Towards a Governance of Low-Code Development Platforms Using the Example of Microsoft PowerPlatform in a Multinational Company

Marvin Heuer
University of Hamburg, Germany
marvin.heuer@uni-hamburg.de

Christian Kurtz
University of Hamburg, Germany
christian.kurtz@uni-hamburg.de

Tilo Böhmann
University of Hamburg, Germany
tilo.boehmann@uni-hamburg.de

Abstract

The advantages of low-code platforms include the ability to better manage corporate processes. These processes can quickly become customized and business critical. Consequently, governance of low-code development platforms gains increasing importance in the IT sector. This paper contributes to the design of a governance for low-code platforms using the example of Microsoft PowerPlatform through the action-design-research-paradigm within a consumer goods corporation. The paper shows both the relationship between IT governance and low-code platforms and what challenges this poses, as well as the importance of governance of low-code platforms in relation to the field of end-user computing. This work aims at developing governance for low-code platforms, and evaluates it using several methods. Based on the results of the naturalistic evaluation, design principles for the development of a governance for low-code platforms are derived. The principles summarize suggestions for designing such governance, providing evidence-based design knowledge for developing governance of low-code platforms.

1. Introduction

Currently, low-code development platforms (LCDPs) are becoming increasingly important in the IT sector. LCDPs are delivered via a platform-as-a-service model and enable users to create fully deployable applications through interaction "via dynamic graphical user interfaces, visual diagrams, and declarative languages" [1, 3, 59]. As part of this, software modules enable data integration from external and internal sources through connectors or the integration of services from external third-party providers, such as social networks. Advantages of LCDPs include the ability to better manage corporate processes in the age of rapidly advancing digitization. In the context of this, these processes can quickly become customized and business-critical. As they accelerate the implementation of requirements, companies can respond more agilely and save IT resources. Various benefits and also risks arise, which are managed by an appropriate governance model. As part of Microsoft 365 the Microsoft PowerPlatform is offered as a LCDP. With the expansion of Microsoft 365, the Microsoft PowerPlatform is one of the most widely used platforms and 97% of Fortune 500 companies use it [1]. Due to the ease of use of LCDPs and the resulting strong appeal to non-technical users ("end-users"), it is important to ensure that the applications developed do not compromise IT security. Likewise, there is a risk of developing "shadow IT". Although these shadow applications initially increase productivity, the applications are unsupported by the IT department and can also lead to IT security problems and a lack of transparency. The international company under consideration here is already currently using Microsoft 365 and is now facing the challenge of governance in the use of the PowerPlatform, which is becoming increasingly important in times of integration of data and systems. As main research question, we formulate the following question:

RQ: How can governance of LCDPs be designed through the ADR paradigm in a multinational organization using Microsoft PowerPlatform as an example?

To address this research question, we conduct a systematic literature review following the process proposed by vom Brocke et al. (2009) [2]. Our research provides a first overview of the analyzed literature with a systemized set of factors which clarify the special features of LCDPs. The results indicate that beyond the aforementioned factors and especially in contrast to the management and introduction of other traditional IS and their governance, a new holistic form of governance for introducing LCDPs in organizations is needed. Due to the nature of LCDPs as a new class of information system, characterized by increasing levels of end-user control and cloud service provision occurring as (not novel) actors in work routines, services, and processes, this governance approach stands out from traditional IS governance approaches. This leads to a research agenda regarding the design of a governance for LCDPs, including the special circumstances of end-user and

URI: https://hdl.handle.net/10125/80171
978-0-9981331-5-7

cloud involvement. This paper shows both the relationship between IT governance and LCDPs and what particular challenges this poses, as well as the importance of governance of LCDPs in relation to the field of end-user computing. To develop a concept, observations, usage statistics and surveys were conducted in the company for six month in an iterative approach to generate insights for the development and implementation of a governance. This paper aims at developing governance for LCDPs using Microsoft PowerPlatform as an example (including environments, connectors, and monitoring) in a multinational company, and evaluates it using various methods (scenarios, simulation, focus groups, and structured surveys). Based on the results of the naturalistic evaluation, design principles for the development of a governance for LCDPs are derived. The principles summarize suggestions for designing such governance, providing evidence-based design knowledge for developing governance of LCDPs. Section 2 starts with an introduction on the most important key theoretical foundations. The further procedure and approach to answer this research question is presented in detail in section 3. The results, discussion and conclusion follow in sections 4 and 5.

2. Theoretical Background

2.1. End-User Computing

Supporting users in an organizational setting has been an important goal of information systems [7-9]. According to [10], "an information system is a system in which human participants and/or machines perform work (processes and activities) using information, technology, and other resources to produce informational products and/or services for internal or external customers" [7, 10]. Within the increasing spread of personal computers and on-going development of client-server architectures, the inclusion of all end-users was embraced, including those from outside the organization's IT department [7, 11]. The term end-user computing (EUC) evolved as a result and describes "an information processing activity in which the individual has direct personal control over all phases of the activity" [12] or "a process in which the user develops applications in an environment that provides access to computing, data, and support resources" [7-8, 13-16]. Several research was performed in other associated areas (including but not limited to e.g., end-user development or end-programming) [7, 17]. Similarly, to these end-user approaches is that the power over actions and resources is given to end-users [7, 18] including the ability for personalizing and customizing interaction with the system [7, 19]. New approaches, like end-user development, offer various possibilities [7, 19], including "simply adjusting parameter values, to recording repetitive interactions in macros, to creating entirely new content and system behaviors using scripts, models such as spreadsheets, or even full-fledged programming languages" [19]. By many examples of novel innovations (e.g., smartphones, Big Data, or LCDPs), users get empowered [7]. This leads to various benefits, such as increase in organizational productivity, systems that are closer to the user, and more satisfaction in implementation [20-21]. Similarly, end users can develop applications faster than before [22].

Different studies [22-27] highlight some problem areas and challenges of EUC, including:

Table 1. Problem areas and challenges of EUC

Lack of an enterprise strategy
Lack of an organizational design between EUC, management, and business departments
Low technical quality of applications
Inadequate documentation
Data security and data integrity
Private computing resources

Some of these challenges were attempted to be solved at the time, but others have not been solved to date. The lack of an enterprise strategy regarding EUC leads to a lack of global information architecture and possible misallocation of resources [22, 28]. Problems of this nature are often addressed in today's end-user development systems by the programs themselves, such as in SaaS solutions (e.g., [29]). Other problems again pose challenges to enterprise IT today, which must be addressed.

2.2. Low-Code Development Platform

The term LCDP refers to development environments within software and the associated development steps are enabled by means of simplified visual methods. The goal is to significantly reduce the development, training, and deployment time for business applications [3], up to ten times more productively than before [6]. Software development gets enabled for end-users that have no coding experience. In LCDPs, humans continue to be the programmers. In contrast to conventional development environments, however, individual programming work hardly takes place here and predesigned software modules are offered instead. Connections to possible third-party providers and their external data must be enabled, e.g., through REST APIs. As in conventional programmed applications, a logic must be defined according to which the application should work, and an interface must be conceptualized. These platforms differ in execution mainly in the way they are programmed.

Overall, it is assumed that LCDPs will become widespread, but without completely replacing back-office systems [4]. One of the most intrusive features about LCDPs is their reusability within the platform. Templates, connectors, plug-ins, and other components are provided within the platforms. These are fully ready to use and can be reused repeatedly. As a result, productivity can be increased during use. In some cases, it is part of the business model of individual companies to sell such components to organizations for a small fee [3].

Many LCDPs combine their functionality with flexible deployment of applications within a cloud or even on-site at the customer's premises. Cloud-based deployment ensures instant availability and supports even complex applications, as extensive computations can be offloaded to cloud servers [5].

Equally beneficial is the LCDP support of developed applications even after the actual development. In contrast to classic application development, which is only involved in the areas of design and implementation, the platforms also act for deployment and maintenance. In addition, projects can be paused or continued at any time. In combination with the support of the entire application lifecycle, this results in competitive advantages, such as a faster time-to-market [4].

2.3. Microsoft PowerPlatform

The Microsoft PowerPlatform is a product family of the Microsoft Corporation consisting of the four applications Power Automate, Power Apps, Power BI, and Power VirtualAgents. In this work, the first two are examined. The applications can be combined with each other, allowing companies to increase the productivity and efficiency of their processes through targeted use. Microsoft PowerPlatform is the leading LCDP for companies [3]. All applications can be used in a web browser or on a mobile phone. As a result, there are hardly any obstacles from the point of view of hardware requirements. In addition, the entire PowerPlatform is based on the low-code paradigm. All applications can be used and operated without any great programming knowledge. This ease of use of the LCDPs results in a strong attraction for non-technical users. All PowerPlatform applications are characterized by their user-friendliness and the lack of lengthy training. Therefore, the platform is well suited for "citizen developers" [3]. There is also often a large selection of interfaces to be found, over 300 connectors in general alone [3]. In addition, the PowerPlatform is constantly being expanded.

The Power Automate application supports the user in automating repetitive tasks. Either personal processes or business processes can be partially automated. Microsoft supports the user by providing templates that can be used to develop simple flows relatively quickly and by providing so-called connectors. These connectors allow third party systems to be integrated into the flow. On the other hand, it is also possible to use the flows for applications outside the Microsoft environment. The connectors make it possible to automate processes that use other Microsoft products.

Power Apps is a platform that allows users to develop small applications with relatively little effort by drag and drop. This enables the orchestration of applications according to a kind of construction kit. Applications are developed in a simplified manner by e.g., visual application designs, and without classic, manual programming. Here, too, Microsoft promotes the use by providing connectors. This makes it possible to use data from various sources such as SharePoint, Salesforce, or other third-party systems.

Connectors are wrappers around an API that allow Power Automate and Power Apps to easily communicate with this service [1]. They are provided as a cloud solution, which is controlled only by Microsoft.

Environments are containers used by administrators "to manage apps, flows, connections and other assets, along with permissions that allow the organization's users to use the resources" [1].

3. Research Approach

In order to develop a governance for LCDP, mainly two techniques were performed. Those two, the systematic literature review and the action design research paradigm, are presented in the following. Additionally, various methods of the ADR cycle were applied and are described in section 4.

3.1. Systematic Literature Review

To identify the current state on LCDPs, we conduct a systematic literature review (SLR) [2].

First, the definition of scope of the SLR grounds in Cooper’s taxonomy [30]. As the procedure described by [61], the focus lies on the results and introduction of governance of LCDP (*Focus*). The research aim is the identification, integration, and summarization of key questions about what challenges the governance of LCDP can impose as opposed to traditional IS (*Goal*). This step is undertaken from a neutral perspective in order to ensure that no existing work inflicts the results, and we achieve representative research findings (*Perspective* and *Coverage*). We organize and cluster the found results (*Organization*). In addition, this literature review is intended to address IS researchers as

well as practitioners concerned with the introduction of governance in a large-scale organization (*Audience*).

In a next step, based on the definitions and terms introduced in the previous Section 2, we analyzed the challenges of governance on LCDP and their organizational impact. Following from this groundwork, we conduct an initial search, considering literature on governance from different perspectives to collect commonly used synonyms for the terms *end-user development* and especially *low-code platforms*.

Moreover, for deriving possible keywords on designing and application of governance of LCDP, we create a list consisting of synonyms for our search words. As part of the "literature search" phase, we used the found keywords to create and improve our search queries. Subsequently, the various found search words were iteratively optimized and extended.

3.2. Action Design Research

The developed research question requires a process that can integrate "various perspectives on the research problem as well as supporting in creating an artifact that covers prescriptive knowledge" on governance of LCDP and end-user development as shown in [31]. We did use action design research (ADR). ADR is well-established and helps creating IT-related artifacts in a rapidly changing, organizational context [31, 32]. In ADR, action and design research is combined to generate prescriptive design knowledge [32]. ADR combines artefact creation, organizational intervention, evaluation and specifically considers artifact design "with practitioners (i.e., individuals with first-hand experience) and end-users" [31, 33]. ADR results in various artefacts that incorporate user influence and their usage in organizational context while refining theory and researcher intent [31-33], and therefore provides support for practitioners in a variety of contexts [31, 33]. In general, methods are based on a fixed number of phases represented in models that describe the links between constructs [31, 34]. After the first stage, ADR aims to shape an IT artifact [31, 32]. It integrates the design, intervention, and evaluation, as influenced by practitioner or end-user involvement [31-32]. Reflection and learning are performed continuously to enable the consolidation of the feedback and the following learning [31, 35]. Since governance for end-user development environments, especially LCDP, is a complicated socio-technical issue, we considered ADR to be the best approach to answer the research question [31] and followed the guidelines of [32] in developing a governance of LCDPs.

4. Results

The following section states the results of the ADR cycles, including the results of the SLR and the expert interviews.

Breakdown of ADR method (based on [35])	
Problem formulation (Stage 1)	Data gathering and documentation by interviews and stakeholder meetings
Building, Intervention, and Evaluation (Stage 2 or BIE stage)	Iterative development, testing & evaluation Formulation of preliminary design principles Focus groups with the ADR team
Reflection and learning (and ongoing BIE) (Stage 2&3)	Iterative development & evaluation Ongoing formalization of design principles
Formalization of learning (and ongoing BIE) (Stage 2&4)	Ongoing iterative development & evaluation Generalization of design principles
Outcomes and reiteration of stages 2-4	Transition of the artifact to the production phase Conclusion of iterative development & testing

4.1. Problem Formulation

This first stage gets triggered "when a problem is perceived in practice or anticipated by researchers" [32, 35]. For the identification of the concrete problem, several steps were done. First, a systematic literature review as described in section 3.1 was performed [2, 37]. Second, expert interviews with six key stakeholders and three other important practitioners were conducted. Third, the current state of the already implemented PowerPlatform in the company was examined regarding the existing flows, apps, environments, and connectors. Fourth, an organization-wide survey was distributed regarding the end user's opinion on the current state of the LCDPs in the company, especially the Microsoft PowerPlatform. The results showed that at the start 24,018 flows and 944 apps already existed on the PowerPlatform. The PowerPlatform provided 419 connectors that were accessible for every employee, non-limited, and provided from the cloud by Microsoft, although only 258 persons use the PowerPlatform (1.25% of employees). In the first stage of the PowerPlatform no environment or governance strategy has been established. Therefore 695 flows used critical connectors. These are connectors, that can link internal data to external services. In addition, connectors can be

provided and activated by Microsoft directly, and be manually deactivated after the first provision by the organization. All developed flows and apps existed in one big environment, where no control about the connectors or the ownership was executed. In addition, no monitoring for usage purposes was implemented regarding the PowerPlatform. Based on these disturbing results, the goal was set to develop and implement a governance for the LCDP under observation.

Based on the conducted interviews and SLR, Table 2 shows the challenges LCDPs pose from a stakeholder, end-user, and literature perspective.

Table 2. Challenges of low-code platforms

Challenges of Low-Code Platforms
Third-party vendors can provide extensions
Little coding - easy access for non-technical users
Fast access and handling of (sensitive) data
Access issues
Fast implementation of requirements - Companies become more agile and IT resources are saved
Danger of "shadow IT"

Interestingly, similar conclusions could be drawn from the literature on EUC. The phenomenon has led to various benefits, such as an increase in organizational productivity, systems that are closer to the user, and more satisfaction in implementation [20, 21]. Likewise, end-users can develop applications faster than before [22]. Compared to the specifics of LCDPs, this showed a surprising similarity [38, 39]. Differences here seem to lie in the following points. In contrast to EUC, LCDPs provide extensions through third-party vendors [29, 40]. These extensions (including connectors) are provided on a service basis by other actors and get pushed into the organization by Microsoft [1]. Data security and data integrity, on the other hand, already posed challenges in EUC [27]. An important difference between EUC and LCDP is that LCDPs are available as PaaS [41]. This solves some problems that EUC faced (lack of global information architecture and possible misallocation of resources) [22].

4.2. BIE Stage, Reflection and Learning

Based on the results of the problem formulation stage, the conducted interviews, and the identified practice-related challenges, for the second stage the IT governance model by Weill (2004) was taken as foundation [42]. Apart from the challenges of LCDPs, certain governance aspects were emphasized as results of the conducted interviews. As a first result, a set of IT decisions was defined that a governance of LCDP needs to make, as shown in Table 3.

Table 3. First decision areas of governance

Decision area	Description
Principles	Principles of using connectors and a data strategy within the low-code platform.
Architecture	Technical implementation of the principles
Critical applications	Business critical applications and their needs
Future issues	Decisions of prioritization of novel changes, especially functions, including specifications for decision support methods

One issue that LCDPs have to address is which modules or connectors are allowed, and how these are allowed to interact. This leads to the question of the principles behind this, and additionally about the data strategy. These questions can be summarized in a decision area, the principles. This first decision area already differentiates the comparatively new LCDPs from the sister counterpart of EUC.

While the principles are captured in this first decision area, they must also be implemented technically. Based on the decision areas according to [42], a second decision area can be derived here, which is called architecture. This refers to the guidelines that implement business requirements.

Business-critical applications can be defined, or even automated on a LCDP. These applications are critical and must not be negatively impacted by governance. Since these applications have a concrete impact on the success of the company, their needs must be considered to a special degree. Therefore, there is a need for a separate decision area.

LCDPs are usually located in a cloud. The functionality is fully available to an organization, but the provision of operation and in particular of novel functions is left to the cloud provider. Moreover, the provision of extensions and connectors gets decided by the owner and the organizations needs to react on it. This results in a fundamentally necessary decision area, as these novel changes can have a strong impact on the usability of the platform. Further, this also includes specifications for decision support methods. Here, too, a special feature becomes apparent in the comparison of LCDP with EUC: the connection to a cloud.

By this iteration and after the according interviews, it was clear that the different business units as stakeholders need to get involved in order to gather all the requirements. This led to a first tentative design principle of early involvement of business units. To further support these theses, structured surveys were conducted with a small group of practitioners [43]. The

respondents were divided into two classes, "business stakeholders" and "individuals". These are designated B or I respectively and numbered consecutively. The individuals were selected based on their activity on Microsoft PowerPlatform. The interviewees were all asked for their views on the LCDP and governance. From the views of the sources, these could be clustered into questions that were similarly addressed by different sources. The next step was to cluster again. This time, the questions from the survey results were clustered into more general areas. This also resulted in five areas thematically. The procedure is shown in Table 4.

Table 4. Methodology of approach

Source B	Source I	Question	Decision area
All	I1,3	Which building blocks are approved?	Principles
All	I1	What principles apply?	
B4	I1	How are principles implemented technically?	Architecture
B1,4-5		How is compliance with principles permanently ensured?	
B1	I3	Which apps are the responsibility of IT? To what extent?	Application Development
B1-3		What happens with business-critical apps?	Critical applications
All		What happens in the event of changes to the PowerPlatform?	Future issues

After this evaluation round, two things became clear. First, that the design principle of early involvement of business units was proved true and even enhanced by getting those units involved permanently for their decisions. Second, it is important to the business units to classify the existing applications for clarifying responsibility and future provision. This was distilled as a second design principle. Table 5 follows, showing the second iteration completed with findings from the surveys and the first iteration. Applications are developed on a LCDP. Due to the special environment, the developed applications are used for processes for which a department is responsible. This raises the question of the responsibility of an IT department and the support of applications. This question also arises if an employee should leave the company and thus no longer provide support for the application.

Table 5. Decision areas for LCDP governance

Decision area	Description
Principles	Principles of using connectors and a data strategy within the low-code platform
Architecture	Technical implementation of the principles
Application development and technical support	Clarification of responsibility and support of applications
Critical applications	Business critical applications and their needs
Future issues	Decisions of novel changes, especially functions, including specifications

In addition to decision areas, archetypes are an important concept of IT governance [43, 44]. The archetypes deal with the question of the responsibility of decisions. The question is, who makes the decisions from the decision areas? Although decisions are changing (in the context of LCDPs), archetypes have not been adjusted. This decision was made because the groups of people who influence decisions do not change. Further, political types were used to categorize governance decisions [42]. For LCDPs, the value increasingly evolves as the platform is used. Since this value only develops in the individual departments, the departments should be represented in governance issues. Their view may be different and should be considered. The IT managers for the LCDPs should also be present. Governance concerns who makes the decisions per decision type ("decision rights") and who is allowed to weigh in on the decision ("contribution rights"). Based on the literature findings and the special constellation of responsibilities in LCDPs, the federal structure is chosen as the input form. In further consideration, the decision areas were divided into business-oriented and technical areas since responsibility differs depending on the orientation. The "Principles", "Critical Applications" and "Future Issues" are business-oriented areas. Since the development of the applications is business-oriented, the involvement of department managers and IT managers is necessary here. This results in federal responsibility for these business-oriented areas. Critical applications are an exception within the business-oriented areas. These should also receive input through the federated form. Decisions are made in a feudal form; the feudal lords are given full control over these critical applications in their technical departments. The technical areas identified are "Architecture" and "Application development and technical support". Due to the realities of LCDPs, the input here is also federated in nature. The heads of the

technical departments contribute their views. Decisions of the technical departments are made in an IT monarchy. The technical implementation of the principles is in its decision and a purely IT-related decision and of hardly any relevance to the business departments. The archetypes are shown in Table 6.

Table 6. Archetypes of governance of LCDP

Archetypes	Input	Decision
Principles	Federal	Federal
Critical applications	Federal	Feudal
Future issues	Federal	Federal
Architecture	Federal	IT monarchy
Application development	Federal	IT monarchy

4.3. Implementation in the organization

Based on the theoretical groundwork and following the tentative design principles, the findings were now to be transferred into practice and the design principles about to get proved. In addition to a comprehensive familiarization with the Microsoft PowerPlatform, an initial stakeholder analysis was carried out in the company. This was done in support of the first design principle of business units' involvement. The identified stakeholders were able to provide insights into the various viewpoints within the group based on their position within the organization [45]. Using the results of the stakeholder analysis and the theoretical insights, a conceptual design for implementation was developed. The conceptual design is divided into three areas: "Environments and Connectors", "Monitoring", and "Other". The three areas resulted from the peculiarity of LCDPs. Topics are divided into the connections of the apps to external services ("Environments and Connectors"), the systematic evaluation of these ("Monitoring") and other topics that cannot be classified in the other two categories ("Miscellaneous"). In the area of "Environments and Connectors," the following three issues were initially identified: (1) Environments for the different environments, divided into departments and the IT department, (2) Departments are responsible for the apps in their environment, (3) Connectors are selectively blocked if critical. First, the respective departments receive their own environments, for which they are responsible. Each department defines two "owners", who are responsible for it. Depending on the environment and department, connectors are selectively blocked or unblocked in a further restrictive manner. This does justice to the special features of LCDPs. In addition, an environment is set up in which applications are checked by the IT department for external conditions. Applications in this environment are offered with a reduced group risk. The "Monitoring" area addressed two points. Firstly, the capture of innovations in the Microsoft PowerPlatform. On the one hand, this involves detecting changes in the functional scope of the Microsoft PowerPlatform and, on the other, capturing developed applications within the Microsoft PowerPlatform. On the other hand, it is about keeping the variety of connectors under control so that countermeasures can be taken if necessary. The "Other" area includes ancillary projects that support the concept. These include, among other things, the provision of resources by the IT department. This means that the departments, which are responsible for the development, are also supported in their development and learning. The IT department can also provide "on case" consulting according to budget. Likewise, processes must be defined within the organization, including a process for establishing environments. In accordance with the governance principles developed for LCDPs, a board is established in which the decisions of the decision areas are made. The owners of the environments meet regularly in this board. This board discusses architectures, best practices and lessons learned. Administrative and governance topics are also discussed in the corresponding archetypes. In Table 7, the initial concept design was taken to the practitioner's group to capture their feedback regarding the design.

Table 7. Practitioner feedback

Practitioners	Feedback
Compliance	Fewer restrictions
Data Protection	Consider business impact
Cyber Security	Higher need for monitoring
IT Operations	Custom connectors
Global Applications	"Hull" generation

Table 7 shows the practitioners' feedback. All practitioners were asked the same question. The question was how they would evaluate the concept and which aspects were important to them. This feedback was reflected in the second iteration of the concept. The following items were added per area: For "Environments and Connectors", it was: "Temporary disabling and proactive monitoring of custom connectors". For "Monitoring", it was: "Verification of developer permissions" and "Establishment of monitoring to check connections". These changes were based on the feedback. Since the IT Operations department paid attention to the custom connectors, specific monitoring was created and designed for them. To address possible "hull" generation, a review of developer permissions was suggested. Hull generation refers to the possibility of bringing "external" and thus unauthorized persons into an environment. This risk is minimized by checking permissions and the connections to any data. The resulting final concept was presented to

the practitioner's group and an end user group as an evaluation. The stakeholders rated the draft as positive. The end users group emphasized the provision of material for supporting the end users in using the platform. This led to a third design principle of supporting end users in compliance, by providing such material, guidelines, and well-designed, compliant interfaces. End users must be supported in these steps for successful governance implementation. The fourth ADR stage relies on the principle of generalized outcomes, which emphasizes that generalization is challenging because the artifact was developed to address a specific situation [32, 60]. The design principles were articulated to enable practitioners and researchers to create other similar artifacts. [46, 47]. The requirements were identified and clustered according to these domains, as in [48]. Design principles were extracted from the categorized requirements to address the overarching problem class of LCDP governance, shown in Table 8.

Table 8. Generalized design principles

DPs	Description
DP1	Classification of existing applications to clarify responsibility and future provision
DP2	Early & permanent involvement of business units to gather requirements and decisions
DP3	Support end-users in compliance by providing guidelines, materials, and compliant interfaces

These design principles emerged from the process of this work and lead to the governance of LCDPs. Therefore, based on these design principles, governance will be able to be developed for other LCDPs. The design principles capture knowledge about the process of creating solutions [43] (here for creating governance for LCDPs) and include knowledge about the creation of other instances that belong to that class [49, 50].

5. Discussion and Conclusion

Building on established studies on the topic of IT governance and LCDPs, this paper aimed to take a closer look at the challenges that arise in the introduction of governance for LCDPs, using the Microsoft PowerPlatform as an example, and to identify a possible solution. The present work focuses not only on the practical part of such an introduction in an organization, but also on the theoretical approach to the problems in this novel area and the connections to the topic complex of EUC. The topic of LCDP is currently still addressed very sparsely in the literature [56]. Although some case studies have documented how LCDPs can support application development (e.g., [51]), there is hardly any direct comparative literature. The challenges identified in the governance of LCDP are very similar to those of EUC from the 1990s (e.g., [22]). In particular, the lack of organizational alignment between EUC, management and business, low technical quality of applications, and data security and integrity [24-26], are challenges that LCDP also faces. A new addition is the distribution layer [54], which has not been considered in this way before. The provision of cloud services is viewed from the perspective of software and platforms [54]. In LCDP, the new level of cloud service addons is added, like the connectors in the context of the paper. Without governance in the area, the ever more rapidly available extensions [55] lead to a danger for the organizations and the operation of a LCDP. In the organization, one challenge was to reach a validation level for the introduction of governance in an already running system [57]. The goal of the paper was to develop a concept for establishing governance at a company on the Microsoft PowerPlatform and to take the first steps to do so. To achieve these goals, the paper was divided into several steps: (1) the creation of governance for the Microsoft PowerPlatform at a consumer goods company and the implementation of initial measures, and (2) the deduction of the significance of the governance of LCDP in relation to the field of EUC. In the first step, governance for the Microsoft PowerPlatform at the company was presented. A governance concept was developed iteratively at the various levels of environments, connectors, monitoring and others. This includes the development of a committee structure to address the new decisions that have to be made [58]. The procedure took place in subdivided governance measures, which together form the concept. Despite the difficulties in validation [57], the concept was evaluated by means of the various stakeholders, among others. Thereby the different feedback was collected, and the usefulness and effectiveness of the concept was further developed and improved [30, 45]. After the evaluation, first steps were taken on site to implement the governance. Therefore, this step represents an important pillar for achieving the objectives of the paper [54]. Although this paper serves as a first insight and reflection as well as a critical examination of a topic that has so far received little attention and has only been investigated to a little extent [51], it represents only one point of view and must not be overinterpreted in this respect. The scope of this research is limited, and the developments and characteristics found in the company are not necessarily transferable or even generalizable to other companies. Based on two aspects of the research design, it can be assumed that the contexts and problems found do represent phenomena typical of other organizations as well: first, because of the multiple distributions within very different companies with regard to the introduction

of LCDP [4], and second, because of the possibility of being able to transfer the conditions found in this company, at least in part, as a basis for other companies facing similar challenges in the introduction of LCDP. The second step was used to relate the relationship between EUC and the topic of governance of LCDPs discussed here. The second step showed that fundamental challenges of EUC in connection with LCDPs and applications that are not from IT are once again gaining relevance [52, 56, 58]. On the one hand, it was revealed that similar problems were considered in the context of EUC, such as concerns with data security and data integrity, as well as inadequate documentation and the associated loss of transparency [24-26]. Other challenges of EUC were already addressed by the nature of LCDPs. However, it became apparent that new problems also arose in the context of the LCDPs, especially through the novel integration of cloud services, through the connectors to connect to them, and through the much stronger networking possibilities. Here, the governance developed was able to embed the EUC in the overall context in a meaningful way and contribute to solving the challenges that arose. This step thus contributes to the general understanding and knowledge of the topic area of "governance of low-code platforms" and thus pays towards the goals of this paper. The evaluation of the three steps shows the effectiveness of the governance concept components and indicates a positive development when introduced in the organization. Future questions could address how other platforms introduce governance, e.g., whether the same concept can be used for this. Likewise, the question can be derived from this whether the time of introduction was chosen correctly or whether this may lead to different developments within the company. This leads to new questions on platforms governance concepts, structural decisions, roles or processes, the influence of corporate structures and employee cultures, and their relation to the development of employees in general.

6. References

[1] Microsoft Corporation. (2020). Administering a low-code development platform [Available at https://aka.ms/powerappsadminwhitepaper].

[2] vom Brocke, J., Simons, A., Riemer, K., Niehaves, B., Plattfaut, R. & Cleven, A. (2015). Standing on the shoulders of giants: Challenges and recommendations of literature search in information systems research. *Communications of the Association for Information Systems*, *37*, 205–224.

[3] Gartner (2020). Magic Quadrant for Enterprise Low-Code Application Platforms [Available at https://gtnr.it/2Uag7kp].

[4] Rymer, J., Mines, C., Hammond, J., Koplowitz, R., Vizgaitis, A. & Reese, A. (2017). Vendor Landscape: A Fork In The Road For Low-Code Development Platforms [Available at https://bit.ly/35qWIy8].

[5] Richardson, C., Rymer, J., Mines, C., Cullen, A. & Whittaker, D. (2014). New Development Platforms Emerge For Customer-Facing Applications [Available at https://bit.ly/3iJK592].

[6] Rymer, J. & Richardson, C. (2015). Low-code platforms deliver customer-facing apps fast, but will they scale up? [Available at https://bit.ly/3vAA1Cg].

[7] Alt, R., Human, S. & Neumann, G. (2020). End-user Empowerment in the Digital Age. *Proceedings of the 53rd Hawaii International Conference on System Sciences*, *53*, 4099– 4101.

[8] Benson, D. (1983). A Field Study of End User Computing. *MIS Quarterly*, *6*(2), 35–45.

[9] Dickson, G., Leitheiser, R., Wetherbe, J. & Nechis, M. (1984). Key Information Systems Issues for the 1980's. *MIS Quarterly*, *8*(3), 135–159.

[10] Alter, S. (2008). Defining information systems - implications for the IS field. *European Journal of Information Systems*, *17*(5), 448–469.

[11] Taylor, M., Moynihan, E. & Wood-Harper, A. (1998). End-user computing and information systems methodologies. *Information Systems Journal*, *8*(1), 85–96.

[12] Hackathorn, R. (1987). End-user computing by top executives. *SIGMIS Database*, *19*(1), 1–9.

[13] Carr, H. (1985). *An Empirical Investigation of the Formal Support for End User Requirements from the Information Center Concept* (Diss.). University of Texas at Arlington.

[14] Quillard, J., Rockart, J., Wilde, E., Vernon, M. & Mock, G. (1983). *A Study of the Corporate Use of Personal Computers* [Working Paper, CISR-WP-109, M.I.T.].

[15] Sprague, R. & McNurlin, B. (1986). *Information Systems Management in Practice* (1. edition). Prentice-Hall.

[16] Amorose, D. (1988). Organizational Issues of End-User Computing. *SIGMIS Database*, *19*(3-4), 49–58.

[17] Barricelli, B., Cassano, F., Fogli, D. & Piccinno, A. (2019). End-user development, end-user programming and end-user software engineering: A systematic mapping study. *Journal of Systems and Software*, *149*, 101–137.

[18] McLean, E. & Kappelman, L. (1992). The convergence of organizational and end-user computing. *Journal of Management Information Systems*, *9*(3), 145–155.

[19] Tetteroo, D., Markopoulos, P., Valtolina, S., Paternò, F., Pipek, V. & Burnett, M. (2015). End-user development in the internet of things era. *Proceedings 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA'15)*, 2405–2408.

[20] Keen, P. & Woodman, L. (1984). What To Do With All Those Micros. *Harvard Business Review*.

[21] Chou, H.-W., Coakley, J. & Harrison, W. (1995). End User Computing Management Practices in Taiwan and the U.S. *PACIS 1995 Proceedings*, *96*, 434–438.

[22] O'Donnel, D. & March, S. (1987). End-user computing environments - Finding a balance between productivity and control. *Information & Management*, *13*(2), 77–84.

[24] Dickson, G. & Wetherbe, J. (1985). *The Management of Information Systems*. McGraw-Hill.

[25] Martin, J. (1982). *Application Development Without Programmers*. Prentice-Hall Inc.
[26] Bergeron, F. & Berube, C. (1988). The Management of the End User Environment: An Empirical Investigation. *Information & Management*, *14*, 107–113.
[27] Henson, R. & Kuzma, J. (2010). End User Computing and Information Security: a retrospective look at the decentralisation of data processing and emerging organisational information risk. *UK Academy for Information Systems Proceedings 2010*, *27*.
[28] Claussen, J., Kretschmer, T. & Mayrhofer, P. (2013). The Effects of Rewarding User Engagement: The Case of Facebook Apps. *Information Systems Research*, *24*(1), 186– 200.
[29] Microsoft Corporation. (2021). Microsoft PowerPlatform [Available at https://bit.ly/3xn999W].
[30] Cooper, H. (1988). Organizing Knowledge Syntheses: A Taxonomy of Literature Reviews. *Knowledge in society, 1*(1), 104-126.
[31] Gimpel, H., Kerpedzhiew, G., König, F. & Meindl, O. (2020). Teaching an old work system new tricks: Towards an integrated method for work system transformation in times of digitalization. *ECIS 2020 Research Paper, 37*.
[32] Sein, M., Henfridsson, O., Purao, S., Rossi, M. & Lindgren, R. (2011). Action Design Research. *MIS Quarterly*, *35*(1), 37–56.
[33] Denner, M.-S., Püschel, L.C. & Röglinger, M. (2018). How to Exploit the Digitalization Potential of Business Processes. *Business & Information System Engineering*, *60*(4), 331-349.
[34] March, S.T. & Smith, G. (1995). Design and natural science research on information technology. *Decision Support Systems, 15*(4), 251-266.
[35] Moloney, M. & Church, L. (2012). Engaged Scholarship: Action Design Research for new Software Product Development. *ICIS 2012 Proceedings*. Orlando, USA.
[37] Webster, J. & Watson, R. (2002). Analyzing the Past to Prepare for the Future: Writing a Literature Review. *MIS Quarterly*, *26*(2), xiii–xxiii.
[38] Craig, A. & Dixon, R. (2001). Issues To Consider When End-Users Develop Their Own Applications. *ACIS 2001 Proceedings*, *17*.
[39] Perez, I. & Sturm, A. (2014). Can End-Users Program? *Proceedings of the European Conference on Information Systems 2014*.
[40] Beimborn, D., Miletzki, T. & Wenzel, S. (2011). Platform as a Service (PaaS). *Wirtschaftsinformatik*, *53*, 371–375.
[41] Sahay, A., Indamutsa, A., Ruscio, D. D. & Pierantonio, A. (2020). Supporting the understanding and comparison of low-code development platforms. *46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, 171–178.
[42] Weill, P. (2004). Don't Just Lead, Govern: How Top-Performing Firms Govern IT. *MIS Quarterly Executive*, *3*(1), 1–17.
[43] Cronholm, S. & Göbel, H. (2019). Design Science Research Constructs: a Conceptual Model. *PACIS 2019 Proceedings*. 6.
[44] Tiwana, A., Konsynski, B. & Venkatraman, N. (2013). Special Issue: Information Technology and Organizational Governance: The IT Governance Cube. *Journal of Management Information Systems*, *30*(3), 7–12.
[45] Brugha, R. & Varvasovszky, Z. (2000). Stakeholder analysis. *Health Policy and Planning*, *15*(3), 239–246.
[46] Markus, M., Majchrzak, A. & Gasser, L. (2002). A design theory for systems that support emergent knowledge processes. *MIS Quarterly*, *26*(3), 179–212.
[47] Walls, J., Widmeyer, G. & Sawy, O. E. (1992). Building an information system design theory for vigilant EIS. *Information Systems Research*, *3*(1), 36–59.
[48] Schacht, S., Morana, S. & Maedche, A. (2015). The Evolution of Design Principles Enabling Knowledge Reuse for Projects: An Action Design Research Project. *Journal of Infomation Technology Theory and Application*, *16*(3), 5–36.
[49] Dasgupta, P. (1996). The economics of the environment. *Environment and Development Economics, 1*(4), 387-428.
[50] Purao, S. (2002). *Design Research in the Technology of Information Systems: Truth or Dare* [Working Paper].
[51] Iho, S., Krejci, D. & Missonier, S. (2021). Supporting Knowledge Integration with Low-Code Development Platforms. *ECIS 2021 Research Papers*, *38*.
[52] Totterdale, R. (2018). Case Study: the Utilization of Low-Code Development Technology to Support Research Data Collection. *Issues in Information Systems*, *19*(2), 132–139.
[53] Zolotas, C., Chatzidimitriou, K. & Symeonidis, A. (2018). RESTsec: a low-code platform for generating secure by design enterprise services. *Enterprise Information Systems*, *12*(8–9), 1007–1033.
[54] Alt, R., Leimeister, J.M., Priemuth, T., Sachse, S., Urbach, N. & Wunderlich, N. (2020). Software-Defined Business. *Business & Information Systems Engineering, 62*(6), 609-621.
[55] Rymer, J. (2018). Why You Need To Know About Low-Code, Even If You're Not Responsible For Software Delivery. [Available at: https://bit.ly/3pWBK3o]
[56] Prinz, N., Rentrop, C. & Huber, M. (2021). Low-Code Development Platforms – A Literature Review. Proceedings of the AMCIS 2021.
[57] Jafarijoo, M. & Joshi, K. (2020). How Do Firms Derive Value from Cloud Computing Investment? Examining the Role of Information Technology Governance. *AMCIS 2020 Proceedings*.
[58] Klotz, S., Westner, M. & Strahringer, S. (2020). From shadow IT to business-managed IT and back again: how responsibility for IT instances evolves over time. *PACIS 2020 Proceedings*.
[59] Tisi, M., Mottu, J.-M., Kolovos, D., de Lara, J., Guerra, E., Di Ruscio, D., Pierantonio, A. & Wimmer, M. (2019). Lowcomote: Training the Next Generation of Experts in Scalable Low-Code Engineering Platforms. *STAF 2019.*
[60] Petersson, A. & Lundberg, J. (2016). Applying action design research (ADR) to develop concept generation and selection methods. *Procedia CIRP 50*, 222-227.
[61] Lewandowski, T., Delling, J., Grotherr, C. & Böhmann, T. (2021). State-of-the-Art Analysis of Adopting AI-based Conversational Agents in Organizations. *PACIS 2021 Proceedings*. 8.