

Design Principles for Machine Learning Marketplaces in Enterprise Systems

Marek Hütsch

University of Duisburg-Essen, Germany
marek.huetsch@icb.uni-due.de

Tobias Wulfert

University of Duisburg-Essen, Germany
tobias.wulfert@icb.uni-due.de

Abstract

While standardized enterprise systems (ES) have become widely accepted, this is not the case for machine learning (ML) implementations, which are mostly developed individually in company-specific projects. Necessary historical data and rare ML capabilities result in a low cross-market ML utilization. To overcome the high usage barriers of ML, it should be incorporated into ES in a standardized manner. Therefore, we propose to implement an ML marketplace. While marketplaces in ES already exist, this paper proposes a marketplace dedicated to the exchange of ML models in a federated learning approach. Accordingly, this work formulates four meta-requirements based on interviews, which are structured by marketplace governance dimensions. With these meta-requirements, an ML marketplace was implemented in a design science research project, from which eight design principles are derived. The design principles address governance dimensions for making ML accessible to many companies and allow them to integrate ML into existing ES.

1. Introduction

Standardized enterprise software has become widely accepted in the area of enterprise systems (ES) [1, 2]. A large number of software modules and applications have been developed and made available by standard software vendors. Enterprise standard software, as opposed to enterprise individual software, plays a decisive role in the use of software solutions in companies [3]. In 2016, for example, the world's four largest software providers included three companies (i.e., Microsoft, Oracle, and SAP) the portfolios of which included a significant proportion of enterprise standard software [4]. Enterprise resource planning systems are known as standard software packages provided by software vendors and which are understood as a central part of ES [5]. Standardization in the ES sector has further increased due to the cloud efforts of software vendors

[2]. While standardization in the form of enterprise standard software provided by software vendors has become largely accepted, this has not been the case for ML software, especially ML models. The increasing importance of ML in companies can be justified by the fact that not only can ML be applied to routine, rule-based cognitive tasks [6, 7], it can also be applied to tasks heretofore thought of as nonroutine, including handwriting recognition and driving [8, 9]. As a result, ML can help to decrease costs and increase the value created by a given company. Robertson [10] stated that establishing dominant standards can help to reduce market penetration cycle times of innovations. A software marketplace can be more effective at implementing industry standards than simply developing individual or company specific ones [11]. Thus, an ML marketplace is helpful in achieving faster market penetration of ML standards across companies. Achieving a de facto standard of ML usage would mean that a large number of companies use ML in a similar way [12]. Probably caused by the minimal standardization of ML in ES, the market penetration of ML in companies is low [13]. There are a plethora of difficulties to deal with when introducing ML solutions into businesses, such as poor data quality, possible discrepancies between a company's business and technological goals, insufficient ML knowledge outside the development team, and a lack of ML expertise in the organization as a whole [14]. Furthermore, due to the extensive effort necessary in ML to process data and create models [15], considerable barriers exist that slow the adoption of ML by otherwise willing companies. This has resulted in few companies exploiting the benefits ML has to offer. When businesses lack ML solutions provided by software vendors and the necessary training data to create their own, they must generate and process their own data or purchase it from data vendors [16]. Such issues may be the cause of the observed current lack of ML solutions within most companies [17]. Standard software vendors can try to reduce these barriers by providing customers with

pre-trained ML models and integration of ML into their standard software packages. Therefore, we propose an approach, that centralizes the data pre-processing and model creation tasks in a dedicated ML marketplace. To use an ML solution in ES, a company could use ML models out of the box from a dedicated marketplace. Therefore the barriers of missing historical training data can be overcome since the models can perform well even if the company has little training data due to the ML concept of federated learning. In federated learning, models are trained in a collaborative way and exchanged between marketplace participants [18, 19]. So far, federated learning is mainly used in single company contexts (e.g., for mobile keyboard prediction [20]). In this work, the federated learning approach is proposed as a technique by which companies can share models via an ML marketplace. Therefore, this paper extends the existing research of Kumar et al. [21]. As we intend to introduce a federated learning marketplace in an inter-company approach, governance rules need to be established for collaboration and co-creation among participating companies. Governance has been identified as an important determinant of marketplace success [22, 23].

This work derives design principles (DPs) for the governance of ML marketplaces from the results of a design science research (DSR) project and interviews. To address this goal, we formulate the following research question: *What should the design principles for the governance of an ML marketplace aimed at achieving a fast market penetration of ML solutions in ES be?* The proposed DPs are prescriptive statements for the implementation of an ML marketplace [24]. This paper contributes to the overall research goal of making ML accessible to a mass market of companies that use ES. To increase the access to ML for companies without proper resources and capabilities, ML must be integrated into ES and provided by standard software vendors.

The remainder of our paper proceeds as follows: the background of federated learning and marketplaces in ES is presented in section two. Section three introduces our research design. In section four we present our meta-requirements (MRs) and DPs for an ML marketplace in ES. We discuss different manifestations of the DPs in section five and summarize our results in section six.

2. Background

In this paper, a computer program is said to learn from experience E, with respect to task class T and performance measure P, if its performance of the tasks

in T, as measured by P, improves with experience E [25]. A forecast task is defined as the use of historical time series data to make predictions about possible events in the future and is used as an example of ML marketplace implementation. ES is defined as a collection of software for enterprises that are delivered by standard software vendors [26]. Sawyer [27] compared packaged software (standard software) to custom information systems and noted that, unlike custom systems, packaged software has a high time to market pressure and its success is measured by its market share. In contrast, most current ML projects in ES are focused primarily on the optimization of existing business tasks [28].

It's important to note in the midst of this discussion about bringing ML solutions to a wide variety of companies that there are already several companies participating in knowledge sharing and transferring. The concept of federated learning was first introduced by Google [18, 19]. In contrast to this approach stands the traditional centralized ML techniques in which the local data sets are uploaded to one server and an ML model is trained by one centralized instance. Federated learning enables multiple actors to build a common, robust ML model without sharing data. Thus federated learning is able to satisfy concerns about critical issues such as data access rights, data privacy and data security. Furthermore, by Wei et al. data privacy was statistically produced in federated learning [29]. Thanks to this research it has been well established that competing companies can safely share information for the training of ML solutions. Federated learning makes use of the ML technique transfer learning. In transfer learning \mathcal{D}_s , which is used for the learning task \mathcal{T}_s , can also be helpful to learn task \mathcal{T}_t based on \mathcal{D}_t . Here, $\mathcal{D} = \{\mathcal{X}, P(\mathcal{X})\}$ consists of the the edge probability distribution $P(\mathcal{X})$ and the feature space \mathcal{X} . A task $\mathcal{T} = \{y, f(x)\}$ contains the target prediction function $f(x)$ and y , which is the result space. For learning task \mathcal{T}_t the predictive function $f_{\mathcal{T}}(\cdot)$ is improved by transferring knowledge from \mathcal{T}_s , \mathcal{D}_s or the predictive function $f_{\mathcal{S}}(\cdot)$. To allow for the transfer of parameters from the already trained model we explicitly add here $f_{\mathcal{S}}(\cdot)$ as a source to transfer knowledge, compared to Tan et al. [30]. For example, the forecast of new products is one area in demand forecasting where transfer learning is already being applied. New products have the issue of missing historical sales data, which Karb et al. [31] solves by determining similar products for new products and applying transfer learning in order to create an accurate forecast for the new product. This way Karb et al. [31] reduced the prediction variance. Furthermore, the presented models by Karb et al. [31] show improved

handling of the bias in the training data if unseen promotions occur for the first time.

ES can include marketplaces for the exchange of software modules. They establish a virtual location to conduct transactions between suppliers and consumers [32]. Marketplaces for ML models would be locations for the exchange of ML models provided by companies with sophisticated ML applications and sought out by companies introducing ML to their business. The ML models can be depicted as transactions of digital products [33]. As marketplaces necessarily involve two (or more) market sides, we draw upon the concept of two-sided markets [34]. Marketplaces match two or more previously distinct markets, reducing transaction costs by exploiting direct and indirect network effects and using them to further propel the multiple market sides [35]. Marketplace owners are able to exploit economies of scale because of marginal costs for adding more participants or an ML model are almost zero. Marketplaces can offer a variety of ML models provided by developers and other companies. The ML models can be described as the periphery of the marketplace while the core is the marketplace itself [36]. Kumar et al. [21] introduced the idea of a marketplace for ML models and placed this idea within existing theories of marketplaces. The authors reviewed existing ML marketplaces like the open-source decentralized ML marketplace TensorTask [21]. Although these marketplaces have not all been incorporated into an ES, there exist multiple marketplaces provided by ES software vendors like the SAP Store or Microsoft App Source. The focus of this research is on ML marketplaces for federated learning in ES that are conceptualized in economic terms as multi-sided markets [35]. In this context, Eisenmann et al. [37] introduced the concepts of marketplace provider and sponsor (marketplace owner) and participants from demand and supply side. While users from the demand side would request and download ML models from the ML marketplace, supply-side users would provide and share these models with others. The marketplace can be managed by its owner using a set of governance dimensions (GDs) that are direct and indirect measures to control the marketplace and participant interactions within an ecosystem [22]. The literature provides a number of structuring and aggregating GDs [38, 23, 39]. In this work, our research team followed the dimensions proposed by Hein et al. [38] with regard to governance structure, resources and documentation, accessibility and control, trust and perceived risk, pricing, and external relationships. We adopted these dimensions because they are the most comprehensive, being derived from diverse domains, and are thus applicable to ML marketplaces.

3. Research Design

DPs codify prescriptive knowledge specifying the design of an artifact (i.e., artificial thing [40]). They are statements, “that prescribe what and how to build an artifact in order to achieve a predefined design goal” [24, p. 4040], capturing, “knowledge about instances of a class of artifacts” [41, p. 39]. DPs are a codification of design knowledge that characterizes the artifact and the creation of the artifact [42]. DPs assist developers in establishing artifacts that fulfill predefined requirements and intended effects [43].

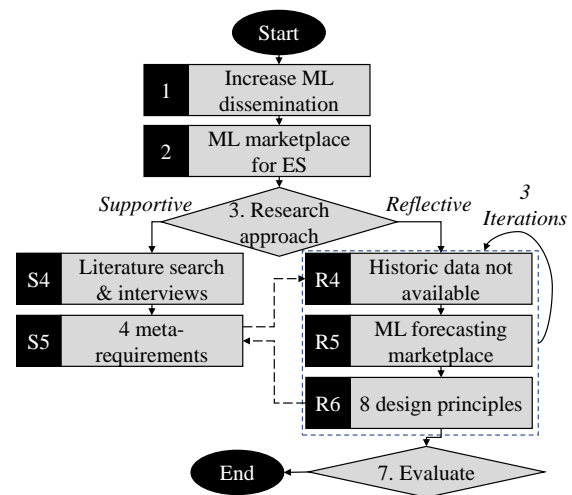


Figure 1. Research Approach [44]

We followed the research design as suggested by Möller et al. [44] to create DPs for ML marketplaces as depicted in Figure 1. (1) The objective of our research is to present a potentially successful method for cross-market penetration of ML in ES. To achieve this objective, we have proposed establishing an ML marketplace capable of enabling the exchange of ML models between companies. (2) The research context is already specified in the previous section, wherein we discussed the approach of federated learning for ML in ES and presented our focus on marketplaces. (3) We took both a supportive and reflective perspective for the derivation of our DPs [44]. (S4) The supportive elicitation of MRs consisted of a literature screening for existing marketplace GDs and qualitative expert interviews to identify the knowledge base. We conducted qualitative expert interviews as a major source of our MRs [45]. The interviews were organized in a semi-structured way according to the six GDs identified in our literature screening [38]. In total, we interviewed eleven interviewees in leading positions related to the implementation and operation of ES in

retail, consultant, and software vending companies. Each of the interviewees reported having substantial experience in the implementation of ML algorithms. The interview questions were scheduled to take 30 minutes. Because of the semi-structured format, we used the single interview duration was between 30 and 60 minutes with an average of 35 minutes (Table 1). All but one of the interview participants were interviewed only once. One managing director (I-1) was particularly involved in the creation of MRs, so the interviews conducted with that manager were divided into 6 appointments of 30 minutes each. There, one marketplace GD was discussed in each interview session. The same questions were asked to all interview participants. The format of semi-structured interviews allows for spontaneously diving deeper into specific topics of interest during the interview [45], which is the reason for the different lengths of the interviews. Due to the coronavirus pandemic and government restrictions, all of the interviews took place remotely. (S5) The transcribed interviews were coded using MAXQDA. Following Mayring [46], the content structuring approach was applied to the GDs and the results consisted of four MRs with a total of 857 coded text segments (Figure 2).

Table 1. Participants of the empirical study.

#	Type	Position	Duration
I-1	Owner	Managing Director	180 min
I-2	Owner	Managing Director	30 min
I-3	Customer	Technical Director	60 min
I-4	Customer	IT Project Manager	35 min
I-5	Customer	IT Project Manager	42 min
I-6	Customer	IT Project Manager	40 min
I-7	Owner	Consultant	35 min
I-8	Customer	IT Staff	30 min
I-9	Customer	IT Staff	37 min
I-10	Customer	IT Staff	35 min
I-11	Customer	IT Staff	30 min

(R4) The objective of the DSR project was to create a marketplace for the exchange of forecasting models, as company-internal data is often insufficient to train ML models. The four MRs and coded text segments defined the scope of our design process for the development of a prototypical marketplace for ML models. The DPs emerged during and after the three design iterations of the artifact. The DPs address the MRs derived from the interviews and form guidelines for the artifact class of ML marketplaces. Our DSR project served as a case study for this research paper. (R5) In the DSR project, an ML marketplace was designed in collaboration with an implementation partner of a standard software vendor

and two German retailers. The involved retailers have an annual turnover of 120 million and 55 billion euros per year, and they serve diverging customer segments and focus on different products. These differences are beneficial for a realistic evaluation of the quality of the marketplace. It was the goal of the DSR project, to show that an ML marketplace can also provide value for participants who had different products and sales structures. In the DSR project, a common ML issue for retailers was addressed. ML forecasts often face the problem of low elasticity regarding optimization goals so low price elasticity in sales forecasts was addressed in the case study. Low price elasticity results from a small number of historical price changes, therefore an ML algorithm has no data to learn from. Price elasticity is needed for price optimization. In the DSR project, the ML model marketplace as suggested by Kumar et al. [21]. (R6) We derived DPs reflectively from this design process. We condensed the pivotal experiences from the design project in eight prescriptive DPs for ML marketplaces in the context of ES (Figure 2). (7) We followed a reusability framework for the first evaluation [47]. For the formulation of each DP, we followed Chandra et al.'s [24] template, providing the property of the ML marketplace, the possible activity of the participants, and the boundary conditions. We applied a narrow understanding of an ML within ES as a boundary condition, and focused on the exchange of ML models among companies and developers with a software vendor as marketplace owner.' The goal of the exchange of ML models is to increase the dissemination of ML models across companies and thereby achieve greater penetration of ES with ML.

4. Results

In this section, we describe ML marketplace DPs for achieving a cross-market penetration of ML solutions in ES. The mapping of the DPs for ML marketplaces to the MRs and GDs is depicted in Figure 2. To provide sufficient context for the various DPs described in this paper, we have formulated the following boundary condition: an ML marketplace is established by a software vendor and supports participating companies using ML in ES. Therefore, the DPs should support software vendors during the establishment of ML marketplaces with the goals of increasing the dissemination of ML in ES and enabling the standardization of ML. As discussed in the introduction, dominant standards can help to reduce the market penetration cycle times of innovations [10].

As the first MR, we observed from interviews (I-1, I-2, I-4, I-5, and I-7) that, the ML marketplace must



Figure 2. Mapping Diagram of Design Principles for ML Marketplaces

fit the marketplace owner's overall software strategy (MR1). This MR was gleaned from the interviews when questions regarding the two GDs *governance structure* and *accessibility & control* were asked. First, the dimension of *governance structure* entails how the authority and responsibility for each class of decisions are divided between marketplace owner and module developers [48]. For example, if the marketplace owner has an open or closed business model this has to fit the marketplace. An open marketplace is more likely to attract additional participants and thus would best match the owner's needs [49]. This requirement would ensure, that marketplace owners' existing customers could easily adapt to the ML marketplace. The second dimension is *accessibility & control*, which means that the marketplace owner decides who is allowed to access the marketplace, controls the modules offered, and specifies mechanisms for evaluating modules [38]. The access should mirror the software vendor's software strategy so that existing customers of the marketplace owner are willing to adapt to the ML marketplace. Closely bundling an ML marketplace with existing ES helps positively shape its impact for all participants and enables customers to grasp the full range of ML products and possibilities. Our results were heavily influenced by insights gained from the DSR project, and are depicted in DP1 (Figure 2).

From the other interviews (I-3, I-8, I-9, I-10) we derived the MR that the quality of the models has to be transparent, and participants and their company-specific data have to be anonymized (MR2). This MR came up while posing questions to interviewees regarding the two GDs *resource & documentation* and *trust & perceived risk*. Resource and documentation means that boundary resources such as APIs and SDKs for cultivating marketplace ecosystems or third-party development

are available and well documented [38]. ES already provides standardized interfaces such as APIs, specific programming languages (e.g., SAP ABAP), and technical documentation [50]. Nevertheless, providing a dedicated marketplace for ML models requires additional boundary resources. The dimension trust and perceived risk means that trust is enhanced and the perceived risk of marketplace participants is minimized by the marketplace [48]. We reformulated MR2 after interviewee I-10 emphasized that there are multiple data privacy restrictions. For example, there are company-specific requirements (e.g., purchase prices must not be uploaded into any cloud servers) or legal requirements (e.g., the European General Data Protection Regulation). These restrictions have been addressed with a model encryption process in DP2 (Figure 2). Anonymization of models in an ML marketplace can be achieved through the use of k-anonymization in federated learning environments by adding artificial noise to parameters of the models before uploading them to the marketplace [29]. Here, data is called k-anonymized if the identifying information of each individual is indistinguishable from at least k-1 other individuals [51]. For example, this makes it difficult to correctly link customers to their associated sensitive attributes. MR2 is further addressed by the use of three DPs regarding model quality evaluation. First, as marketplace participants will be determining which models get downloaded, it is critical that the evaluation metrics available to them be easy to interpret and expressive. Such metrics should allow participants to compare different models and provide them with concrete recommendations (DP3). There must be evaluation metrics for general model performance (e.g., general prediction error) and for specific business processes where the

ML model is to be integrated (e.g., forecasts price elasticity when integrating the forecast into the price optimization process). Common metrics can be found in related literature (e.g., forecast metrics [52, 53], optimization metrics [54, 55], clustering metrics [56, 57]). Second, uploading companies should provide their own evaluation metrics for model quality since models are explicitly trained for this context, and this evaluation represents best-case results (**DP4**) [58]. The self-assessment could serve as a viable indication for companies from the same industry. The marketplace owner as an intermediary has often access to some data of customers', who have a premium or special relationship with the marketplace owner [59]. The marketplace owner can use these data to generate evaluation metrics of the uploaded model to test its inference performance. For example, procure data provided by participants from select industries and use it as the basis for cross-market average performance evaluations of the model [49]. Third, **DP5** was formulated because the DSR project showed that it is important for participants to evaluate their decisions in software investments (Figure 2). Benefit metrics may help to evaluate how much a company benefits from the ML marketplace. The performance of an ML application should be evaluated not only on how it performs with local data and self-generated models but also on how it performs with models from the ML marketplace using the aforementioned metrics. These evaluations can only be performed in hindsight after the best models have been found. Marketplace participants should be provided with benefit metrics that outline the performance of the used ML models with and without the ML marketplace. This would support decision-making regarding investing in the ML marketplace.

The pricing dimension of GDs defines who is setting the price, who decides on participation, who is paying, and who receives a value [38]. Interview questions in regard to this dimension (I-1, I-2, I-5, and I-6), resulted in **MR3**. Software (MR1) and pricing strategy (MR3) were split into two dedicated MRs because interview participant I6 suggested to elaborate on both independently. The ML marketplace should have a pricing model in which benefiting companies must pay and contributing companies get extra privileges in order for users to feel being treated fairly (**DP6**). As defined in DP5, metrics should be defined dependent upon how much a company is benefiting from each model. The overall pricing (i.e., what a customer is paying in total to the marketplace owner) per marketplace participant should take into account how much a company is benefiting from the marketplace and how much the

marketplace is benefiting from the participant. It is possible to calculate exactly how much a company has benefitted from ML models procured through the ML marketplace by comparing performances metrics with and without ML models from the marketplace, so a fair price can be calculated. Furthermore, the ML marketplace should have a pricing model that fits the marketplace owner's pricing strategy in order for users to access the ML marketplace together with other software components of the ES (**DP7**). To make it as easy as possible for a company that is already a customer of the software vendor to become a marketplace participant for companies, the ML marketplace pricing should fit the overall marketplace owner's pricing strategy. For example, existing customers should not be required to learn new authorization or financial processes in order to become marketplace participants. Providing ES and the ML marketplace together reduces integration complexity, thus reducing integration costs [60].

Finally, interviewees I-1, I-3, I-9, and I-11 contributed to the dimension *external relationships*, which involves the *management of inter-firm dependencies* and the *architecture of participation* [38]. The interviewees suggested that the ML marketplace should be open for third-parties of the marketplace owner not providing but requiring ML models, to increase network effects as an important driver of ecosystem success (**MR4**). Google's research director Peter Norvig once claimed that "We don't have better algorithms. We just have more data.". This quote is usually linked to the article on the unreasonable impact of data by Halevy et al. [61]. It is clear that more data is not always better [62], but it is assumed that a federated learning approach, like the one we propose for an ML marketplace, significantly improves the quality of the models within it. Better ML results can result in better elasticities which are important for optimizations [63, 16]. For example, [64] show that a 1% improvement in price results in an average operating profit improvement of 11.1%. When marketplace participants are able to significantly improve their ML models, competitive advantages can be achieved and market pressure is generated on competitors that do not belong to an ML marketplace. A higher market pressure on competitors that do not belong to an ML marketplace is generated with every new ML marketplace participant. Therefore the ecosystem of the ML marketplace can be strengthened by MR4. While implementing an ML marketplace for our DSR project, we developed **DP8** which states that the ML marketplace should be provided with sophisticated APIs so that functionality is provided to

standard software or third-party platforms (e.g. Linux), as it is also suggested by literature on marketplace boundaries [58]. Models should be interpretable by common programming languages (e.g. Python or R) in order for users to either use the model with standard software or develop it further and then make it available to the marketplace again, which would enable flexible use of the models (DP8). This flexible use of the ML models can result in improved ML model lifecycles. In this approach, ML models can be downloaded, fine-tuned by the downloading company using its specific data, and uploaded again to provide future marketplace participants with a model that has been trained with even more data, and is thus slightly better.

5. Discussion

In our DSR project, we implemented an initial version of an ML marketplace that was focused on forecasting models for an implementation partner of a globally operating software vendor. The above-mentioned DPs were triggered by the MR derived from interviews and the DSR project. In what follows, we provide an initial evaluation [47] and discuss the context of the DPs introduced, briefly mention other applications possible with different marketplace owners, and consider the incorporation of other marketplaces into ES. Although we plan to conduct an additional round of expert interviews to evaluate our eight DPs, we provide an initial evaluation following the framework for light re-usability evaluation of DPs [47] as proposed by Möller et al. [44]. As DPs for a class of systems form a unit of prescriptive knowledge, they are evaluated as a comprehensive set [47]. The evaluation criteria are accessibility, importance, novelty and insightfulness, actability and guidance, and effectiveness [47]. We ensured that the DPs are accessible for potential platform owners by shaping them in accordance with the advice given by our interviewees. The importance criterion has been satisfied for our DPs by providing a solution for the underrepresentation issue of ML in ES through the proposal of standardized guidelines for the successful establishment of ML [10]. By applying GDs to marketplaces in ES, we provide a novel perspective for an implementation of an ML marketplace important that can attract myriad participants [22]. We provide direct guidance for software providers implementing an ML marketplace. When implementing an ML marketplace, the software provider already has customers for his existing ES and can use a piggy-backing strategy and secure a large number of customers [65, 66]. Our DPs are expected to be effective for software vendors

who are implementing an ML marketplace because the customer side of the marketplace already exists. Rather than having to secure customers, it could focus on the other market side and find external participants who can provide ML models [67]. Such cases would result in reduced time and effort while increasing the likelihood of obtaining both a critical mass of customers and a de facto standard for federated learning. [49, 10, 68].

As mentioned earlier, we assume the context of standardized ES provided by a software vendor as a boundary for our DPs with the goal of supporting an ML solution implementation by a mass market. This assumption is a reasonable one since ES is the de facto standard for enterprise software on the market and ML belongs to the class of enterprise application software used in companies [12]. Because the ML marketplace provides pre-trained ML models, they must be integrated into the corresponding ES of the marketplace participant. Such dedicated system architecture must be elaborated upon. When big data or data stream must be handled a corresponding architecture must be developed, as indicated by [69]. There are some DPs that should be able to transfer to other contextual boundaries such as DP2, DP3, DP4, DP5, DP6, and DP8. DP1 and DP7 take into account the context of ES and the ES vendor strategy. We assume that an ML marketplace can be implemented and used by customers without these two DPs. That said, we also assume that market penetration can be increased to incorporate the ML marketplace into a given ES. The above design principles are valid across all industries. While the meta requirements were derived from the interviews, the design principles were derived from a DSR project. This DSR project took place in the retail industry. However, the design principles were formulated in such a way that they are valid across all industries. Thus, the design principles for the ML marketplace follow a similar approach to existing ES marketplaces, such as SAP Store (sapstore.com) or Microsoft App Source (appsource.microsoft.com). These are also cross-industry solutions that can be filtered by industry. Likewise, it is conceivable in the ML marketplace that ML models can be filtered by industry.

The ML marketplace can be implemented by multiple stakeholders. For example, standard software vendors and consulting companies could implement an ML marketplace. Furthermore, the ML marketplace could be developed by a customer of an ES vendor. Because of the various possibilities outlined above, each marketplace owner should have a dedicated pricing model (DP7). As the standard software vendor is already selling software, he should continue to use his

pricing model in his ML marketplace. Because ML models would be continuously improved, implementing a continuous license pricing model similar to that of cloud software makes the most sense [70].

While the DPs are derived from a forecasting marketplace and formulated with the boundary condition of ML marketplaces in ES provided by a software vendor, they may also be relevant in other marketplace contexts. For example, there are other marketplaces in the context of company-wide standard software. External developers can develop additional programs, which augment the standard software. They are able to upload these developments to a software-specific marketplace afterward and sell it to other ES vendor's customers [71]. These kinds of marketplaces exist for most large ES (e.g., ORACLE cloud marketplace, salesforce appexchange, SAP Store). These modules are provided by implementation partners or consulting companies that have investigated a missing functionality in the standard software multiple times, and then they developed software with that function for the marketplace. With little adaptation, DPs of ML marketplaces may also be true for this kind of marketplaces with an appropriate adoption. For example, appropriate evaluation metrics or APSs must be found. The MRs were derived from interviews and the DPs were derived from a DSR project to implement and evaluate an ML marketplace. Although the DSR project to implement an ML marketplace was carried out with industry-specific market participants in retail, the DPs mentioned are valid across all industries. In particular, such marketplaces are often created across industries in the context of ES and solutions can be filtered for specific industries.

This research also has its limitations. The DPs identified may be limited to the context of ES delivered by standard software vendors. It may be interesting to investigate the ML marketplace in different contexts, such as standard software vendors for office applications like e-mail programs, which often include many ML applications, such as spam detection, spell checking, etc. We have not investigated user interface DPs, as we focused on GDs of ML marketplaces, and there already exist many interface design guidelines.

6. Conclusion and Future Research

The result of our research paper is twofold. First, we extracted four MRs for ML marketplaces in ES from interviews with domain experts for ES which have substantial experience in the implementation of ML algorithms. The transparency of the model quality while maintaining anonymity (MR2) and the

openness of the ML marketplace proposed (MR4) are crucial requirements for the integration of ML in future ES. Second, we propose eight DPs for such an ML marketplace derived from a DSR project on a marketplace focusing on ML models for forecasts. Through an ML marketplace, ML models can help to shape the future of ES. This will simplify the integration of ML in ES and foster cross-market penetration. By exchanging ML models among companies in a federated learning approach, companies can overcome a lack of missing historical data necessary to train models in ML. Thus, additional functionality can be integrated with additional ML models downloaded from an ML marketplace to an ES. These ML models can be individualized, by fine-tuning the downloaded ML models to a company's specific scenario [72, 73, 74]. This paper contributes to the existing literature by investigating the ES context for ML marketplaces (federated learning). Furthermore, there are other ML techniques that can be used to address the problem of missing data, such as the concept of generative adversarial networks, which could be investigated by future research [75]. Because our DPs focus on GDs for ML marketplaces, future research may investigate ML marketplaces providing an environment for developing ML solutions and training ML models. For collaboration across marketplaces and the co-creation of ML models, suitable interfaces and data formats must be defined by the marketplace owner that enable exchange at the technological level. For this purpose, cross-company ecosystem modeling on multiple levels can be useful. Another important avenue for future research would be to evaluate our DPs within existing ES and general marketplaces.

References

- [1] M. Ali and L. Miller, "Erp system implementation in large enterprises-a systematic literature review," *Journal of Enterprise Information Management*, vol. 30, no. 4, 2017.
- [2] A. Benlian and T. Hess, "Comparing the relative importance of evaluation criteria in proprietary and open-source enterprise application software selection - a conjoint study of ERP and Office systems," *Information Systems Journal*, vol. 21, no. 6, pp. 503–525, 2011.
- [3] Y. Van Everdingen, J. Van Hillegersberg, and E. Waarts, "Enterprise resource planning: Erp adoption by european midsize companies," *Communications of the ACM*, vol. 43, no. 4, pp. 27–31, 2000.
- [4] PWC, "Global 100 software leaders by revenue," 2016.
- [5] B. Bender, C. Bertheau, and N. Gronau, "Future erp systems: A research agenda," in *IEIS2021*, 2021.
- [6] D. Autor and D. Dorn, "the Growth of Low Skill Service Jobs and the Polarization of the," *National Bureau of Economic Research Working paper series*, vol. 15150, pp. 1689–1699, 2013.

- [7] M. Goos, A. Manning, and A. Salomons, "Job polarization in europe," *American economic review*, vol. 99, no. 2, pp. 58–63, 2009.
- [8] S. M. Veres, L. Molnar, N. K. Lincoln, and C. P. Morice, "Autonomous vehicle control systems - A review of decision making," *Proceedings of the Institution of Mechanical Engineers. Part I: Journal of Systems and Control Engineering*, vol. 225, no. 3, pp. 155–195, 2011.
- [9] T. Plötz and G. A. Fink, "Markov models for offline handwriting recognition: A survey," *International Journal on Document Analysis and Recognition*, vol. 12, no. 4, pp. 269–298, 2009.
- [10] T. S. Robertson, "How to reduce market penetration cycle times," *MIT Sloan Management Review*, vol. 35, no. 1, p. 87, 1993.
- [11] G. Timbrell and G. Gable, "The sap ecosystem: A knowledge perspective," in *Enterprise Resource Planning: Solutions and Management*, pp. 209–220, IGI Global, 2002.
- [12] E. Viardot, I. Mccarthy, and J. Chen, "Editorial: Standardization in a Digital and Global World: State-of-the-Art and Future Perspectives," *IEEE Transactions on Engineering Management*, vol. 68, no. 1, pp. 11–17, 2021.
- [13] IDC, "Idc forecasts improved growth for global ai market in 2021," 2021.
- [14] V. Uren, "Critical success factors for artificial intelligence projects," in *EurOMA Conference 2020*, 2020.
- [15] D. Delen, "A comparative analysis of machine learning techniques for student retention management," *Decision Support Systems*, vol. 49, no. 4, pp. 498–506, 2010.
- [16] M. Hütsch and T. Wulfert, "Implementation of Placement Optimization in Electronic Commerce: Towards a Data Generation Method to Increase Forecast Elasticity," in *PACIS2021*, (Dubai), pp. 1–12, 2021.
- [17] N. Zolas, Z. Kroff, E. Brynjolfsson, K. McElheran, D. N. Beede, C. Buffington, N. Goldschlag, L. Foster, and E. Dinlersoz, "Advanced technologies adoption and use by us firms: Evidence from the annual business survey," tech. rep., National Bureau of Economic Research, 2021.
- [18] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2016.
- [19] H. B. McMahan, E. Moore, D. Ramage, and B. A. y Arcas, "Federated learning of deep networks using model averaging," *arXiv preprint arXiv:1602.05629*, 2016.
- [20] A. Hard, K. Rao, R. Mathews, S. Ramaswamy, F. Beaufays, S. Augenstein, H. Eichner, C. Kiddon, and D. Ramage, "Federated learning for mobile keyboard prediction," *arXiv preprint arXiv:1811.03604*, 2018.
- [21] A. Kumar, B. Finley, T. Braud, S. Tarkoma, and P. Hui, "Marketplace for ai models," *arXiv preprint arXiv:2003.01593*, 2020.
- [22] A. Tiwana, *Platform Ecosystems*. Waltham: Morgan Kaufman, 2014.
- [23] C. Alves, J. Oliveira, and S. Jansen, "Software Ecosystems Governance-A Systematic Literature Review and Research Agenda," in *ICEIS2017*, pp. 26–29, 2017.
- [24] L. Chandra, S. Seidel, and S. Gregor, "Prescriptive Knowledge in IS Research: Conceptualizing Design Principles in Terms of Materiality , Action , and Boundary Conditions," in *HICSS2015*, pp. 4039–4048, 2015.
- [25] T. M. Mitchell *et al.*, "Machine learning," 1997.
- [26] D. L. Olson and S. Kesharwani, *Enterprise information systems: Contemporary trends and issues*. Singapore: World Scientific, 2009.
- [27] S. Sawyer, "Packaged software: implications of the differences from custom approaches to software development," *European journal of information systems*, vol. 9, no. 1, pp. 47–58, 2000.
- [28] J. Vom Brocke, W. Maaß, P. Buxmann, A. Maedche, J. M. Leimeister, and G. Pecht, "Future work and enterprise systems," *Business & Information Systems Engineering*, vol. 60, no. 4, pp. 357–366, 2018.
- [29] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. Quek, and H. V. Poor, "Federated learning with differential privacy: Algorithms and performance analysis," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3454–3469, 2020.
- [30] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *ICANN2021*, pp. 270–279, 2018.
- [31] T. Karb, N. Kühl, R. Hirt, and V. Glivici-Cotruta, "A network-based transfer learning approach to improve sales forecasting of new products," *arXiv preprint arXiv:2005.06978*, 2020.
- [32] S. Wang and N. P. Archer, "Electronic marketplace definition and classification: literature review and clarifications," *Enterprise Information Systems*, vol. 1, no. 1, pp. 89–112, 2007.
- [33] S.-y. Choi, A. B. Whinston, and D. Stahl, *The Economics of Electronic Commerce*. Indianapolis: Macmillan Computer Publishing, 1997.
- [34] M. Armstrong, "Competition in two-sided markets," *RAND Journal of Economics*, vol. 37, no. 3, pp. 668–691, 2006.
- [35] J. Rochet and J. Tirole, "Platform competition in two-sided markets," *Journal of the European Economic Association*, vol. 1, no. 4, pp. 990–1029, 2003.
- [36] K. S. Staykova and J. Damsgaard, "A Typology of Multi-sided Platforms: The Core and the Periphery," in *ECIS2012*, 2015.
- [37] T. Eisenmann, G. Parker, and M. Van Alstyne, "Opening Platforms: How, When and Why?," in *Platforms, Markets and Innovation* (A. Gawer, ed.), pp. 131–162, Cheltenham: Edward Elgar, 2009.
- [38] A. Hein, M. Schreieck, M. Wiesche, and H. Kremer, "Multiple-Case Analysis on Governance Mechanisms of Multi-Sided Platforms," in *MKW2016*, (Ilmenau), pp. 1613–1624, 2016.
- [39] "Technology ecosystem governance," *Organization Science*, vol. 25, no. 4, pp. 1195–1215, 2014.
- [40] H. Simon, *The Sciences of the Artificial*. Cambridge, Massachusetts: MIT Press, 3 ed., 1996.
- [41] M. Sein, O. Henfridsson, S. Purao, M. Rossi, and R. Lindgren, "Action design research," *MIS quarterly*, vol. 35, no. 1, pp. 37–56, 2011.

- [42] N. Cross, "Designerly Ways of Knowing: Design Discipline Versus Design Science," *Design Issues*, vol. 17, no. 3, pp. 49–55, 2001.
- [43] R. Lukyanenko and J. Parsons, "Design theory indeterminacy: What is it, how can it be reduced, and why did the polar bear drown?," *Journal of the Association for Information Systems*, vol. 21, no. 5, pp. 1343–1369, 2020.
- [44] F. Möller, T. M. Guggenberger, and B. Otto, "Towards a Method for Design Principle Development in Information Systems," *Lecture Notes in Computer Science*, pp. 208–220, 2020.
- [45] M. D. Myers and M. Newman, "The qualitative interview in IS research: Examining the craft," *Information and Organization*, vol. 17, no. 1, pp. 2–26, 2007.
- [46] P. Mayring, *Qualitative Content Analysis: Theoretical Foundation, Basic Procedures and Software Solution*. Klagenfurt: Beltz, 2014.
- [47] J. Iivari, M. Rotvit Perlt Hansen, and A. Haj-Bolouri, "A proposal for minimum reusability evaluation of design principles," *European Journal of Information Systems*, no. May, pp. 1–18, 2021.
- [48] S. Nambisan, "Information technology and product/service innovation: A brief assessment and some suggestions for future research," *Journal of the association for information systems*, vol. 14, no. 4, p. 1, 2013.
- [49] D. S. Evans and R. Schmalensee, *Catalyst code: the strategies behind the world's most dynamic companies*. Harvard Business School Press, 2007.
- [50] B. Eaton, S. Elaluf-Calderwood, C. Sorensen, and Y. Yoo, "Distributed tuning of boundary resources: the case of Apple's iOS service system," *MIS Quarterly*, vol. 39, no. 1, pp. 217–243, 2015.
- [51] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.
- [52] A. Botchkarev, "Performance metrics (error measures) in machine learning regression, forecasting and prognostics: Properties and typology," *arXiv preprint arXiv:1809.03006*, 2018.
- [53] J. G. D. Gooijer and R. J. Hyndman, "25 years of time series forecasting," *International Journal of Forecasting*, vol. 22, no. 3, pp. 443–473, 2006.
- [54] S. Mirjalili and A. Lewis, "Novel performance metrics for robust multi-objective optimization algorithms," *Swarm and Evolutionary Computation*, vol. 21, pp. 1–23, 2015.
- [55] S. Jiang, Y.-S. Ong, J. Zhang, and L. Feng, "Consistencies and contradictions of performance metrics in multiobjective optimization," *IEEE transactions on cybernetics*, vol. 44, no. 12, pp. 2391–2404, 2014.
- [56] P. Grabusts, "The choice of metrics for clustering algorithms," in *ISCDE2011*, vol. 2, pp. 70–76, 2011.
- [57] A. Biswas and B. Biswas, "Defining quality metrics for graph clustering evaluation," *Expert Systems with Applications*, vol. 71, pp. 1–17, 2017.
- [58] A. Ghazawneh and O. Henfridsson, "Balancing platform control and external contribution in third-party development: the boundary resources model," *Information systems journal*, vol. 23, no. 2, pp. 173–192, 2013.
- [59] A. Hagiwara and J. Wright, "Multi-sided platforms," *International Journal of Industrial Organization*, vol. 43, pp. 162–174, 2015.
- [60] G. D. Eppen, W. A. Hanson, and R. K. Martin, *Bundling-new products, new markets, low risk*. Purdue University, Krannert Graduate School of Management, 1991.
- [61] A. Halevy, P. Norvig, and F. Pereira, "The unreasonable effectiveness of data," *IEEE Intelligent Systems*, vol. 24, no. 2, pp. 8–12, 2009.
- [62] I. Pilászy and D. Tikk, "Recommending new movies: even a few ratings are more valuable than metadata," in *Proceedings of the third ACM conference on Recommender systems*, pp. 93–100, 2009.
- [63] N. Borin and P. Farris, "A sensitivity analysis of retailer shelf management models," *Journal of Retailing*, vol. 71, no. 2, pp. 153–171, 1995.
- [64] M. V. Marn and R. L. Rosiello, "Managing price, gaining profit," *McKinsey Quarterly*, pp. 18–18, 1992.
- [65] B. Bender, N. Habib, and N. Gronau, "Digitale Plattformen: Strategien für KMU," *Wirtschaftsinformatik & Management*, vol. 13, no. 1, pp. 68–76, 2021.
- [66] L. C. Reillier and B. Reillier, *Platform Strategy*. London: Routledge, 2017.
- [67] M. W. Van Alstyne, G. G. Parker, and S. Paul Choudary, "Pipelines, platforms, and the new rules of strategy," *Harvard Business Review*, vol. 2016, pp. 54–62, 2016.
- [68] M. Hütsch, "Comparing architectures of neural networks for an integration in enterprise systems: A retail case study," *Procedia Computer Science*, vol. 181, pp. 619–627, 2021.
- [69] A. L'heureux, K. Grolinger, H. F. Elyamany, and M. A. Capretz, "Machine learning with big data: Challenges and approaches," *Ieee Access*, vol. 5, pp. 7776–7797, 2017.
- [70] M. Al-Roomi, S. Al-Ebrahim, S. Buqrais, and I. Ahmad, "Cloud computing pricing models: a survey," *International Journal of Grid and Distributed Computing*, vol. 6, no. 5, pp. 93–106, 2013.
- [71] J. Owyang, "The collaborative economy," 2013.
- [72] J. Dodge, G. Ilharco, R. Schwartz, A. Farhadi, H. Hajishirzi, and N. Smith, "Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping," *arXiv preprint arXiv:2002.06305*, 2020.
- [73] J.-S. Lee and J. Hsiang, "Patentbert: Patent classification with fine-tuning a pre-trained bert model," *arXiv preprint arXiv:1906.02124*, 2019.
- [74] H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and T. Zhao, "Smart: Robust and efficient fine-tuning for pre-trained natural language models through principled regularized optimization," *arXiv preprint arXiv:1911.03437*, 2019.
- [75] M. Leznik, P. Michalsky, P. Willis, B. Schanzel, P.-O. Östberg, and J. Domaschka, "Multivariate time series synthesis using generative adversarial networks," in *Proceedings of the ACM/SPEC International Conference on Performance Engineering*, pp. 43–50, 2021.