

Towards an approach integrating various levels of data analytics to exploit product-usage information in product development

Klein, Patrick; van der Vegte, Wilhelm Frederik; Hribernik, Karl; Klaus-Dieter, Thoben

DOI

[10.1017/dsi.2019.269](https://doi.org/10.1017/dsi.2019.269)

Publication date

2019

Document Version

Final published version

Published in

Proceedings of the International Conference on Engineering Design, ICED

Citation (APA)

Klein, P., van der Vegte, W. F., Hribernik, K., & Klaus-Dieter, T. (2019). Towards an approach integrating various levels of data analytics to exploit product-usage information in product development. *Proceedings of the International Conference on Engineering Design, ICED, 2019-August*, 2627-2636.
<https://doi.org/10.1017/dsi.2019.269>

Important note

To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

TOWARDS AN APPROACH INTEGRATING VARIOUS LEVELS OF DATA ANALYTICS TO EXPLOIT PRODUCT-USAGE INFORMATION IN PRODUCT DEVELOPMENT

Klein, Patrick (1); van der Vegte, Wilhelm Frederik (2); Hribernik, Karl (3); Klaus-Dieter, Thoben (1)

1: University of Bremen; 2: Delft University of Technology; 3: BIBA - Bremer Institut für Produktion und Logistik GmbH

ABSTRACT

By applying data analytics to product usage information (PUI) from combinations of different channels, companies can get a more complete picture of their products' and services' Mid-Of-Life. All data, which is gathered within the usage phase of a product and which relates to a more comprehensive understanding of the usability of the product itself, can become valuable input. Nevertheless, an efficient use of such knowledge requires to setup related analysis capabilities enabling users not only to visualize relevant data, but providing development related knowledge e.g. to predict product behaviours not yet reflected by initial requirements.

The paper elaborates on explorations to support product development with analytics to improve anticipation of future usage of products and related services. The discussed descriptive, predictive and prescriptive analytics in given research context share the idea and overarching process of getting knowledge out of PUI data. By implementation of corresponding features into an open software platform, the application of advanced analytics for white goods product development has been explored as a reference scenario for PUI exploitation.

Keywords: Analytics, Simulation, Design methods, Semantic data processing, User centred design

Contact:

Klein, Patrick
University Bremen
Faculty of Production Engineering
Germany
klp@biba.uni-bremen.de

Cite this article: Klein, P., van der Vegte, W.F., Hribernik, K., Klaus-Dieter, T. (2019) 'Towards an Approach Integrating Various Levels of Data Analytics to Exploit Product-Usage Information in Product Development', in *Proceedings of the 22nd International Conference on Engineering Design (ICED19)*, Delft, The Netherlands, 5-8 August 2019.
DOI:10.1017/dsi.2019.269

1 INTRODUCTION

Usage information can be considered a valuable source of knowledge for predicting usage and behaviours of current and future products. Traditional ways of collecting product-usage information (PUI) include observation of human subjects, conducting user surveys and taking interviews (Park *et al.*, 2013). Also, researchers have equipped product units already owned by users with sensors and communication units to collect information (e.g., Yang *et al.*, 2007). However, now that information-collection capabilities and connectivity are increasingly becoming standard features of products, it becomes even easier for manufacturers to obtain PUI (Porter and Heppelmann, 2015), so that users of the same product can be compared with each other. The successful exploitation of PUI such as sensor data becomes a key success factor for future product developments

In the EU-funded FALCON project (2015-2017), we have investigated opportunities to exploit collected data from connected products in several ways. One main deliverable of the project was a ‘virtual open platform’ (VOP) to collect and process data generated by connected products and related social media, with the objective to extract actionable knowledge to be used as input for (re)design of products and related services (Hribernik *et al.*, 2017). This paper elaborates on our explorations to support product development with data analytics to improve anticipation of actual usage in redesigned products and related services, in particular by means of rule-based *design automation* and through *simulation*. These efforts have been reported on in separate, dedicated papers; here we have attempted to put these contributions into a broader framework comprising the full spectre of product-data analytics, and to make explicit which elements of that spectre have not been included or considered yet. In that sense it is an explorative paper rather than a research report for archival purposes.

All data gathered during the usage phase of a product and that relates to its usability and performance can be valuable input for the design of its successor. However, the acquired usage data from different sources such as social media, product-embedded information devices (PEIDs) and integrated sensors is usually referring to product instances, e.g. a temperature value from a device in a specific household. In addition, the data is usually collected over time-periods e.g. a recording of temperature values of a washing machine in household.

In contrast, information in product design (except make-to-order) traditionally refers to a product class, e.g. a product line, or series and those parameters, which are required as input for product design activities (calculations, dimensioning etc.) are usually represented by single values. Thus, as outlined exemplarily in the use-case (Section 4) statistic functions are used to match acquired data with design parameters. Further data and information may become relevant to other components than the ones related to which they are acquired. (Lützenberger *et al.*, 2016). For instance temperature monitoring used to control a fridge can be used as input for the design of insulation of a sibling fridge model and analytic functions will provide max and average temperature values typically used for dimensioning the insulation layer.

2 UTILISATION OF USAGE INFORMATION IN CONTEXT DATA ANALYTICS

By applying data analytics to PUI from combinations of different channels, companies can get a more complete picture of their products’ and services’ middle-of-life, which can help improve processes in other lifecycle phases such as product design. Nevertheless, an efficient use of such usage-knowledge requires the identification of the correct dependencies between PUI and design parameters, and appropriate data transformation.

A washing machine may serve as a sample: according to a leading European producer (collaborating with the authors in this field of research), it is possible to collect about 40 data streams from different sensors implemented in the washing machine and in addition about 40 status variables derived from the embedded control unit. Currently, the data is not structurally exploited neither for the improvement of new product generations nor for improving the software services belonging to the product. So far, the company has only implemented a feedback-loop to identify failures and their root causes. Thus, transferring PUI to design is currently a challenge of analysing and exploiting data-sets and not so much of sensor placement and data acquisition itself.

This status appears to be confirmed by literature findings. Most of the use cases focus on the identification of failures and their root causes, while the automated change of design parameters, is rarely addressed (Lehmhus *et al.*, 2015). For designers of IoT products and systems, no comprehensive,

academically underpinned methods are available to provide guidance in recognizing and exploiting the opportunities of data collection and processing technologies (Holler *et al.*, 2016). High-level workflows such as high-level cross-industry standard process for data mining (CRISP-DM, Jackson, 2002) do not address concrete design tasks and do not cover integration of data processing with physical systems. There is a lack of research for a systematic transfer of usage information (such as sensor data) into design in general, since an efficient use of such knowledge requires the setup of related descriptive, predictive or prescriptive analytics on the one hand and the identification of dependencies between usage-related parameters and design parameters or component attributes on the other. What is also missing is insight into the level of automation that can be achieved. Most of the literature seems to suggest that data mining, machine learning, etc. are highly automated but they also mention ‘manual’ activities in the processing chain and often presume IT skills.

In the remainder of the paper, we first provide general descriptions of the various steps that PUI-based analytics might entail, zooming in from a higher-level workflow that is supported by the literature, onto lower-level specific steps (Sections 2.1-2.3), then we will elaborate on what was implemented in our prototypical platform (Section 3) and how it was applied in two concrete business cases (Section 4). Based on these results, we will finally revisit our high-level model and present a detail-level assessment of what could actually be implemented and deployed, what level of automation could be realized and what challenges remain open for future work (section 5).

2.1 Data analytics and levels of sophistication in analytics

Analytics has evolved from business intelligence (BI) (Watson, 2014), which largely corresponds to descriptive analytics. It includes data warehousing, dashboards and end-user reporting focusing on historical data (Chaudhuri *et al.*, 2011), which is still the core functionality of conventional BI software (Watson, 2014). It also involves techniques such as descriptive statistics, exploratory visualization, hypothesis testing and querying, that are not considered data-mining (Kotu and Deshpande, 2015). Davenport (2006) introduced term business analytics to highlight a shift from descriptive statistics to predictive modelling. To reflect the increasing competences of analytics solutions, the ‘analytics maturity model’ suggested by Puget (2015) builds upon a model proposed by Gartner (cf. Linden *et al.*, 2017), in which human involvement in decision-making decreases with increasing maturity level. Accordingly, descriptive analytics reveals *was* happening. A step up in maturity produces insights from history: *diagnostics*, revealing causes, reasons and patterns as well as *predictions*, providing insight into what will happen. Both use the same statistics and machine-learning core technologies. Prescriptive analytics, at the highest maturity level, directly supports decision making. According to Puget (2015) and other sources (e.g., Delen and Demirkan, 2013), prescriptive analytics uses simulation, optimization and rules.

In the context of digitalised or connected products, Porter and Heppelmann (2015) concur that (i) descriptive analytics aims to capture a product’s condition, environment and operation, (ii) diagnostics examines causes of reduced product performance or failure, (iii) predictive analytics aims to detect patterns foretelling impending events and (iv) prescriptive analytics identifies improvement measures.

2.2 General workflow for data analytics based on PUI

The descriptive, predictive and prescriptive analytics functionalities in the given research context share the approach and overarching process of getting knowledge out of PUI data shown in Figure 1, which largely corresponds to processes suggested by Han *et al.* (2011) and Seliger *et al.* (2008), and which is typically executed by taking several different loops through its processing steps. Before any descriptive, predictive or prescriptive analytics is applied, the PUI has to be acquired and prepared. The first part of this workflow is supported by the data acquisition layer of the platform as described in section 4. The PUI can be PEID data or other middle-of-life data from social media and reports such as maintenance logs. The possible concrete contents of the next two steps, which are the actual descriptive and advanced analytics steps, will be explained in the next two sub-sections based on the literature.

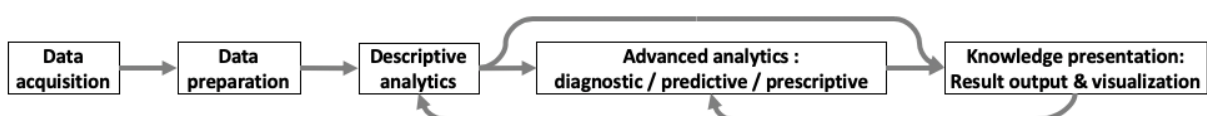


Figure 1. General workflow for analytics based on PUI

2.3 Descriptive analytics

Descriptive analytics is usually a combination of querying and data wrangling. Data wrangling (Kandel *et al.*, 2011) involves several transformations of the data set. From these, aggregation and filtering provide the most important contribution to descriptive analytics. Filtering may also use descriptive statistics functions such as mean, min/max or variance as its criterion (Kotu and Deshpande, 2015). The first iteration in data mining processes, also known as data exploration (ibid.), typically proceeds to the knowledge presentation step right after basic descriptive analytics (shown as a bypass in Figure 1). The results (e.g. tables and visualizations) allow the user to identify the advanced analytics tools to be used in the next iteration, to specify their parameters, and to adapt the set of wrangling and descriptive analytics operations.

2.4 Advanced analytics: diagnostic, predictive and prescriptive analytics

The terms prediction, diagnostics and forecasting are sometimes used as synonyms, but cover different aspects. Diagnostics, especially when applied to identify trends and patterns (according to Puget, 2015) is considered to be a form of forecasting by others (e.g. Armstrong, 1985). Prediction can be achieved through machine learning and through simulation. In machine learning, cross-sectional and longitudinal approaches can be distinguished (Armstrong, 2010). Using observed information, cross-sectional predictions target a value of something unobserved, if the variable to be forecast exhibits a relationship with one or more other predictor variables (Hyndman and Athanasopoulos, 2014). Examples of cross-sectional prediction techniques are regression, classification and clustering. Purely longitudinal approaches such as extrapolation and ARIMA, also referred to as time-series forecasting, use only information on the variable to be forecast, and do not attempt to discover the factors affecting its behaviour. Thus they will extrapolate only trends and recurring patterns (Hyndman and Athanasopoulos, 2014), although combinations using cross-sectional data are also possible.

Simulation as a prediction approach can be used to test the response of a model of the real world (e.g., a product model) to future stimuli that are either hypothesized (not based on real-life data such as PUI) or based on time-series forecasting of data such as PUI. As we have applied it as described in Section 4.2, simulation also can be used to 'predict' the response of a conceptualized future system to real-life stimuli in a prescriptive-analytics context.

Prescriptive analytics attempts to influence or recommend an action to be taken within a given time constraint in order to realize a desired outcome (Koch, 2015). By allowing modification of the simulation model, simulations become prescriptive and enables evaluation of design alternatives under the influence of PUI, thus generating actionable knowledge from PUI. 'Actionable' expresses that the knowledge is supposed to provide insight in how, given the actual usage, the product's design can be improved in terms of performance. Accordingly, prescriptive analytics includes generating rules to support decision making.

3 DATA ANALYTICS IN FALCON

In FALCON, *descriptive-analytics* support was mainly focused on aggregation and filtering. Aggregation can be performed based on one user-specified feature (e.g., consider all data by product ID) applies user-specified operations (e.g., min, max, average) to the aggregated data to generate knowledge about the other features (e.g., average of sensor reading x and maximum daily usages of function Y per product ID). Aggregation functions can also be applied to the whole data set (maximum of sensor reading z in all the data). Filtering, largely implemented as *querying*, returns data samples fulfilling a user-specified criterion formulated in terms of features (e.g., only the product units with sensor reading $x_{max} > a$ between t_i and t_j). In the platform this is supported by a business logic module (Figure 2). Established statistical functions have been incorporated for such a filtering and chosen with respect to needs of the designers and product developers (Lützenberger *et al.*, 2016). Afterwards the outcome of aggregation and filtering can be mapped to the design parameters of the product or to be more precise to the affected component. Regarding *advanced analytics*, support for prescriptive analytics was realized by coupling rule-based design automation to the analytics results (4.1). For prescriptive analytics through simulations and predictive/diagnostic analytics based on machine learning the project relied on external software - Matlab Simulink (4.2) and RapidMiner/Weka, respectively. Unfortunately, due to confidentiality considerations, we cannot report on the machine-learning results here.

3.1 Implementation and integration into an open platform

The modules which have been designed to enable the different levels of analytics have implemented to become an embedded part of an integrated software platform enabling to support real use cases in different application sectors. Figure 2 below shows a simplified architecture of the overall VOP, which is described in its implemented layers in the following to illustrate dependencies among data analytics and data acquisition. A more detailed overview of the platform was given by [Hribernik et al. \(2017\)](#).

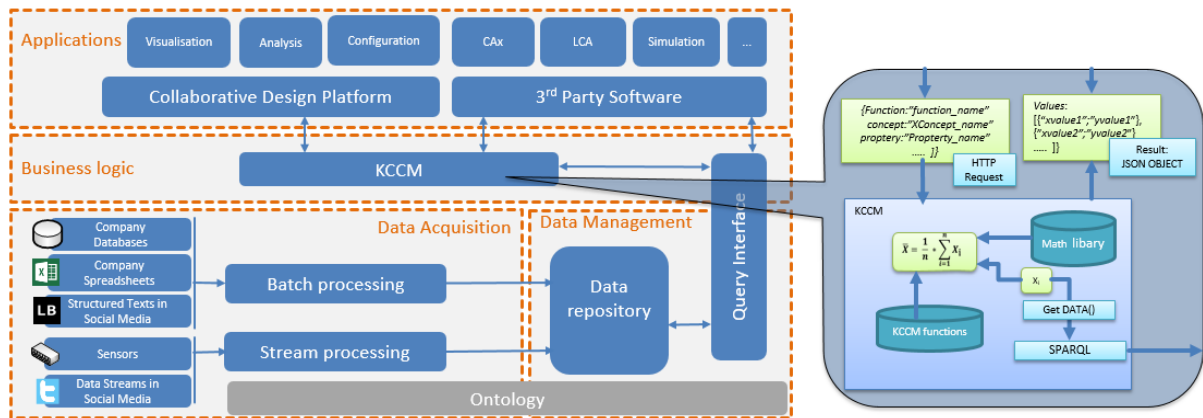


Figure 2. Simplified architecture of the FALCON virtual open platform

The lowest level provides modules for data acquisition and data access. Several acquisition modules connect the platform to different PUI sources. Batch processing allows connecting to sources including the company database, structured documents, e-commerce backends and semi-structured texts. Semi or unstructured text is processed using natural language processing as described in more detail in (Deng *et al.*, 2017). The usage data is structured and made interoperable within the VOP by means of an ontology, which constitutes an information model fundamental to the entire platform. All platform operations, such as searching for PUI, storing the PUI, querying, filtering and analysing it, can be performed with support of the ontology and its underlying description logics. In the case of a particular washing-machine model, a typical query could for instance be “Which program options of a particular washing machine model are never used?”. For this purpose, first the concepts were deposited in the ontology to store the existing functions and options of the respective washing machine model. Furthermore, concepts for usage data collection were deposited and linked semantically among each other and to the existing functions. The semantic links enable a simple request to return the list of unused options because data originating from the product development phase is related to the PUI. Without such a reference, in terms of only considering the usage data the question could not have been answered.

An ontology in itself cannot process information exchange and post-processing. This requires a suitable IT system that records PUI, links it to the ontology and integrates it into target systems, taking product development processes and applications into account. Further functionality provides semantic uplift and storage of PUI, that is, mapping it to the ontology and storing it in a data repository (a Triplestore). This way, PUI can be comprehensively queried in their semantic context and analysed, visualised or further processed by the business logic related components.

Since a direct usage of the PUI in product development processes or software is typically not meaningful or possible, the data is first brought into a form and aggregation level that can be handled by the software or by users. As an essential part of the business logic layer the Knowledge Consolidation & Cross sectoral Management module (KCCM) provides (i) functions for transformation of accessed data, filtering of accessed data and (ii) statistical and counting operations. In principle, these can be used by other modules and/or by different instances of the same module (in particular the visualization module), thus allowing a kind of service orchestration in preparation of front-end modules. The KCCM is foreseen to process as a backend module providing bundled features without an own graphical user interface.

The KCCM provide features (functions) for transformation of accessed data, the core ones of which are able to:

- express semantical neighbourhood of entities, exploring the benefits of the contextual relevance. (e.g. sensor data belonging to a specific washing program)

- transform state/appearance of information such as unit conversion, or percentage of a single item to a set of items enabling information aggregation from data sets
- refer to statistical/mathematical operations enabling information aggregation and descriptive analytics (Sample: Calculate Arithmetic Mean of sensor values; add a column with Fahrenheit temperatures converted from a column in Celsius, etc.)

While the first point (semantics) is covered by the query language capabilities (see below), for processing, the KCCM has been enriched with an additional Java library enabling mathematical or logical operations otherwise not commonly available in Java - the Apache Commons Mathematics Library. Of course, the KCCM module is not limited to this library and can be enhanced by other libraries providing for instance capabilities for further pattern recognition or trend analysis.

The KCCM module connects to the Query Interface (SPARQL) provided by the Data Management layer. Since the complexity of SPARQL Queries is encapsulated by the KCCM functions, no query-related programming skills are necessary to communicate with the KCCM module.

All functions of the KCCM's main interface, which is a REST (representational state transfer) compatible web service ([Battle and Benson, 2008](#)), can be accessed by using HTTP request tools, such as HTTP requester or RESTClient. The results can be output to all the platform's applications as well as to 3rd party software via an open API. The features are encapsulated as services and provided to the system based on the JavaScript Object Notation (JSON) standard. As outlined in Figure 2 a HTTP request can be sent to the KCCM module, and a JSON container is providing the results as a service afterwards (to query one of the specific functions a JSON object has to be defined as well).

Next to business logic functions, the platform offers a data export module that converts user-specified selections from the collected data to comma-separated values (CSV), a basic spreadsheet format that can be read by most 3rd party applications (ref. to section 4.2).

Tools and applications covering the functions typically requested by stakeholders are provided within a browser-based collaborative environment, which facilitates a synchronised, common view for multiple users working at different locations, who can filter, visualise and analyse PUI via an intuitive GUI. A dedicated modelling language called KbeML has been defined as an extension of SysML, which supports experts in modelling, communicating and documenting information flows and transformations related to their PUI-based development activities, allowing tracking and documenting PUI transformation and analysis operations ([Lützenberger et al., 2016](#)).

4 APPLICATION IN WHITE GOODS TEST CASES

This section only indicates the broad range of capabilities spanned by the general workflow for forecasting and simulation based on PUI and is due to the given limits not exhaustive.

4.1 Prescriptive analytics used for washing machine bearings selection

As most products, a washing machine is an assembly consisting of several components and sub-assemblies - for instance not only heating elements, and water filters, but also the washer drum and the main shaft with appropriate bearings. These components themselves are defined by design parameters such as drum diameter or distance of shaft shoulders. In the given example, the identification of appropriate bearings is based on PUI from embedded sensors, i.e., spin value, load weight, etc. In particular, the weight of the eccentric load of laundry during washings as well as the spin value (number of revolutions) are required for the calculation of bearing lifetime. To that end, a range of values derived from several washings can be used as input if preprocessed in advance, as is enabled by the KCCM module (section 3.1). It provides functionality to transform and analyse data, that can be applied in a prescriptive manner to any information collected beforehand, so that it can be exploited by product development, for example the maximum load depending on the spin speeds and on the selected washing program.

Based on descriptive input, i.e., the average of the array of centrifugal forces on the bearings, the bearing load is calculated. The forces acting on the bearings are identified using a free body diagram, which is transferred into prescriptive mechanical equations based upon the geometrical dependencies.

Without detailing the moment-balance system, it can be stated that in addition to the unbalanced force the product-related parameters (shaft dimensions) are required to have the system fully determined as well. The output - the forces acting on each bearing - are major input for calculating the lifetime of bearings (represented as bearing load P in the equations). In addition, the lifetime equation requires P

and three other parameters, C , L_h , and n . The dynamic load rating C is treated as an Output-Parameter, since this parameter enables the selection of an appropriate bearing in standard part catalogues. The lifetime L_h usually relates to product requirements can also be related to the related to PUI in terms of average lifetime of past products as well number of spins, n . Thus, the diagram (Figure 3) represents a complete formal description for the calculation and selection of appropriate bearings based upon descriptive analytics of usage data - its notation is based on KbeML (ref. to Lützenberger *et al.*, 2016).

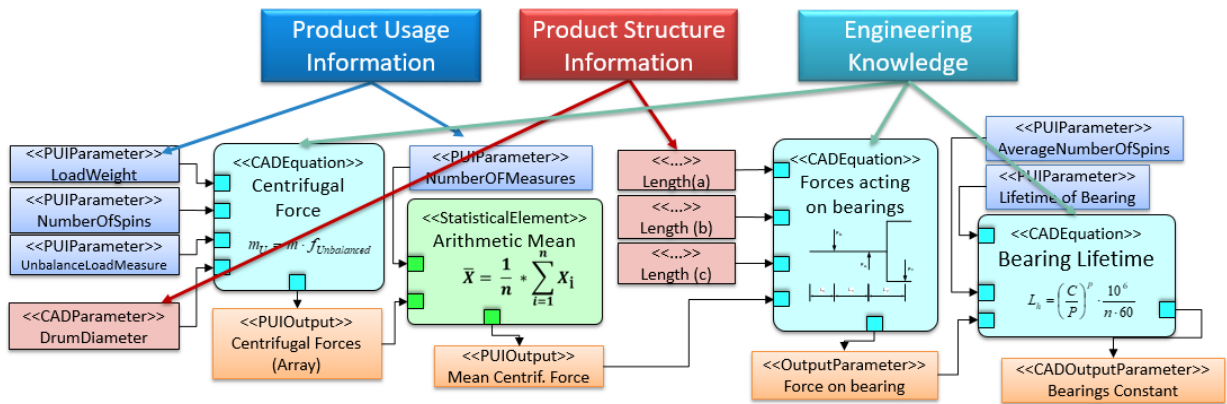


Figure 3. Formal description for the calculation of appropriate bearings using KbeML

While the bearing related engineering knowledge is common, the case underlines the potential of our approach to directly relate design parameters to selection of components and thus laying a profound base for the automated adaptation of a product design based on usage data.

4.2 Simulation of fridge-freezers based on PUI to support decision making in design

In product design, simulations are typically used to assess the performance of the design. To take into account the influence of user interactions, artificial cyclic load patterns are often applied, as if the user repeats the interaction with its average characteristics and average periodicity (e.g., Geppert, 2011). Real-life PUI offer the opportunity to perform more realistic simulations in which also irregularities in user behaviours and their timing are taken into account. To demonstrate the potential of simulation with PUI, a fridge-freezer was selected, from which the platform's data export module generated CSV files with timestamps and the PUI of interest from all connected units, after which the queried data was wrangled using external software (RapidMiner) to create simulation inputs representing individual product units.

We found that simulations make sense only if the timing of changes in the PUI actually influences the investigated performance aspects. In our case, the performance aspect *energy consumption* is dynamically influenced by the PUI item *door-opening timings* while the product is operating. For a washing machine (ref. to 5.1), on the other hand, the timing of the key interactions, program selection and laundry handling, does not influence performance - except when considering effects on a large time scale (e.g., seasonal influences on program selection) - and the time stamps have no added value.

Simulations can be deployed prescriptively to compare design alternatives generated by modifying the simulation model that are simulated with the same PUI from fielded units as input. To improve heat exchange inside, the appliance is equipped with fans. However, when the door is open, more heat is also exchanged with the warmer air outside if the fan is running. This can possibly be mitigated by additional control of the fan or by eliminating the fan. Thus, the design variations we decided to compare by exposing them to real usage information are: (i) no fan, (ii) a fan controlled based on compressor activity only and (iii) a controlled fan that also switches off if the door is open. Figure 4 illustrates how real-life data (PUI) produced by the VOP is used as input for simulation of which the outcomes reflect the effect of door openings that are represented in the PUI on energy consumption.

Running separate simulations for the two compartments, the energy consumption of each simulation run is recorded, and for each design variant the PUI-based simulation is compared to a reference simulation with no door openings. Consequently, for each variant the relative increase in energy consumption due to door-openings can be computed. So far, we ran simulations with sets of PUI from nine households using the appliance for a duration ranging from 15 to 47 days. The simulation model only uses. By using abstractions of geometric properties rather than a 3D model, we could run simulations at over 1,500×

real time and evaluate the multiple-day usage episodes within an hour. It could be shown that energy consumption figures resulting from the simulations reasonably correspond to real-life observations.

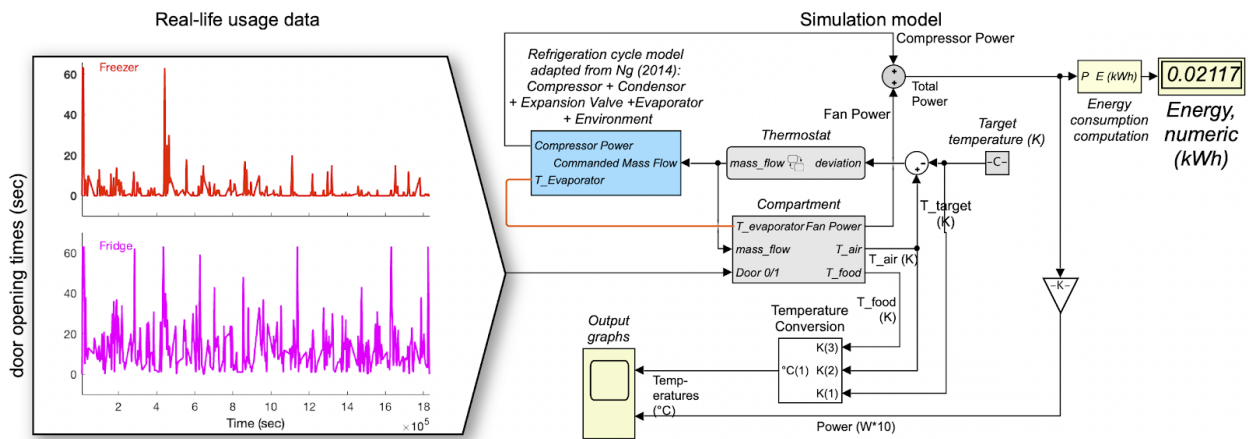


Figure 4. Simulation model and result for fridge-freezer simulation in MATLAB/Simulink

Comparing the design variants, using a fan expectedly turns out to save energy by about 2.6%. Depending on the design variant, simulations with door-opening PUI show a relative increase in energy consumption ranging from <0.1% up to 1.54%. In all cases, the relative increases are larger for the fridge compartment, and absolute increases are larger for the freezer compartment – which is responsible for 94% of the electricity consumption anyway. In the case of design variant (ii), door openings cost more energy: for the freezer with its dominant share in the energy consumption, on average, the relative influence of door openings increases by a factor of 20. Design variant (iii) turned out to be effective, reducing the influence of door openings by 40%, down from a factor of 20 for the fanless option to a factor of 12. Overall, for both compartments, this third design variant showed to be the most energy-efficient. Apparently, it optimally combines the positive effects of convection when the door is closed with mitigating the negative effect of door openings. The gains are relatively small, but they are not without impact considering the fact that domestic refrigeration is responsible for 4.5% of the total global electricity consumption (Harrington *et al.*, 2018). Using figures from the International Energy Agency (www.iea.org), we can illustrate these findings as follows: considering these design variants for all refrigerators worldwide, applying a fan corresponds to a CO₂ emissions reduction by an equivalent of about 9,400 cars, and the difference in impact of door openings between variant (ii) and (iii) would correspond to the emissions of 3,300 cars (assuming 13,000 km/y per car, 130g CO₂/km on average). As our main goal was to investigate the opportunities PUI-based simulation offers to compare design alternatives, we have not spent efforts in accurately mirroring the behaviour of a particular product design. Assuming that the usage data does not depend on design details, our investigations can be said to apply, in a comparative fashion, to a hypothetical design and variations on it. This way, we also did not have to expose company-confidential design information from the business cases. More details on the simulation setup and outcomes can be found in (Van der Vegte *et al.*, 2019).

5 DISCUSSION AND CONCLUSIONS

The possibility to exploit PUI by linking descriptive analytics to adjust design-related parameters and/or performing simulations is likely to have impact on the way future products will be designed. It is expected to lead to time savings. It does not require physical prototyping and recruiting human subjects for end-user testing and is therefore cost effective. However, automated knowledge discovery still seems to be a future promise, as most processes of navigating through the data-processing options still require several ‘manual’ steps. In Figure 5 we have zoomed in on the processes of Figure 1 and inserted the detail-level processes discussed in section 2.3 and 2.4, specifying their place in the actually realized workflows (Section 3 and 4) as input-output relations. The ‘no computer support yet’ steps show where ‘manual’ work is needed to enable the transition to the next steps. At the top of the diagram, we have also indicated that quite some system-user input is required, which also reveals absence of automation.

Going by its central position in Figure 5, perhaps the most crucial step lacking computer support is matching the available (pre-processed) data to the sought-after knowledge and the available advanced

analytics approaches. Actually, reasoning back from this match when it has been identified to the original data, or even a ‘tabula-rasa’ product where the data-collection options are still open, is not supported by any method that we know of, let alone design-support software. Other steps require external software (e.g., simulation and data-mining packages). Although incorporating that functionality into the platform could streamline the data-processing experience, it seems to be a rather unproductive proposition to “reinvent the wheel” and after that try to keep up with further developments in the already capable 3rd party software. Offering seamless interfacing with such software, in combination with supporting the matchmaking mentioned in the previous paragraph seems to be a better proposition.

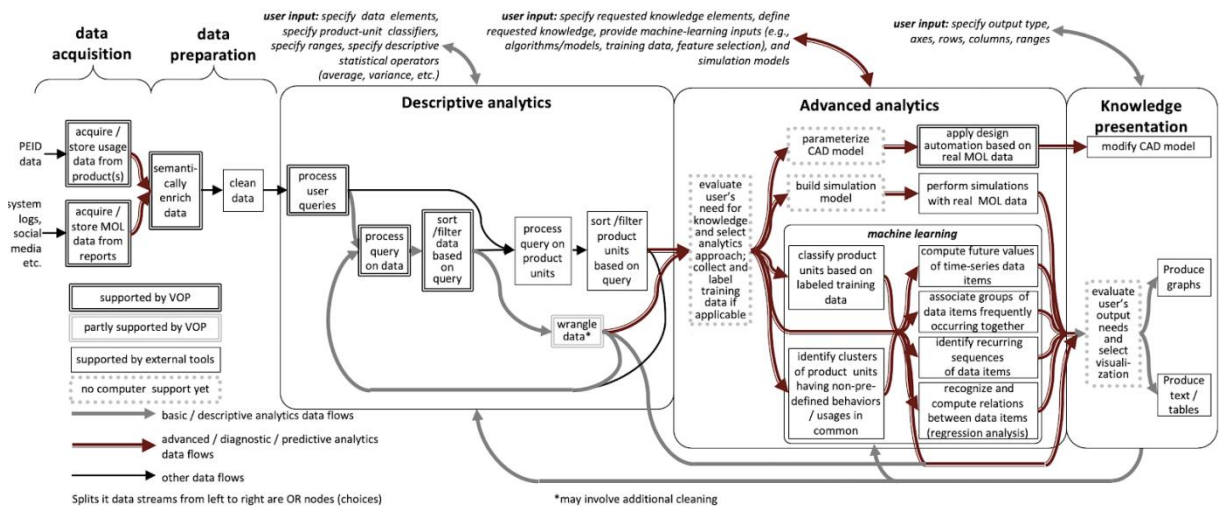


Figure 5. Detailed workflows for analytics based on PUI with current levels of support

Although we only investigated specific products, for which the obtained results are by no means spectacular, we can provide some general recommendations based on the work presented in this paper. Firstly, in designing each first generation of a product range to collect and transfer data, the data collection capacities to be included in the design must be anticipated. For example, a refrigerator could, unlike the one we simulated, also keep track of its own energy consumption so that simulations to study effects of user behaviour on energy consumption would only be useful if the current design has to be compared with alternative future redesigns only available as models.

Secondly, for product units out on the market, PUI-based simulations and design automation can be used to study effects of real-life usage. For instance, in the case of simulation, if certain end-user actions represented in the data appear to reduce performance (as in our case the door openings), comparison with reference data that lack these manifestations (in our case fictitious input with the door always closed) can quantify the effect. If it is serious enough, designers can ideate possible solutions to mitigate it, implement these in the simulation model and run simulations with the same real-life data to compare the effectiveness of the proposed solutions.

An improvement computed by design automation or a verified through simulation can be implemented in a redesign, or in some cases included in a software update for fielded products. PUI-based design automation and simulations will mostly facilitate (re)designing not-too-drastic variations on existing designs. After all, the usage-related input signals to the original product must also be meaningful in the modified product. If an opaque fridge door is replaced by a transparent one, it will no longer be opened just to check the contents, and the collected door data are likely no longer meaningful.

As our focus has been on supporting redesign, we have currently only considered off-line processing of PUI. In future work it might be useful to investigate to what extent elements of our approach can facilitate online, real-time product adaptation.

REFERENCES

- Armstrong, J.S. (2010), *Long-Range Forecasting*, 2nd Edition, Available at SSRN 666990. Wiley, New York.
 Armstrong, W.W. (1985), “The dynamics of articulated rigid bodies for purposes of animation”, *The Visual Computer*, Vol. 1, pp. 231–240.
 Banerjee, A., Bandyopadhyay, T. and Acharya, P. (2013), “Data analytics: Hyped up aspirations or true potential”, *Vikalpa*, Vol. 38, pp. 1–11.

- Battle, R. and Benson, E., (2008), "Bridging the semantic Web and Web 2.0 with Representational State Transfer (REST). Web Semantics: Science, Services and Agents on the World Wide Web", *Semantic Web and Web 2.0*, Vol. 6, pp. 61–69. <https://doi.org/10.1016/j.websem.2007.11.002>
- Chaudhuri, S., Dayal, U. and Narasayya, V. (2011), "An overview of business intelligence technology", *Communications of the ACM*, Vol. 54, pp. 88–98.
- Davenport, T.H. (2006), *Competing on analytics*. *harvard business review*, 84, Vol. 98.
- Delen, D. and Demirkan, H. (2013), "Data, information and analytics as services", *Decision Support Systems*, Vol. 55, pp. 359–363.
- Geppert, J. (2011), *Modelling of Domestic Refrigerators' Energy Consumption under Real Life Conditions in Europe*, Friedrich Wilhelms University, Bonn.
- Han, J., Pei, J. and Kamber, M. (2011), *Data mining: concepts and techniques*, Elsevier.
- Harrington, L., Aye, L. and Fuller, R.J. (2018), "Opening the door on refrigerator energy consumption: quantifying the key drivers in the home", *Energy Efficiency*, Vol. 11, pp. 1519–1539. <https://doi.org/10.1007/s12053-018-9642-8>
- Holler, M., Uebernickel, F. and Brenner, W. (2016), "Understanding the Business Value of Intelligent Products for Product Development in Manufacturing Industries", in: *Proceedings ICIME. Presented at the Proceedings of the 2016 8th International Conference on Information Management and Engineering, ACM*, pp. 18–24.
- Hribernik, K., Franke, M., Klein, P., Thoben, K.-D. and Coscia, E. (2017), "Towards a platform for integrating product usage information into innovative product-service design", in: *Proceedings of ICE/ITMC. Presented at the Engineering, Technology and Innovation (ICE/ITMC), 2017 International Conference on, IEEE*, pp. 1407–1413.
- Hyndman, R.J. and Athanasopoulos, G. (2014), *Forecasting: principles and practice*. OTexts.
- Jackson, J. (2002), "Data Mining; A Conceptual Overview", *Communications of the Association for Information Systems*, Vol. 8, p. 19.
- Kandel, S., Heer, J., Plaisant, C., Kennedy, J., van Ham, F., Riche, N.H., Weaver, C., Lee, B., Brodbeck, D. and Buono, P. (2011), "Research directions in data wrangling: Visualizations and transformations for usable and credible data", *Information Visualization*, Vol. 10, p. 271–288.
- Koch, R. (2015), "From business intelligence to predictive analytics", *Strategic Finance*, Vol. 96, p. 56.
- Kotu, V. and Deshpande, B. (2015), "Predictive analytics and data mining: concepts and practice with rapidminer", *Morgan Kaufmann*.
- Lehmhus, D., Wuest, T., Wellsandt, S., Bosse, S., Kaihara, T., Thoben, K.-D. and Busse, M. (2015), "Cloud-Based Automated Design and Additive Manufacturing: A Usage Data-Enabled Paradigm Shift", *Sensors (Basel)*, Vol. 15, pp. 32079–32122. <https://doi.org/10.3390/s151229905>
- Lützenberger, J., Klein, P., Hribernik, K. and Thoben, K.-D. (2016), "Improving Product-Service Systems by Exploiting Information From The Usage Phase. A Case Study. Procedia CIRP", *Product-Service Systems across Life Cycle*, Vol. 47, pp. 376–381. <https://doi.org/10.1016/j.procir.2016.03.064>
- Park, J., Han, S.H., Kim, H.K., Cho, Y. and Park, W. (2013), "Developing elements of user experience for mobile phones and services: survey, interview, and observation approaches", *Human Factors and Ergonomics in Manufacturing and Service Industries*, Vol. 23, pp. 279–293.
- Porter, M.E. and Heppelmann, J.E. (2015), "How smart, connected products are transforming companies", *Harvard Business Review*, Vol. 93, pp. 96–114.
- Puget, J.F. (2015), "Analytics Landscape", *IT Best Kept Secret Is Optimization*.
- Seliger, G., Gegusch, R., Müller, P. and Blessing, L. (2008), "Knowledge generation as a means to improve development processes of industrial product-service systems", In *Manufacturing Systems and Technologies for the New Frontier*, Springer, pp. 519–524.
- Van der Vegte, W.F., Kurt, F. and Sengöz, O.K. (2019), "Simulations based on real product-usage information to support redesign for improved performance: exploration of practical application to domestic refrigerators", *Journal of Computing and Information Science in Engineering*, Vol. 19 No. 3, <http://dx.doi.org/10.1115/1.4042537>.
- Watson, H.J. (2014), "Tutorial: Big data analytics: Concepts, technologies, and applications", *Communications of the Association for Information Systems*, Vol. 34, pp. 1247–1268.
- Yang, X., Moore, P.R., Wong, C.-B., Pu, J.-S. and Kwong Chong, S. (2007), "Product lifecycle information acquisition and management for consumer products", *Industrial Management and Data Systems*, Vol. 107, pp. 936–953.

ACKNOWLEDGMENTS

Part of this research was funded under the EC Horizon 2020 Programme, in the context of the FALCON project (<http://www.falcon-h2020.eu/>). The authors wish to acknowledge the Commission and all the FALCON project partners for the fruitful collaboration.