

Providing a Sustainable, Adaptive IT Infrastructure for Portable Micro-CHP Test Benches

Dominik Schöner, Richard Pump, Henrik Rüscher, Arne Koschel, Volker Ahlers

Abstract During the transition from conventional towards purely electrical, sustainable mobility, transitional technologies play a major part in the task of increasing adaption rates and decreasing range anxiety. Developing new concepts to meet this challenge requires adaptive test benches, which can easily be modified e.g. when progressing from one stage of development to the next, but also meet certain sustainability demands themselves.

The system architecture presented in this paper is built around a service-oriented software layer, connecting a modular hardware layer for direct access to sensors and actuators to an extensible set of client tools. Providing flexibility, serviceability and ease of use, while maintaining a high level of reusability for its constituent components and providing features to reduce the required overall run time of the test benches, it can effectively decrease the CO₂ emissions of the test bench while increasing its sustainability and efficiency.

Keywords Battery Electric Vehicles • Portable Micro-CHP unit • Test Bench • SOA • Adaptive IT Infrastructure • Sustainable Software Engineering

1 Introduction

In 2007, the German government published their Integrated Energy and Climate Programme (IKEP) [2], defining its climate protection goals until 2020, which describes electric mobility as a factor in the process of reaching those goals. The target of one million electric vehicles on German roads by 2020 [11] meanwhile was – as of 1. January 2016 – still 844,133 vehicles away, according to the Kraftfahrt-Bundesamt (German Federal Motor Transport Authority) [7], resulting in an adoption rate of 0.3 % for electric and hybrid passenger cars.

D. Schöner (✉), R. Pump, A. Koschel, V. Ahlers
University of Applied Sciences and Arts Hannover, Faculty IV, Department of Computer Science, Ricklinger Stadtweg 120, 30459 Hannover, Germany
e-mail: dominik.schoener@hs-hannover.de

H. Rüscher
University of Applied Sciences and Arts, Faculty II, Department of Mechanical Engineering, Bismarckstraße 2, 30173 Hannover, Germany

To alleviate some of the issues of range anxiety in battery electric vehicles (BEV), transitional concepts combining electric motors with internal combustion engines have been developed. Hybrid electric vehicles for example use their battery for assistive purposes only and charge it via recuperation. Range extenders meanwhile are small-scale combustion engines integrated into BEV to reduce charge depletion rates, but – unlike hybrid vehicles – are usually not capable of providing enough energy for propulsion without battery support.

Since 2013 an interdisciplinary research project at the University of Applied Sciences and Arts Hannover is studying a new concept evaluating the scalability of mobile micro combined heat and power units [14]. The goal is to replace conventional range extender with a unit scaled in accordance with the BEV's thermal requirements. This allows the cogeneration of heat and power by using the thermal output of the combustion engine as well as its torque, increasing fuel efficiency and reducing CO₂ emissions compared to other range extender concepts [8]. The unit's small scale [5] and a compact design make it portable allowing its removal from the BEV for use in buildings [12] as well as other applications like outdoor activities.

After having started out with a 15 kW test bench for the initial evaluation of the portable micro-CHP (pmCHP) concept and optimising the unit's design space [5], the project is in the process of finalising a scaled-down prototype.

As the test bench hardware changed across different stages of development, the software infrastructure had to adapt with only minor modifications to maintain a high degree of usability for all members of the interdisciplinary team. For this purpose, a reusable software platform for test beds was developed along with firmware for its controller, a web-based frontend and a LabVIEW-based control station [15].

The major contribution of this publication is the detailed description of key concepts and the internal workings along with implementation details behind this platform's service infrastructure and how they help to increase its sustainability. The following pages give a short overview of the system's architecture in section 3, explaining the requirements imposed on the system and the services involved along with their internal and external communication interfaces. A more detailed description of three key services for test bed configuration, test run replay and balance group evaluation follows in section 4, detailing internal structures and processes. The paper concludes with section 5, providing a summary of what has already been achieved and an outlook over topics for potential future research.

2 Related Work

The basis for the test bench infrastructure presented in this paper was published in [15], with a focus on usability and the interdisciplinarity of its usage. It provides an overview of the entire test bed, while a more detailed description of key components and implementation details is presented in this paper. The concept of employing pmCHP units as range extenders in BEV and buildings was published in [8], with [13] further investigating the scalability of the concept for use in buildings.

Like the evolution of the hardware for drivetrain testing – e.g. comparing textbooks from 1999 [6] and 2014 [10] – the hard- and software to control and monitor them evolved as well. An example of this can be found in [16], presenting a test bed centred around a single-board hard- and software bundle for modelling, monitoring and control, bringing simplicity, albeit at a potential loss of hardware flexibility.

As a means of software-based flexibility and adaptivity, [10] suggests the use of SOA, which uses standardised interfaces to increase the speed and reduce the cost of development. In [1], a service-oriented reference architecture for smart grid integration is proposed, in which standardised interfaces also play an important role due to diversity of devices relevant to the smart grid. This becomes especially important to the development process, once prototype status is reached and the integration of the pmCHP into smart homes and other scenarios moves into focus, requiring support for standards like e.g. IEC 61850-7-420 [4].

Meanwhile another factor to be considered especially in research test benches for sustainable technologies, is the sustainability of the test bench infrastructure itself. The definition of the term ‘Green IT’ covers a wide range of topics related to the environmental impact of IT, but also including social and economic factors [9]. A differentiation is made in [3] between ‘Green in IT’, which aims to reduce the environmental footprint of IT itself and ‘Green by IT’, evaluating means by which IT can become a tool to reduce environmental impacts from other sources. The considerations in this paper therefore regard sustainability as a goal to be achieved by creating a reusable IT infrastructure – making the soft- and hardware itself more sustainable – and by using it to reduce emissions of the actual test bench.

3 System Architecture

The first iteration of the test bench consisted of the 15 kW unit which had to be controlled manually via a switchboard directly connected to the respective actuators and with sensor readings taken by a data logger. While this was already sufficient for running many of the initial test scenarios, this setup only allowed for offline analysis of test runs with only few live readings being available via gauges.

Although some machine failures could be prevented by experienced personnel due to noise levels and several fail-safe mechanisms, fault analysis relying on measurements was only possible post-mortem, making reliable test bench operation by inexperienced personnel (e.g. students) infeasible. Due to data being recorded offline in CSV format and manual actuator operation, the immediate effects of minor alterations could also only be inspected after completing a test run, potentially leading to a high number of reruns for the same test scenario.

A set of requirements with the goal of increasing test bench efficiency and sustainability by reducing personnel requirements as well as decreasing the overall number of scenario reruns is therefore presented in the following subsection.

3.1 Requirements

To increase test bench usability, one of the main requirements is to provide near real-time sensor readings to its operators (RQ1) as well as the capability to – automatically as well as manually – transmit commands to actuators (RQ2). As an additional safety measure, the system should be able to monitor certain metrics for thresholds to prevent emergency situations by signalling the operator on exceedance as well as enacting predefined procedures in case of critical exceedance (RQ3). These measures are intended to allow safe and reliable test bench operation e.g. for students, without permanent supervision by experienced personnel, and run under near real-time conditions. All tasks with hard real-time requirements, like engine control, must be handled by external components (e.g. the controller boards [15] (see Fig. 1)), which are not in the scope of this paper.

Since unnecessary reruns of test scenarios result in additional wear on test bench components as well as CO₂ emissions, the system needs to constantly record every sensor reading and actuator command (RQ4) to be able to replay them in a manner similar to performing a live test run (RQ5). This includes the capability to skip through and pause the replay as well as re-evaluating correction formulae – used to compensate for different tolerances and characteristics of sensors – and balance groups – collecting data from multiple sensors to balances between inputs and outputs – to avoid reruns due to erroneous post-processing of raw sensor data.

The configuration of balance groups, sensors and actuators, along with their characteristics and locations in the test bench, due to its experimental nature, is subject to frequent changes. Therefore, the system must be easily reconfigurable to reflect these changes without interfering with its capability to replay older recordings or requiring extensive knowledge of programming (RQ6).

In addition to this configuration level adaptivity, the software architecture of the system should impose a high level of modularity, allowing for individual components to be replaced, removed or added as work on the project progresses (RQ7). This is especially important when progressing from one stage of development to the next, as this usually incurs switching to a completely new test bench, which would otherwise induce time intensive redevelopment of complex software components, in turn decreasing the sustainability of the software. Combined with the reconfigurability of the system, its modularity also allows its continued use in the long term e.g. for practical courses or student projects, giving the entire test bench a sustainable perspective even beyond the current research project.

3.2 System Context

Before going further into detail on the software components in section 3.3, the system context diagram in figure 1 provides an overview of the test bench environment. It can be divided into test benches – consisting of the controller boards, sensors,

actuators and the machine itself – and three different categories of front-end applications (control station, visualization and configuration interface) with the service infrastructure acting as mediator and containing the business logic.

The *controller boards* form the boundary between the physical sensors/actuators and the IT infrastructure and provide a basic, binary, TCP/IP-based interface. A client – e.g. test bench services – can initiate a connection to a board, creating a command and a data channel, similar to conventional FTP. While the command channel is used to transmit actuator commands (see RQ2) and controller-specific settings (e.g. sample rate) from the client to the test bench, the data channel transports a stream of sensor readings from the test bench to the client (see RQ1).

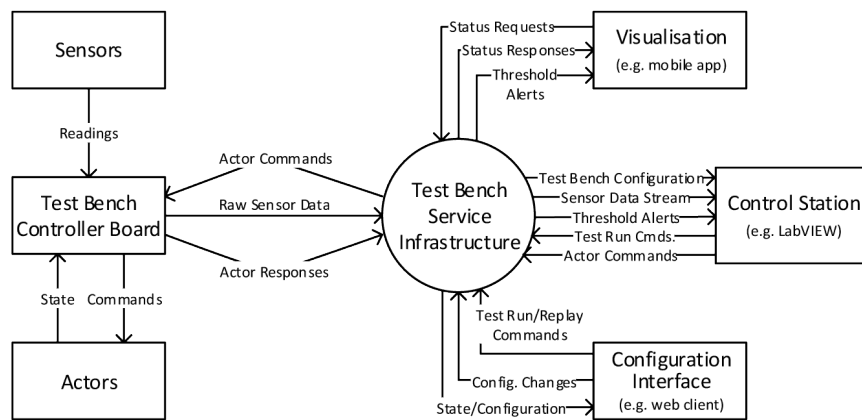


Fig. 1 System context diagram showing an overview of the test bench environment with all external components surrounding the service infrastructure.

The *control station* must be provided with the test bench configuration and a constant stream of processed and enriched sensor data (see RQ1) and expects to be able to transmit commands for actuators and initiate and stop test runs (see RQ2). Like the visualization, it must also be notified of threshold alerts to keep the operator informed about potential alerts (see RQ3), although the visualization only serves informational purposes and requests status updates on its own without any control capabilities. The configuration interface – a web-based application in this case – provides the operator with a graphical front-end reflecting the current state and setup of the test bench and allowing changes to be made to both (see RQ6).

3.3 Service Architecture

Internally, the service infrastructure is made up of a total of eight different services (fig. 2), whose orchestration is handled by *CurrentValues*, which also provides processed sensor and actuator readings to external clients (see RQ1). Additional core services are provided by *Persistence* – storage and retrieval of entities such as settings and measurements (see RQ4) to and from a database or similar data storage –

and *Configuration*, which handles data on the sensor and actuator setup of the test bench as well as arbitrary settings for all other services (see RQ6).

All the other services are optional or at least replaceable components – e.g. the *Connection* service used to connect to the controller boards – for which drop-in replacements grant connectivity to different boards (see RQ7). The only service with direct access to *Configuration* – aside from *CurrentValues*, receiving all raw sensor readings for processing – is *Actuator* (see RQ2), which offers interfaces for controlling all actuators retrieved from *Configuration* and can be disabled if the test bench is only to be monitored or no actuators are present.

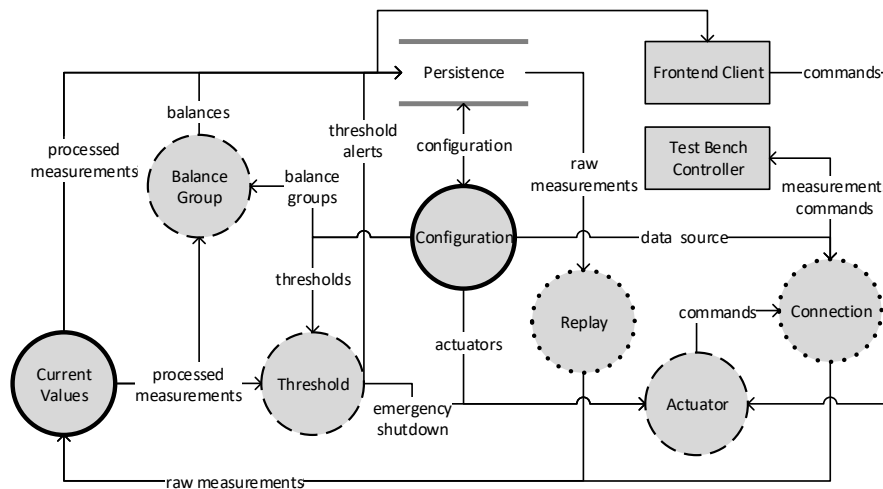


Fig. 2 Overview of the service architecture with data flows between the different services (circles) and external entities (rectangles) with core services with bold, optional with dashed and exchangeable with dotted borders.

Replay is an on-demand, drop-in replacement for *Connection*, used when running replays instead of live test runs (see RQ5) and therefore also has no connection to *Actuator*. The *Balance* service retrieves balance groups from *Configuration* and monitors all processed readings from *CurrentValues* to keep them up-to-date and available to clients. *Threshold* also monitors these readings, checking them for violations of configurable hard and soft limits (see RQ3) and can be deactivated e.g. for benchmark scenarios, to allow intentional exceedance of safety limits.

3.4 Communication Protocols and Interfaces

While internal communication as well as interfaces made available to external clients are based on RESTful APIs with JSON payloads being transmitted via HTTP, communication with the test bench controllers is handled by the controller-dependent implementation of *Connection*. This allows the use of custom binary protocols,

which can be tailored to each individual controller's firmware, reducing computational requirements for communication on the embedded hardware.

The use of well documented RESTful interfaces for all other communication allows for a high degree of extensibility regarding the integration of additional services and new clients e.g. for simulation (see RQ7). This makes the entire architecture more resilient against changing requirements across all development stages from experimental to prototype test bench to actual field tests, since the software can easily be extended by adding new services. Using a service-oriented architecture also helps to limit the impact of major changes to specific services.

4 Configuration and Persistence

As mentioned in section 3.3, *Configuration* is an integral part of the test bench service infrastructure, as it manages configurations for all other services, providing a key feature for the flexibility of the entire system. The *Replay* service, although not critical to the system itself, plays a key role in its sustainability, reducing the CO₂ emissions of the test bench by avoiding unnecessary reruns of scenarios. It is being supported in this task by the highly reconfigurable *Balance* service, which allows the customised computation of heat flows and summarisation in balance groups.

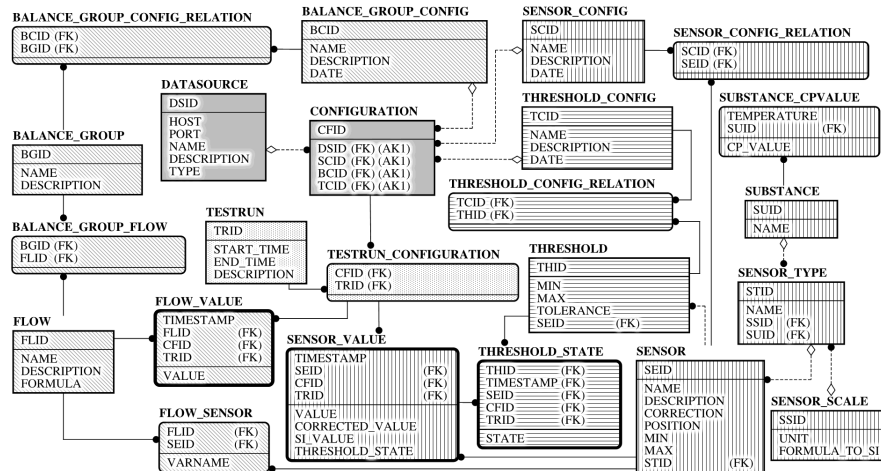


Fig. 3 ER diagram for the test bench configuration and measurement storage.

4.1 Configuration – Adapting Test Bench Changes

The most important feature of the *Configuration* service is the management of the test bench's sensor setup as can be seen the right part of figure 3 (vertical lines).

Using this model, arbitrary sensors can be created by configuring new sensor types, combining a substance and its specific heat capacity at constant pressure levels (C_p) with the SI unit and custom conversion formula for the sensor.

A set of sensors can then be combined into a sensor configuration, which in turn can be used together with configurations for other components of the system and a data source (test bench controller or *Replay*) to form a configuration useable for test runs. Since each optional service comes with its own configuration model, individual configurations can be easily recombined, reused and modified between runs and components (e.g. sensor types) of it even across different test benches.

The combination of configurations with test runs and the capability to create partial and full copies of configurations allows multiple version of them to be kept in storage, such that previous versions connected to previous test runs always remain available. This enables the system to replay archived test runs without running into complications due to changed hardware configurations in the test bench, while still making it possible to modify parts of the virtual configurations to evaluate these modifications before performing another live test run. In the ER diagram (fig. 3), the entities are marked with horizontal lines for *Threshold* and diagonal lines for *Balance*, with test run specific entities and readings highlighted by thicker borders.

4.2 Replay – Re-examining Recorded Test Runs

Although the data model in figure 3 stores raw sensor readings as well as derived values, the *Replay* service only uses the raw values when playing back a previously recorded test run. While this incurs higher computational loads than directly replaying the derived values instead, it also allows all conversion, heat flow and balance group formulae to be re-evaluated in the process, giving users the possibility to adjust and test them before performing another live run on the actual test bench. This not only saves time – thereby already increasing the sustainability of test bench operation – but also avoids unnecessary CO₂ emissions, further improving the test bench's ecological and economical sustainability.

Since all readings stored by *Persistence* contain a timestamp, *Replay* can also jump to an arbitrary point within a recorded test run, fast forwarding all readings taken before that point in time through the test bench's business logic. It is therefore possible to re-examine any time window within a recording without having to go through the entire duration of the test run first and pause execution at any given point e.g. when trying to trace back errors in balance calculations.

4.3 Balance – Keeping Track of Inputs and Outputs

Two key performance metrics in a CHP test bench are its thermal and fuel efficiency: How much of the heat generated by the ICE can be harnessed for heating

purposes and how much is lost as waste heat. And how much of the chemical energy stored in its fuel is converted into useable thermal and electrical energy and how does it compare to conventional RE and electrical heating elements.

The *Balance* service therefore provides the tools to analyse these metrics by combining multiple sensors to compute energy and material flows according to fully customisable formulae. These flows can then in turn be combined to form balance groups, giving precise insights into how much energy is lost at which point of the system or how much fuel is consumed to generate which output.

This in turn allows researchers to identify potential deficiencies in the current system and further optimise it to decrease its CO₂ emissions and increase its efficiency, making the concept more sustainable and providing an eco-friendlier alternative to conventional RE in BEV.

5 Conclusions and Perspectives

In this paper, it is shown how key features of the service components of the test bed architecture presented in [15] help to increase the sustainability of a pmCHP test bench and reduce its CO₂ emissions. This is achieved by creating a highly adaptive infrastructure with few core services – e.g. *Configuration* – which provide the needed flexibility and standardised RESTful interfaces for improved interoperability. Additional services provide features to record and replay live test runs with the possibility to reconfigure balance groups and formulae between replays. An overview of the data flows between individual services was given in fig. 2, with a detailed ER diagram in fig. 3 providing deeper insights into the flexibility provided by the data model behind the *Configuration* service.

While the infrastructure will remain in service in its current form for the foreseeable future, additional research is being made on further extending its feature set to increase its adaptivity as well as grant deeper insight into different usage scenarios of the pmCHP concept. The project is therefore currently evaluating the use of complex event processing to better detect, predict and prevent machine failures as well as integrate the pmCHP into a smart home or smart grid scenario.

Other directions for further research include integration of simulation components into the system or use of recorded test runs for the calibration of external simulations, as well as the development of a prototype system for real-world testing of pmCHP units in vehicles under aspects of scalability, performance and sustainability. This also includes the evaluation of pmCHP usage in fleets and car pools, but also for private use e.g. in recreational settings on board of caravans.

Acknowledgments This project was supported by the VolkswagenStiftung and the Ministry for Science and Culture of Lower Saxony (project funding number VWZN2891). We would like to thank the students who participated in the bachelor project building this service infrastructure and all our colleagues from the research focus ‘Scalability of mobile Micro-CHP units’ and the Institute for Engineering Design, Mechatronics and Electro Mobility (IKME) at the University of Applied Sciences and Arts Hannover for their support and the productive cooperation.

References

1. Appelrath, H.-J., Beenken, P., Bischofs, L., Uslar, M. (eds.): IT-Architekturentwicklung im Smart Grid. Perspektiven für eine sichere markt- und standardbasierte Integration erneuerbarer Energien. Springer Berlin Heidelberg, Berlin, Heidelberg (2012)
2. Bundesministerium für Wirtschaft und Energie: Eckpunkte von Meseberg für ein integriertes Energie- und Klimaprogramm. Die 29 Maßnahmen des IEKP. Meseberg, Deutschland. <http://www.bmwi.de/Redaktion/DE/Downloads/E/eckpunkt-fuer-ein-integriertes-energie-und-klimaprogramm.pdf> (2007). Accessed 26 Apr 2017
3. Calero, C., Piattini, M. (eds.): Green in Software Engineering. Springer International Publishing, Cham (2015)
4. Cleveland, F.M.: IEC 61850-7-420 communications standard for distributed energy resources (DER). In: 2008 IEEE Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century. Energy Society General Meeting, Pittsburgh, PA, USA, 20. t- 24.07.2008, pp. 1–4. IEEE (2008). doi: 10.1109/PES.2008.4596553
5. Hanif, H.I., Minnrich, J.P., Schmicke, C.R.P., Rüscher, H., Gusig, L.-O.: Bauraum- und gewichtstechnische Untersuchung einer mobilen mikro-PCU zur Bereitstellung von Wärme, Kälte und elektrischem Strom im E-Fahrzeug. In: Steinberg, P. (ed.) Wärmemanagement des Kraftfahrzeugs X. Energiemanagement, pp. 262–279. expert Verlag, Renningen (2016)
6. Hederer, A.: Elektromotorenprüfung. Computerunterstützte Elektromotorenprüfung mit klassischen und modellgestützten Verfahren für die Qualitätssicherung. Kontakt & Studium, vol. 536. expert Verlag, Renningen-Malmsheim (1999)
7. Kraftfahrt-Bundesamt: Der Fahrzeugbestand im Überblick am 1. Januar 2016 gegenüber 1. Januar 2015. Flensburg, Germany. http://www.kba.de/DE/Statistik/Fahrzeuge/Bestand/Ueberblick/2016/2016_b_ueberblick_pdf.html (2016). Accessed 26 Apr 2017
8. Minnrich, J.P., Rüscher, H., Schmicke, C.R.P., Gusig, L.-O.: Integration einer mikro-PCU im Thermomanagement eines Elektrofahrzeugs unter Berücksichtigung von Reichweite und Emissionen. In: Steinberg, P. (ed.) Wärmemanagement des Kraftfahrzeugs IX. Energiemanagement, pp. 324–340. expert Verlag, Renningen (2014)
9. Murugesan, S.: Harnessing Green IT. Principles and Practices. IT Prof. (2008). doi: 10.1109/MITP.2008.10
10. Paulweber, M., Lebert, K.: Mess- und Prüfstandstechnik. Antriebsstrangentwicklung ; Hybridisierung ; Elektrifizierung. Der Fahrzeugantrieb. Springer Vieweg, Wiesbaden (2014)
11. Presse- und Informationsamt der Bundesregierung: Deutschland soll Leitmarkt werden. Elektromobilität, Berlin, Germany. <https://www.bundesregierung.de/ContentArchiv/DE/Archiv17/Artikel/2013/05/2013-05-27-elektromobilitaet.html> (2013). Accessed 26 Apr 2017
12. Rüscher, H., Bitner, D., Saul, D., Guwy, A., Premier, G., Gusig, L.-O.: Application scenarios for a dual use of a portable micro-CHP unit in a BEV and building. Sustainability in Energy and Buildings: Research Advances Vol. 6 (2017) Submitted Mar 2017
13. Rüscher, H., Minnrich, J., Schmicke, C., Gusig, L.: Applicability and scalability of mobile mCHP units in mid-size battery electric vehicles and detached houses with different energy standards Vol. 4, 18–23 (2015)
14. Schmicke, C., Rüscher, H., Gusig, L.: Strategies for combined use of power conditioning units in vehicles and buildings. Sustainability in Energy and Buildings: Research Advances Vol. 3, 31–36 (2014)
15. Schöner, D., Pump, R., Schmicke, C., Minnrich, J.P., Rüscher, H., Ahlers, V., Koschel, A.: IT-Unterstützung von BHKW-Prüfständen in der angewandten Forschung. In: Mayr, H.C., Pinzger, M. (eds.) INFORMATIK 2016, Klagenfurt, Österreich, 26. - 30. Sept. 2016. Lecture Notes in Informatics, vol. 259, pp. 1227–1238. Gesellschaft für Informatik, Bonn (2016)
16. Schupbach, R.M., Balda, J.C.: A versatile laboratory test bench for developing powertrains of electric vehicles. In: 2002 IEEE 56th Vehicular Technology Conference Proceedings. 2002 IEEE 56th Vehicular Technology Conference, Vancouver, BC, Canada, 24 - 28 Sept. 2002, pp. 1666–1670. IEEE, Piscataway, NJ (2002). doi: 10.1109/VETECF.2002.1040499