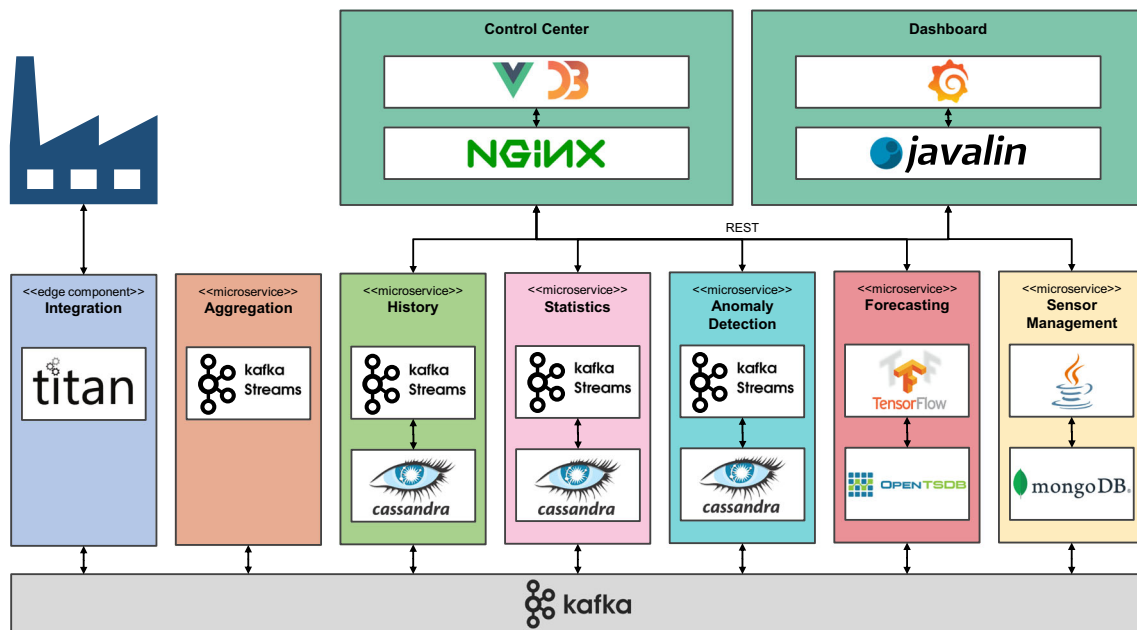
Fig. 4 Microservice-based pilot architecture of the Titan Control Center for analyzing electrical power consumption

Kafka (Kreps et al. 2011) in our pilot implementation. Moreover, the Titan Control Center features two single-page applications that visualize analyzed data and allows for configuring the analyses.

In the following, we present how each measure proposed in Section 5 can be implemented using the Titan Control Center.

6.1 Near real-time data processing

Power consumption data is processed in near real time at all architectural levels of the Titan Control Center. This start by the ingestion of monitoring data and immediate filter, convert, and aggregate operations in the Titan Flow Engine at the edge. The final integration step is sending the monitoring data to the messaging system. Following the publish–subscribe pattern, microservices subscribe to this data stream and are notified as soon as new data arrive. In the same way, individual microservices communicate with each other asynchronously. Apache Kafka as the selected messaging system is proven for high throughput and low latency (Goodhope et al. 2012). Within microservices, we process data using stream processing techniques (Cugola and Margara 2012). This implies that microservices continuously calculate and publish new results as new data arrive. For implementing stream processing architectures in most of the microservices we use Kafka Streams (Sax et al. 2018). As all computations are performed in near real time, also the visualizations can be updated continuously. Hence, the visualization applications (see Section 6.7) periodically request new data from the individual services.

6.2 Multi-level monitoring

The Aggregation microservice (Henning and Hasselbring 2019) of the Titan Control Center computes the power consumption for groups of machines by aggregating the power consumption of the individual subconsumers. This microservice subscribes to the stream of power consumption measurements coming from sensors, aggregates these measurement continuously according to configured groups, and publishes the aggregation results via the messaging system as if they were real sensor measurements. In addition to sensor measurements, however, these data are enriched by summary statistics of the aggregation.

As proposed in Section 5.2, the Aggregation microservice supports aggregating sensor data in arbitrary nested groups and multiple such nested group structures in parallel. In one of our studied enterprises, we integrate power consumption data of different kinds of sensors, which provide data in different frequencies. An important requirement for the Aggregation service was therefore to support different sampling frequencies. Furthermore, besides the focus on scalability throughout the entire Control Center architecture, an important requirement for this microservice is to reliably handle downtimes and out-of-order or late arriving measurements. Therefore, it allows to configure the required trade-off between correctness, aggregation latency, and performance (Henning and Hasselbring 2020).

The Sensor Management microservice of the Titan Control Center allows to assign names to sensors and arrange these sensors in nested groups. For this purpose, the Titan Control Center's visualization components provides a

corresponding user interface. The Sensor Management service stores these configuration in a MongoDB (MongoDB 2019) database. It publishes changes of group configurations via the messaging system such that the Aggregation service (and potentially other services) are notified about these reconfigurations. The Aggregation service is designed in a way that, when receiving reconfigurations, it immediately starts aggregating measurements according to the new group structure. Further, as aggregations are performed on measurement time and not on processing time, it supports reprocessing historical data.

6.3 Temporal aggregation

Both types of temporal aggregations discussed in Section 5.3 are supported by the Titan Control Center. As both types serve different purposed, they are implemented in individual microservices. Both services subscribe to input streams, which provide monitored power consumption from sensors as well as aggregated power consumption for groups of machines.

Aggregating Tumbling Windows The History microservice receives incoming power consumption measurements and continuously aggregates all data items within consecutive, non-overlapping, fixed-sized windows. The results of these aggregations are stored to an Apache Cassandra (Lakshman and Malik 2010) database as well as published for other services. The History service supports aggregations for multiple different window sizes in parallel, allowing to generate time series with different resolutions. To prevent the amount of stored data from becoming too large, time series of different resolutions are assigned different *times to live*. Thus, the Titan Control Center allows, for example, to store raw measurements captured with high frequency for only one day, but aggregated values in minute resolution for years. Window sizes and times to live can be individually configured according to requirements for trackability and availability of storage infrastructure.

Aggregating Temporal Attributes The Statistics microservice aggregates power consumption measurements by a temporal attribute (e.g., day of week) to determine an average course of power consumption, for example, per week or per day. These statistics are continuously recomputed, stored in a Cassandra database, and published for other services, whenever new input data arrives. In our studied pilot cases we found out that in particular the average consumptions over the day, the week, and the entire year allow to detect pattern in the consumption. Furthermore, aggregating temporal attributes such as the month of the year over one year allows to observe how monthly peak loads evolve over time.

6.4 Correlation

The Titan Control Center provides different features for correlating power consumption data. One of these features is graphical correlation of power consumption of different machines or machine groups. Our visualization component (see Section 6.7) provides a tool, which allows a user to compare the power consumption of multiple consumers in time series plots (see Fig. 5). It displays multiple time series plots below each other, each containing multiple time series. The user can zoom into the plots and shift the displayed time interval. All charts are synchronized by the time domain, thus zooming or shifting one plot also effects the others (Johanson et al. 2016). This tool allows operators to analyze interesting points in time (such as outtakes or load peaks) in more detail.

Together with the newspaper printing company, we implemented a first proof of concept for correlating real-time production data with power consumption data. We correlated the printing machines' power consumption with their printing speed. For this purpose, we integrated the production management system using the Titan Flow Engine and visualized both types of data in our visualization component. Even though we were able to show the feasibility of such a real-time correlation, we identified that for in-depth analyses, power consumption data with higher accuracy is required. Similarly, we prototypically correlated the power consumption of air conditioning systems with weather data. We identified a high impact of the outside temperature on the power consumed for cooling and, thus, use weather data as a feature for our forecasting implementations (see Section 6.6).

6.5 Anomaly detection

The Titan Control Center envisages individual microservices for independent anomaly detection tasks and, hence, allows to choose an appropriate technique for each task. This includes individual techniques for different production environments and even for different machines.

With our pilot implementation, we already provide an Anomaly Detection microservice, which detects anomalies based on summary statistics of the previous power consumption. These statistics (e.g., per hour of week) are continuously recomputed by the Stats microservice (see Section 6.3) for each machine and machine group and published via the Control Center's messaging system. Our Anomaly Detection microservice subscribes to this statistics data stream and joins it with the stream of measurements (from real machines or aggregated groups of machines). Ultimately, this means each incoming measurement is compared to the most recent summary statistics of the corresponding point in time and machine.

Fig. 5 Screenshot showing the graphical correlation of power consumption using the Titan Control Center

If the measured power consumption deviates to much from the average consumption of the respective hour and weekday, it is considered as an anomaly. More precisely, for a measurement x and summary statistics providing the arithmetic mean μ and standard deviation σ, the service computes the absolute distance from the arithmetic mean $d = |x - \mu|$ and tests if $d < k\sigma$, where k is the configurable number of standard deviations. All detected anomalies are again published to a dedicated data stream via the messaging system, allowing other microservices to access detected anomalies. Moreover, the microservice stores all detected anomalies in a Cassandra database.

The currently implemented method for detecting anomalies is rather simple. It does not require complex model training or manual modeling, but is not able to consider trends, seasonality over larger time periods, or external variables. We are working on extending our pilot implementation, in order to join the measurement stream with the data stream published by the forecasting service (see Section 6.6). This implementation will consider measurements as anomalies if they deviate too much from the prediction, which is a common approach for anomaly detection.

6.6 Forecasting

Similar to anomaly detection, we envisage individual Forecasting microservices for different types of forecasts, for example, used for different power consumers. Forecasting benefits notably from the microservice pattern since technologies used for forecasting often differ from the ones used for implementing web systems. The Titan Control Center supports arbitrary Forecasting microservices, each using its own technology stack. The only requirement for a Fore-casting service is that it is able to communicate with other services via the messaging system.

Our pilot implementation already features a microservice that performs forecasts using an artificial neural network with TensorFlow (Abadi et al. 2016). This neural network is trained offline using historical data and mounted into the microservice at start-up. During operation, the Forecasting microservice subscribes to the stream of measurements (again monitored or aggregated) and feeds each incoming measurement into the neural network. The forecast results are stored in an OpenTSDB (The OpenTSDB Authors 2018) time series database and published to a dedicated stream via the messaging system.

In a first proof of concept, we build and trained such neural networks together with the newspaper printing company. We selected a set of machines in the company with different power consumption patterns and trained individual networks per machine. These neural networks use not only the historical power consumption of their machines as input, but also the power consumption of other machines as well as environmental data, such as the outside temperature. We deploy individual instances of our Forecasting microservice for each neural network, allowing for individual forecasts of each machine.

6.7 Visualization

As suggested in Section 5.7, the Titan Control Center features web applications for visualizing power consumption data. Since visualization serves as a measure to integrate the results of other measures, we also regard the visualization software components as integration of the individual analysis microservices. The Titan Control Center provides two single-page applications for visualization: a graphical

Fig. 6 Screenshot of the Titan Control Center
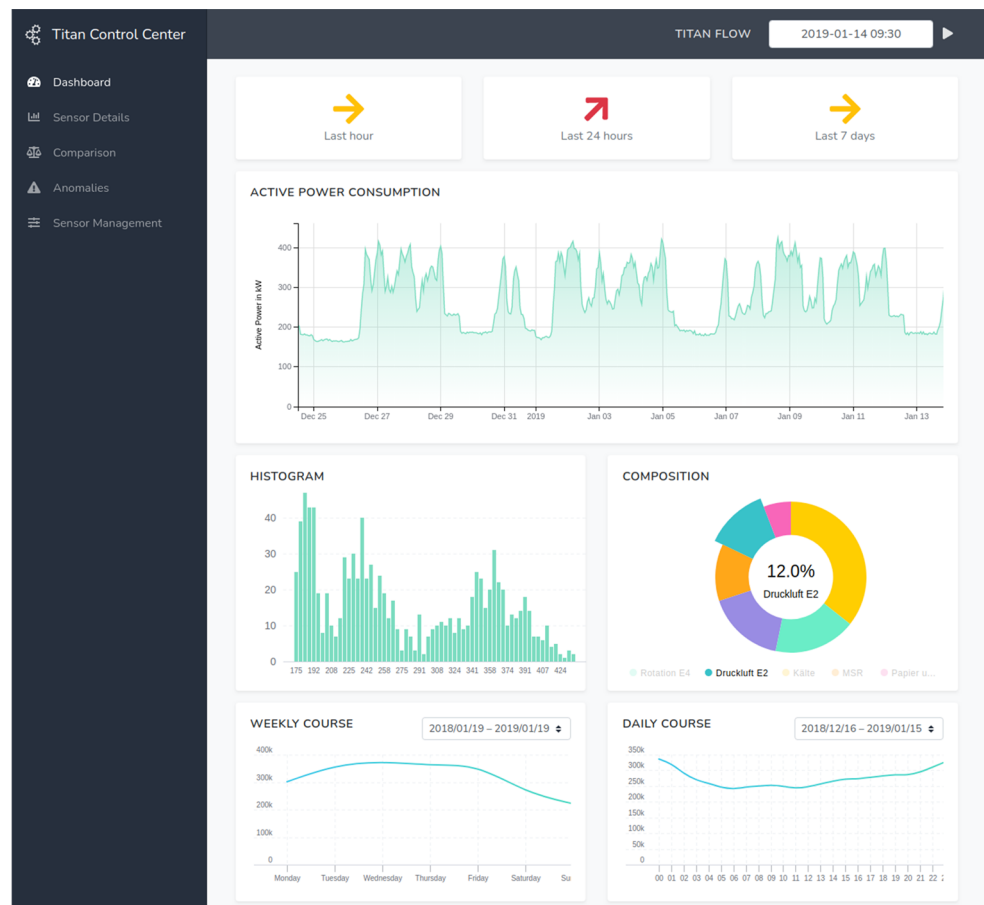
user interface, tailored to the specific functions of the Titan Control Centers, and a dashboard for simple, but highly adjustable data visualizations. In the following, we describe both applications and their corresponding use cases.

Control Center The Titan Control Center user interface[2] serves to provide a consistent access to all functionalities of the Titan Control Center. This includes visualizing the analysis results of microservices, but also control functions for configuring microservices. The user interface is implemented with Vue.js (You 2019) and D^3 (Bostock et al. 2011).

Figure 6 shows a screenshot of the Titan Control Center's summary view. It consists of several components which collect and show the individual analysis results for the entire production. A time series chart displays the power consumption in course of time. This chart is interactive, allowing to zoom and shift the displayed time interval. Colored arrows indicate how the power consumption evolved within the last hour, the last 24 hours, and the last 7 days. A histogram shows a frequency distribution of metered values serving to detect potential for load peak reduction. A pie chart breaks down the total power

consumption into subconsumers. Line charts display the average course of power consumption over the week or the day, as provided by the Statistics microservice (see Section 6.3). The visualizations are periodically updated with new data. This causes, for example, the time series diagram to shift forward continuously and the arrows to change color and direction.

Apart form this summary view, our pilot implementation also provides the described types of visualization for individual machines and groups of machines. Starting from an overview of the total power consumption, a user can thus navigate through the hierarchy of all consumers. Furthermore, the single-page application allows to graphically correlate data (see Section 6.4) and to configure machines and machines groups maintained by the Sensor Management service. Visualizations of forecasts and detected anomalies are currently under development.

Dashboard The second application is a pure visualization dashboard implemented with Grafana (Grafana Labs 2020) (see Fig. 7). It provides a set of common visualizations such as line charts, bar charts, and gauges. As presented in Fig. 7, we mainly display time series charts as bar or line charts. The dashboard is highly adjustable, meaning that users

---

[2]We provide a public show case of the Titan Control Center at http://samoa.se.informatik.uni-kiel.de:8185.

Fig. 7 Screenshot of the Titan dashboard implemented with Grafana (Wetzel 2019)

can add, modify, and rearrange chart components. Such adjustments can be performed graphically and only require usage of provided interfaces. Thus, especially IT savvy production operators can customize dashboards. Moreover, they can create own dashboards and share them among users. In this way individual dashboards, for example, for management and production operators can be implemented.

In contrast to the Control Center, this dashboard does not provide any control functions (e.g., for sensor configuration) and no complex interactive visualizations (e.g., the comparison tool). Thus, it only serves as an extension to the Control Center, allowing for visual analysis and reporting. In particular, this dashboard covers use cases, where power consumption data should be integrated in existing dashboards (as it is the case in one studied enterprise) or if dashboards should be customized by production operators.

6.8 Alerting

Altering in the Titan Control Center is implemented using the Titan Flow Engine in the integration component. All messages that are published to the messaging system can again be consumed by the Titan Flow Engine and processed in flows. This way, production operators can create and adjust alerting flows directly within the production environment. Our pilot implementation already provides a flow that sends an email whenever an anomaly in power consumption is reported. In dedicated bricks, the operator can filter the types of anomaly an alert should be generated for and configure how the email should be sent (e.g., message and receiver). The flow engine allows to model flows that perform arbitrary actions in the production environment when alerts are received. This includes communications with machines again, for example, to show alerts on machine monitors.

7 Conclusions and future work

In a pilot study with two manufacturing enterprises, we identify that analyzing power consumption data promises to achieve goals of categories such as reporting, optimization, fault detection, and predictive maintenance. In an additional literature review, we observe that research of various disciplines suggests measures for achieving these goals. Based on this literature review and expert knowledge within our

pilot cases, we suggest to implement the following measures in software for achieving these goals: real-time data processing, multi-level monitoring, temporal aggregation, correlation, anomaly detection, forecasting, visualization, and alerting. Finally, we show how microservices, stream processing, and fog computing can serve for implementing the proposed measures in a power consumption analytics platform.

For future work, we plan to take advantage of the modular architecture of the Titan Control Center by extending our pilot implementations. In particular, ongoing research focuses on developing more precise forecast and anomaly detection approaches as well as detailed visualizations. Further, we plan to conduct extensive evaluations in our studied enterprises.

Declarations

Conflict of Interests The authors declare that they have no conflict of interest.

References

Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin M, Ghemawat S, Irving G, Isard M, Kudlur M, Levenberg J, Monga R, Moore S, Murray DG, Steiner B, Tucker P, Vasudevan V, Warden P, Wicke M, Yu Y, Zheng X (2016) TensorFlow: A system for large-scale machine learning. In: Proceedings of the 12th USENIX conference on operating systems design and implementation, USENIX Association, USA, OSDI'16, pp 265–283

Albadi MH, El-Saadany EF (2008) A summary of demand response in electricity markets. Electr Power Syst Res 78(11):1989–1996. https://doi.org/10.1016/j.epsr.2008.04.002

Arora S, Taylor JW (2016) Forecasting electricity smart meter data using conditional kernel density estimation. Omega 59:47–59. https://doi.org/10.1016/j.omega.2014.08.008. Business Analytics

Bernstein D (2014) Containers and cloud: From LXC to Docker to Kubernetes. IEEE Cloud Computing 1(3):81–84. https://doi.org/10.1109/MCC.2014.51

Bischof S, Trittenbach H, Vollmer M, Werle D, Blank T, Böhm K (2018) HIPE: An energy-status-data set from industrial production. In: Proceedings of the ninth international conference on future energy systems, ACM, New York, NY, USA, e-Energy '18, pp 599–603

Bonomi F, Milito R, Zhu J, Addepalli S (2012) Fog computing and its role in the Internet of Things. In: Proceedings of the first edition of the MCC workshop on mobile cloud computing, ACM, New York, NY, USA, MCC '12, pp 13–16

Bostock M, Ogievetsky V, Heer J (2011) D^3 data-driven documents. IEEE Trans Vis Comput Graph 17(12):2301–2309. https://doi.org/10.1109/TVCG.2011.185

Bundesamt für Wirtschaft und Ausfuhrkontrolle (BAFA) (2020) Merkblatt stromkostenintensive Unternehmen 2020. Brochure. http://www.bafa.de/SharedDocs/Downloads/DE/Energie/bar_merkblatt_unternehmen.pdf, Accessed: 2020-12-29

Bunse K, Vodicka M, Schönsleben P, Brülhart M, Ernst FO (2011) Integrating energy efficiency performance in production management – gap analysis between industrial needs and scientific literature. J Clean Prod 19(6):667–679. https://doi.org/doi/10.1016/j.jclepro.2010.11.011

Cagno E, Worrell E, Trianni A, Pugliese G (2013) A novel approach for barriers to industrial energy efficiency. Renew Sust Energ Rev 19:290–308. https://doi.org/10.1016/j.rser.2012.11.007

Carbone P, Katsifodimos A, Haridi S (2019) Stream window aggregation semantics and optimization. In: Sakr S, Zomaya AY (eds) Encyclopedia of big data technologies, Springer, pp 1615–1623

Chandola V, Banerjee A, Kumar V (2009) Anomaly detection: A survey. ACM Comput. Surv. 41(3):15:1–15:58. https://doi.org/10.1145/1541880.1541882

Chou J-S, Telaga AS, Chong WK, Gibson GE (2017) Early-warning application for real-time detection of energy consumption anomalies in buildings. J Clean Prod 149:711–722. https://doi.org/10.1016/j.jclepro.2017.02.028

Chou J-S, Telaga AS (2014) Real-time detection of anomalous power consumption. Renew Sust Energ Rev 33:400–411. https://doi.org/10.1016/j.rser.2014.01.088

Chujai P, Kerdprasop N, Kerdprasop K (2013) Time series analysis of household electric consumption with arima and arma models. In: Proceedings of the international multiconference of engineers and computer scientists, pp 295–300

Cooremans C, Schönenberger A (2019) Energy management: A key driver of energy-efficiency investment? J Clean Prod 230:264–275. https://doi.org/10.1016/j.jclepro.2019.04.333

Cugola G, Margara A (2012) Processing flows of information: From data stream to complex event processing. ACM Comput. Surv. 44(3):15:1–15:62. https://doi.org/10.1145/2187671.2187677

Din GMU, Marnerides AK (2017) Short term power load forecasting using deep neural networks. In: 2017 International conference on computing, networking and communications (ICNC), pp 594–598

Fiedler T, Mircea P-M (2012) Energy management systems according to the ISO 50001 standard – challenges and benefits. In: 2012 International conference on applied and theoretical electricity (ICATE), pp 1–4

Garcia Lopez P, Montresor A, Epema D, Datta A, Higashino T, Iamnitchi A, Barcellos M, Felber P, Riviere E (2015) Edge-centric computing: Vision and challenges. SIGCOMM Comput. Commun. Rev. 45(5):37–42. https://doi.org/10.1145/2831347.2831354

Goodhope K, Koshy J, Kreps J, Narkhede N, Park R, Rao J, Ye VY (2012) Building LinkedIn's real-time activity data pipeline. IEEE Data Eng. Bull. 35:33–45

Grafana Labs (2020) Grafana. https://grafana.com/grafana, Accessed: 2020-12-29

Hasselbring W, Henning S, Latte B, Möbius A, Richter T, Schalk S, Wojcieszak M (2019) Industrial DevOps. In: 2019 IEEE

International conference on software architecture companion, pp 123–126

Hasselbring W, Steinacker G (2017) Microservice architectures for scalability, agility and reliability in e-commerce. In: 2017 IEEE International Conference on Software Architecture Workshops (ICSAW), pp 243–246

Henning S, Hasselbring W (2020) Scalable and reliable multidimensional sensor data aggregation in data-streaming architectures. Data-Enabled Discovery and Applications 4(1):1–12. https://doi.org/10.1007/s41688-020-00041-3

Henning S, Hasselbring W (2021) The Titan Control Center for Industrial DevOps analytics research. Software Impacts 7:100050. https://doi.org/10.1016/j.simpa.2020.100050

Henning S, Hasselbring W, Möbius A (2019) A scalable architecture for power consumption monitoring in industrial production environments. In: 2019 IEEE International conference on fog computing, pp 124–133

Henning S, Hasselbring W (2019) Scalable and reliable multidimensional aggregation of sensor data streams. In: 2019 IEEE International conference on big data, pp 3512–3517

Herman J, Herman H, Mathews MJ, Vosloo JC (2018) Using big data for insights into sustainable energy consumption in industrial and mining sectors. J Clean Prod 197:1352–1364. https://doi.org/10.1016/j.jclepro.2018.06.290

Herrmann C, Thiede S (2009) Process chain simulation to foster energy efficiency in manufacturing. CIRP J Manuf Sci Technol 1(4):221–229. https://doi.org/10.1016/j.cirpj.2009.06.005

International Energy Agency (2019) World energy balances 2019

ISO 50001 (2018) Energy management systems – Requirements with guidance for use. Standard, International Organization for Standardization, Geneva, CH

Jadhav AR, Kiran MPRS, Pachamuthu R (2021) Development of a novel iot-enabled power- monitoring architecture with real-time data visualization for use in domestic and industrial scenarios. IEEE Trans Instrum Meas 70:1–14. https://doi.org/10.1109/TIM.2020.3028437

Johanson A, Flögel S, Dullo C, Hasselbring W (2016) OceanTEA: Exploring ocean-derived climate data using microservices. In: Proceedings of the sixth international workshop on climate informatics, NCAR Technical Note NCAR/TN, pp 25–28

Jovanović B, Filipović J (2016) ISO 50001 standard-based energy management maturity model – proposal and validation in industry. J Clean Prod 112:2744–2755. https://doi.org/10.1016/j.jclepro.2015.10.023

Kanchiralla FM, Jalo N, Johnsson S, Thollander P, Andersson M (2020) Energy end-use categorization and performance indicators for energy management in the engineering industry. Energies 13(2):369. https://doi.org/10.3390/en13020369

Kreps J (2014) Questioning the Lambda architecture. https://www.oreilly.com/radar/questioning-the-lambda-architecture, Accessed: 2020-12-29

Kreps J, Narkhede N, Rao J (2011) Kafka: A distributed messaging system for log processing. In: Proceedings of 6th international workshop on networking meets databases, Athens, Greece

Lakshman A, Malik P (2010) Cassandra: A decentralized structured storage system. SIGOPS Oper. Syst. Rev. 44(2):35–40. https://doi.org/10.1145/1773912.1773922

Lasi H, Fettke P, Kemper H-G, Feld T, Hoffmann M (2014) Industry 4.0. Business & Information Systems Engineering 6(4):239–242. https://doi.org/10.1007/s12599-014-0334-4

Liu X, Nielsen PS (2018) Scalable prediction-based online anomaly detection for smart meter data. Inf Syst 77:34–47. https://doi.org/10.1016/j.is.2018.05.007

Martínez-Álvarez F, Troncoso A, Asencio-Cortés G, Riquelme J (2015) A survey on data mining techniques applied to electricity-related time series forecasting. Energies 8(11):13162–13193. https://doi.org/10.3390/en81112361

Marz N, Warren J (2015) Big data: Principles and best practices of scalable realtime data systems, 1st edn. Manning Publications Co., USA

Masoodian M, Buchwald I, Luz S, André E (2017) Temporal visualization of energy consumption loads using time-tone. In: 2017 21st International conference information visualisation (IV), pp 146–151

Masoodian M, Lugrin B, Bühling R, André E (2015) Visualization support for comparing energy consumption data. In: 2015 19th International conference on information visualisation, pp 28–34

Miragliotta G, Shrouf F (2013) Using Internet of Things to improve eco-efficiency in manufacturing: A review on available knowledge and a framework for IoT adoption. In: Emmanouilidis C, Taisch M, Kiritsis D (eds) Advances in production management systems. Competitive manufacturing for innovative products and services, Springer Berlin Heidelberg, Berlin, Heidelberg, pp 96–102

Mohamed N, Al-Jaroodi J, Lazarova-Molnar S (2018) Energy cloud: Services for smart buildings. In: Rivera W (ed) Sustainable cloud and energy services: principles and practice, Springer International Publishing, pp 117–134

Mohamed N, Al-Jaroodi J, Lazarova-Molnar S (2019) Leveraging the capabilities of Industry 4.0 for improving energy efficiency in smart factories. IEEE Access 7:18008–18020. https://doi.org/10.1109/ACCESS.2019.2897045

MongoDB I (2019) Mongodb. https://www.mongodb.com, Accessed: 2020-12-29

Morrison JP (2010) Flow-based programming, 2nd edition: A new approach to application development. CreateSpace, Paramount, CA

Newman S (2015) Building microservices, 1st edn. O'Reilly Media, Inc., Newton

Pfandzelter T, Bermbach D (2019) IoT data processing in the fog: Functions, streams, or batch processing? In: 2019 IEEE International conference on fog computing (ICFC), pp 201–206

Qi Q, Tao F (2019) A smart manufacturing service system based on edge computing, fog computing, and cloud computing. IEEE Access 7:86769–86777. https://doi.org/10.1109/ACCESS.2019.2923610

Quiroz JC, Mariun N, Mehrjou MR, Izadi M, Misron N, Radzi MAM (2018) Fault detection of broken rotor bar in ls-pmsm using random forests. Measurement 116:273–280. https://doi.org/10.1016/j.measurement.2017.11.004

Rackow T, Javied T, Donhauser T, Martin C, Schuderer P, Franke J (2015) Green Cockpit: Transparency on energy consumption in manufacturing companies. Procedia CIRP 26:498–503. https://doi.org/10.1016/j.procir.2015.01.011. 12th Global Conference on Sustainable Manufacturing – Emerging Potentials

Rist T, Masoodian M (2019) Promoting sustainable energy consumption behavior through interactive data visualizations. Multimodal Technologies and Interaction 3(3):56. https://doi.org/10.3390/mti3030056

Sahal R, Breslin JG, Ali MI (2020) Big data and stream processing platforms for industry 4.0 requirements mapping for a predictive maintenance use case. J Manuf Syst 54:138–151. https://doi.org/10.1016/j.jmsy.2019.11.004

Sax MJ, Wang G, Weidlich M, Freytag J-C (2018) Streams and tables: Two sides of the same coin. In: Proceedings of the international workshop on real-time business intelligence and analytics, ACM, New York, NY, USA, BIRTE '18, pp 1:1–1:10

Schulze M, Nehler H, Ottosson M, Thollander P (2016) Energy management in industry – a systematic review of previous findings

and an integrative conceptual framework. J Clean Prod 112:3692–3708. https://doi.org/10.1016/j.jclepro.2015.06.060

Sequeira H, Carreira P, Goldschmidt T, Vorst P (2014) Energy Cloud: Real-time cloud-native energy management system to monitor and analyze energy consumption in multiple industrial sites. In: 2014 IEEE/ACM 7th International conference on utility and cloud computing, pp 529–534

Shrouf F, Ordieres J, Miragliotta G (2014) Smart factories in Industry 4.0: A review of the concept and of energy management approached in production based on the Internet of Things paradigm. In: 2014 IEEE International conference on industrial engineering and engineering management, pp 697–701

Shrouf F, Gong B, Ordieres-Meré J (2017) Multi-level awareness of energy used in production processes. J Clean Prod 142:2570–2585. https://doi.org/10.1016/j.jclepro.2016.11.019

Shrouf F, Miragliotta G (2015) Energy management based on Internet of Things: practices and framework for adoption in production management. J Clean Prod 100:235–246. https://doi.org/10.1016/j.jclepro.2015.03.055

Shrouf F, Ordieres-Meré J, Garcí?a-Sánchez A, Ortega-Mier M (2014) Optimizing the production scheduling of a single machine to minimize total energy consumption costs. J Clean Prod 67:197–207. https://doi.org/10.1016/j.jclepro.2013.12.024

Singh S, Yassine A (2018) Big data mining of energy time series for behavioral analytics and energy consumption forecasting. Energies 11(2):452. https://doi.org/10.3390/en11020452

Szydlo T, Brzoza-Woch R, Sendorek J, Windak M, Gniady C (2017) Flow-based programming for IoT leveraging Fog Computing. In: 2017 IEEE 26th International conference on enabling technologies: infrastructure for collaborative enterprises (WETICE), pp 74–79

Tesch da Silva FS, da Costa CA, Paredes Crovato CD, da Rosa Righi R (2020) Looking at energy through the lens of Industry 4.0: A systematic literature review of concerns and challenges. Computers & Industrial Engineering 143:106426. https://doi.org/10.1016/j.cie.2020.106426

The OpenTSDB Authors (2018) Opentsdb. http://opentsdb.net, Accessed: 2020-12-29

Thollander P, Paramonova S, Cornelis E, Kimura O, Trianni A, Karlsson M, Cagno E, Morales I, Jiménez Navarro JP (2015) International study on energy end-use data among industrial smes (small and medium-sized enterprises) and energy end-use efficiency improvement opportunities. J Clean Prod 104:282–296. https://doi.org/10.1016/j.jclepro.2015.04.073

Vijayaraghavan A, Dornfeld D (2010) Automated energy monitoring of machine tools. CIRP Ann 59(1):21–24. https://doi.org/doi.org/10.1016/j.cirp.2010.03.042

Vikhorev K, Greenough R, Brown N (2013) An advanced energy management framework to promote energy awareness. J Clean Prod 43:103–112. https://doi.org/10.1016/j.jclepro.2012.12.012

Wang G, Koshy J, Subramanian S, Paramasivam K, Zadeh M, Narkhede N, Rao J, Kreps J, Stein J (2015) Building a replicated logging system with Apache Kafka. Proc. VLDB Endow. 8(12):1654–1655. https://doi.org/10.14778/2824032.2824063

Wetzel DB (2019) Entwicklung eines Dashboards für eine Industrial DevOps Monitoring Plattform. Bachelor's Thesis, Kiel University

Yan J, Meng Y, Lu L, Li L (2017) Industrial big data in an Industry 4.0 environment: Challenges, schemes, and applications for predictive maintenance. IEEE Access 5:23484–23491. https://doi.org/10.1109/ACCESS.2017.2765544

Yang C-T, Chen S-T, Liu J-C, Liu R-H, Chang C-L (2020) On construction of an energy monitoring service using big data technology for the smart campus. Clust Comput 23:265–288. https://doi.org/10.1007/s10586-019-02921-5

You E (2019) Vue.js. https://vuejs.org, Accessed: 2020-12-29

Zhang Y, Ma S, Yang H, Lv J, Liu Y (2018) A big data driven analytical framework for energy-intensive manufacturing industries. J Clean Prod 197:57–72. https://doi.org/10.1016/j.jclepro.2018.06.170

Zheng J, Xu C, Zhang Z, Li XL (2017) Electric load forecasting in smart grids using long-short-term-memory based recurrent neural network. In: 2017 51st Annual conference on information sciences and systems (CISS), pp 1–6