

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

A Multiple Mini Case Study on the Adoption of Low Code Development Platforms in Work Systems

Sebastian Käss¹, Susanne Strahinger², and Markus Westner¹

¹Faculty of Computer Science and Mathematics, OTH Regensburg, Regensburg, Germany

²Faculty of Business and Economics, TU Dresden, Dresden, Germany

Corresponding author: Markus Westner (e-mail: Markus.Westner@oth-regensburg.de).

ABSTRACT Although adopting Low Code Development Platforms (LCDPs) promises significant efficiency and effectiveness improvements for application development, its adoption still needs further empirical research. This paper uses a combinatorial approach to research LCDP adoption and presents the results of a multiple mini case study with 36 cases on LCDP adoption. A combination of the Socio-Technical Systems theory and the Technology-Organisational-Environment model is used as a theoretical lens. In this paper, we show that LCDP adoption is a multifaceted phenomenon and identify three archetypes for LCDP adoption (i.e., IT Resource Shortage Mitigators, Application Development Democratisers, and Synergy Realisers) and one archetype for LCDP non-adoption (i.e., Intricacy Adversaries). Each archetype can be interpreted as an individual path towards LCDP (non-)adoption. Based on these archetypes, we derive seven starting points for practitioners to adopt LCDPs in work systems. Moreover, by using the theoretical lenses, the paper shows that for an LCDP adoption to occur, an optimisation of the social and technical sub-systems is required.

INDEX TERMS Low Code Development Platform, Low Code, Adoption, Case Study

I. INTRODUCTION

Information Technology (IT) is a crucial driver for organisations to create value and stay competitive [1]. Therefore, organisations must increase their speed in developing applications within budget and time constraints [2]. However, a significant gap in skilled developers for application development exists [3], e.g., with over 137,000 IT vacancies in 2022 in Germany [4]. Traditionally, outsourcing has been a common strategy applied to address this shortage [5]. However, recent developments suggest that this approach is becoming less attractive [5]. Low Code Development Platforms (LCDPs) are promoted to address these challenges by increasing efficiency and effectiveness for application development and by empowering end users to develop applications [6]–[8]. For instance, Forrester Research claims that LCDPs “*have the potential to make software development as much as 10 times faster than traditional [development] methods*” [9]. LCDPs are advertised to support professional software developers and so-called *citizen developers* with little to no programming experience [6], [7]. Therefore, the global

market size for LCDPs is expected to grow significantly, from USD 13.2 billion in 2020 to USD 45.5 billion by 2025 with a Compound Annual Growth Rate of 28 % [10].

Despite practitioners' widespread adoption of LCDPs, academia has just recently begun researching LCDP adoption. However, a better understanding of LCDP adoption can help to explain the rapid adoption growth and to address factors hindering it. Therefore, we researched LCDP adoption in work systems and identified 12 drivers and 19 inhibitors as factors for LCDP adoption in [11]. We conducted a Delphi study, analysed the drivers' and inhibitors' importance, and found that experts agreed on the most and least important drivers and inhibitors for adoption [11]. However, for the factors between the most and least important, the experts' ranking was context-dependent [11]. Our Delphi study only researched single factors and neglected the combinatorial effects between different factors. Another study researched LCDP adoption on an organisational level with 18 semi-structured expert interviews, archival data, and user reviews [1]. The authors identified ten aspects supporting and six aspects hindering the

adoption of LCDPs [1]. Furthermore, in [1] it was argued that researching LCDP adoption just from a technology adoption perspective falls short, as Low Code Development (LCD) is an entirely new way of application development. Therefore, a combination of technical, organisational and environmental aspects must be considered when researching LCDP adoption [1]. We argued in [12] in the same direction and proposed a combination of technical, organisational, and environmental factors to research LCDP adoption in work systems. To represent the technical aspects, we considered results from previous cloud computing (CC) adoption research. We justified this by the Platform as a Service (PaaS) cloud delivery model used by LCDPs [12]–[14]. We included factors from agile software development method (SDM) adoption to consider the transformed application development approach [12]. Moreover, we showed that LCDP adoption is a multifaceted phenomenon which should be researched by combining different factors that influence each other [12]. Consequently, we built a combinatorial research model with 13 factors in [12]. Nevertheless, we could not yet empirically validate the model, which is the purpose of this paper. Therefore, the paper at hand addresses the following research questions (RQ):

RQ1: What factors determine LCDP adoption in work systems?

RQ2: Which combinations of these factors lead to LCDP adoption in work systems?

The unit of analysis is the LCDP adoption decision in work systems, i.e., “*systems in which human participants and/or machines perform work [...] using information, technology, and other resources*” [15, p. 75]. The focus is on work systems where professional and citizen developers adopt LCDPs as systems to develop applications. Academic literature distinguishes between the adoption and post-adoption phases [16]. In the adoption phase, an organisational view is taken, i.e., the organisation must decide upon the adoption, selects a provider, and enables the usage within the organisation [16]. In the post-adoption phase, the decision to adopt the LCDP for a specific project is taken in a work system. Therefore, this paper researches the post-adoption phase. To answer the research questions, we use a multiple case study, which allows us to research the adoption in its real-world context. However, as we collect 36 cases to be able to investigate combinatorial effects and each case has a somewhat limited depth (i.e., one interview per case), we call the methodology a multiple mini case study. A similar approach was chosen by [17] when researching Software as a Service adoption.

The paper at hand has four contributions. First, we confirm ten of the 13 factors for LCDP adoption from our prior research published in [12]. Furthermore, we find two additional factors (i.e., the *sophistication of the to-be-developed application* and the *replacement of shadow IT*) that are important for LCDP adoption. Second, we show that LCDP adoption results from a combination of different factors and is a multifaceted phenomenon. Based on this, we identify three archetypes for LCDP adoption (i.e., *IT Resource*

Shortage Mitigators, *Application Development Democratisers*, and *Synergy Realisers*) and one archetype for LCDP non-adoption (i.e., *Intricacy Adversaries*). Third, based on the Socio-Technical Systems (STS) theory, we show that the technical and social sub-systems must be jointly optimised for LCDP adoption to occur. Finally, based on the LCDP adoption archetypes, we derive seven starting points for practitioners for LCDP adoption.

The paper is structured as follows: Section two outlines the conceptual background, introduces the research model, and presents the current state of research on LCDP adoption. Section three presents the methodology of the multiple mini case study, and the results are shown in section four. In section five, we discuss the results considering previous literature and derive practical starting points. The study concludes with the contributions, limitations, and presentation of further research directions.

II. Research Background

A. LCDPs – Conceptual Background

LCD offers a new approach to developing applications using interactive graphical interfaces instead of traditional programming [1], [6]. For LCD, so-called LCDPs are used to foster application development through visual, drag-and-drop techniques. Therefore, LCDPs are easy to start with for developers with and without prior knowledge [11], [18], [19]. LCDPs operate as a PaaS cloud delivery model [12]–[14] and enable citizen developers to quickly develop various applications [18]. As such, LCDPs fit the broader trend of technology democratisation (i.e., traditionally required coding for application development can now be accomplished by citizen developers) [18]. In academic literature, this trend to democratise technology and bring a higher degree of responsibility for IT components and tasks to the business departments is described as business-managed IT [20]–[22]. It is unsurprising that multiple software vendors have built LCD functionality in their product portfolio in recent years, e.g. SAP [23], ServiceNow [24], Salesforce [25], or Oracle [26].

LCDPs combine previously well-known concepts such as rapid application development, fourth-generation programming languages, model-driven development, and computer-aided software engineering [7], [27]–[30]. Therefore, literature agrees that the trend of LCDPs is not an entirely new phenomenon [11]. The term LCDP describes a list of heterogeneous development platforms with different functionalities, multiple scenarios of use, and different target audiences [11], [7]. We focus on LCDPs used by citizen developers and professional developers within an organisation to develop applications. Hence, these LCDPs allow full application development with a user interface, business logic, workflows, and connected data services [31]. Gartner refers to this type of LCDP as Low Code Application Platforms and leading vendors in this market are Mendix, Microsoft, OutSystems, Salesforce, and ServiceNow [32].

B. LCDP Adoption

I. LCDP ADOPTION RESEARCH

Technology adoption is the diffusion and first use of technology within a work system [33]. Research on technology adoption distinguishes various levels of adoption, from organisational to individual adoption [34]. Academic literature further differentiates between the initial adoption phase and the subsequent post-adoption phase [16]. During the adoption phase, organisations decide on the adoption, choose a provider, and facilitate implementation within the organisation [16]. In the post-adoption phase, individuals or work systems decide to adopt the LCDPs, e.g., for a specific project. For this paper, we take a work system view, due to three reasons. First, LCDPs promote the collaborative application development with multiple developers, i.e., an application is developed by more than one individual. Therefore, the adoption usually occurs on a higher level than the individual level. Second, the goal of LCDP adoption is to develop many applications within a department. Thus, many post-adoption decisions occur on the work system level (whereas an organisational adoption occurs only once – when the organisation decides to adopt the LCDP). Therefore, the work system view helps to research the majority of LCDP adoptions. Third, compared to traditional technology adoption, LCDP adoption differs, as LCDP adoption is “*a technology and method of software development where both elements [technology and method] are intertwined*” [1, p. 1]. Thus, the relationship between the technology and the human participants needs to be analysed. Taking a work system view, helps to analyse this relationship in-detail.

From an academic perspective, a few papers research LCDP adoption from different lenses. In a recent literature review [35] summarise academic literature on drivers and inhibitors for LCDP adoption. The authors find that improved software development efficiency and reduced entry barriers for application development are the most often discussed drivers for LCDP adoption [35]. Moreover, in [36], posts from LCDP online forums are analysed to research LCDP adoption on an individual level. The authors conclude that faster development (i.e., higher efficiency) and lower development complexity are the most often discussed drivers for LCDP adoption. The authors of [37] research LCDP adoption on an organisational level by surveying IT experts and find that accelerated digital transformation and reduced dependency on IT developers are the main drivers for LCDP adoption. Another paper that researches LCDP adoption on an organisational level is [1]. The authors argue that LCDP adoption is a technology adoption that also transforms the application development process [1]. Therefore, a combination of technical, organisational, and environmental aspects must be studied to research LCDP adoption [1]. In summary, the authors identify ten aspects supporting and six hindering the adoption of LCDPs and structure them along the Technology-Organisation-Environment (TOE) model. However, none of the previously discussed empirical research takes a work system view for the analysis.

To the best of our knowledge, the only empirical research that applies a work system view for LCDP adoption is our Delphi study in [11]. We conducted a ranking-type Delphi study with 17 experts for LCDP adoption and identified 12 drivers and 19 inhibitors for LCDP adoption in work systems [11]. The experts in our study agree that the most important drivers are the improved efficiency of software development, reduced entry barriers for application development, and reduced required knowledge for application development [11]. The experts also agree that the most important inhibitors are the lack of LCD culture and reluctance to change. Moreover, the experts highlight that the drivers and inhibitors influence each other, and a single factor is not sufficient to explain LCDP adoption [11]. Therefore, LCDP adoption in work systems should be researched with a combinatorial approach [11], which cannot be done by a Delphi study. Therefore, we decided to build a combinatorial research model which we published in [12]. In the paper at hand, we test this research model for LCDP adoption with a multiple mini case study. This multiple mini case study approach allows us to research a contemporary phenomenon (i.e., LCDP adoption) in its real-world context [38].

II. RESEARCH MODEL

A. Theoretical Lenses for LCDP Adoption

Most previous research on LCDP adoption is practice-oriented and does not apply a theoretical lens [12]. Therefore, for this study, we use the research model and theoretical lenses developed in our prior research in [12]. To build the research model, we derived 13 empirically testable factors from existing CC and agile SDM adoption research. We describe the core components and relationships of the research model in the following section. The full model is published in [12].

To research CC adoption, various technology adoption theories have been used [34]. One research direction that examines the adoption on an individual level uses the Technology Acceptance Model or its extensions [34]. However, this paper focuses on the adoption within work systems (i.e., at a higher level). Thus the models proposed by a second research direction, specifically the TOE model [34], [39] are more suitable. The TOE model proposes that the adoption of an innovation is influenced by a combination of *technology*, *organisational*, and *environmental* factors [40], [41]. Due to its simplicity, the TOE model has been used frequently to research why organisations adopt novel technologies [1], e.g., CC [42], [43], Enterprise Resource Planning (ERP) systems [44], e-business systems [45], and organisational adoption of LCDPs [1].

In the TOE model, the *technology* includes all technologies (e.g., equipment and processes) relevant to the organisation [40], [46]. The *organisation* describes multiple characteristics, such as the management structure and support, culture, technical competence, availability of resources, organisational readiness, technical competence, or organisational size [46]. Several studies suggest that an organisation's characteristics are crucial when researching technology adoption [1]. The *environment* consists of all

external factors that affect technology adoption [40], [41], [47]. The *environment* usually includes factors like the industry's structure, regulatory environment, and the pressure of technology providers, which influence the adoption decision [46].

Due to its simplicity, the TOE model is often criticised for its generic nature [48]. Therefore, we suggest using it as a guiding framework but extending the *technology* and *organisation* components with elements from the STS theory for further analysis [12]. The STS theory was already used by

[49] to research the adoption of computer-aided software engineering systems, which have conceptual similarities with LCDPs [28], [30]. The STS theory analyses the interaction between the *social sub-system* and *technical sub-system*, both of which a work system comprises [50]–[52]. According to [50], the *social sub-system* comprises a *structure sub-system* and a *people sub-system*. In contrast, the *technical sub-system* encompasses a *technology sub-system* and a *task sub-system*.

TOE	STS	Factor	Description	Effect from previous research
Technology	Technology sub-system	Security and data privacy concerns	Security and data privacy concerns encompass the fear of decision-makers regarding the possible exposure of data by the CC provider, the sharing and virtualisation of resources, and the transfer of data over the Internet [53].	Mixed
		Compatibility	Compatibility is the alignment of innovations with an organisation's values, experience, requirements, and technical infrastructure [54].	Positive
		Vendor lock-in	Vendor lock-in occurs when customers become heavily reliant on a particular CC vendor due to proprietary systems that hinder their ability to switch to alternative vendors [55], [56].	Negative
	Task sub-system	Expected efficiency improvements	Expected efficiency improvements encompass cost savings, decreased time to market, and overall improvements in business performance [12].	Positive
Organisation	People sub-system	Complexity	Complexity is the degree to which an innovation is perceived as relatively complex to use [54].	Negative
		Previous experience	Previous experience is the extent to which an innovation can be tested before its adoption or the knowledge gained from pilot projects [56].	Positive
		Training opportunities	Training opportunities are the formal mechanisms employed by an organisation to promote and facilitate learning [12].	Positive
		Organisational culture	Organisational culture is a set of established assumptions developed by a group as it learns to address external challenges and internal integration, which are deemed valid and taught to new members as the appropriate way to perceive, think, and respond to those challenges [57].	Positive
		Usefulness	Usefulness refers to the extent to which the adoption of an LCDP is perceived as beneficial [12].	Positive
	Structure sub-system	Expected working mode improvements	Expected working mode improvements comprise all working mode-related factors, e.g., teamwork, communication, and a shared understanding between business and IT units [12].	Mixed
		Top management support	Top management support is the executives' active and enthusiastic approval of innovation [58].	Positive
		Internal IT capabilities	Internal IT capabilities comprise the availability and expertise of internal IT resources and IT infrastructure [56].	Positive
Environment	n/a	External pressure	External pressure encompasses the pressure from industry competitors, trends, trading partners, and the regulatory environment [45].	Mixed

Table 1: Research model for LCDP adoption based on [12]

In the context of LCDPs, these sub-systems are already defined by [14]: The *structure sub-system* is the nature of an organisation's communication, authority, and setup; the *people sub-system* comprises all stakeholder-specific factors; the *task sub-system* includes all application development-specific factors, whereas the *technology sub-system* includes all platform-specific issues [14]. Based on the STS theory, we can assume that for work systems to achieve their goals (i.e., adopting a technology), the *social* and *technical sub-systems* must be balanced and jointly optimised [12], [59]. Moreover, sub-systems must be open and responsive to their environment [12], [59].

In summary, we derive three tentative propositions from the theoretical lenses for the combinatorial research model. First, a combination of technical, organisational, and environmental factors must be considered when researching LCDP adoption (as indicated by the TOE model and STS theory) [12]. Second, multiple sub-systems must be balanced and jointly optimised for LCDP adoption (as indicated by the STS theory). Third, environmental factors are essential for LCDP adoption (as indicated by the TOE model and STS theory) [12].

B. Resulting Research Model

Based on the theoretical lenses and findings from prior CC and agile SDM adoption research, we constructed a research model (cf. Table 1) that is the basis for our empirical research in [12]. We used the analogy between CC and agile SDM adoption to derive the factors for the research model [12]. We argue that as LCDPs are cloud services using a PaaS delivery model [14], all factors which were found to influence CC adoption in general must also influence LCDP adoption [12]. Moreover, the adoption of LCDPs changes application development significantly [1]; from a process with traditional programming to a process with visual drag-and-drop functions [1], [6], [60]. To account for these changes, we included previous findings from agile SDM adoption literature [12].

Previous research is ambiguous regarding the effect of individual factors. Therefore, we propose to use a conceptual approach, where combinations of multiple factors cause the outcome (adoption or non-adoption) [12]. Existing CC adoption literature finds asymmetric effects of factors [12], e.g., high security concerns leading to lower CC adoption [39], with other studies showing the opposite [61]. Additionally, for IT innovation adoption in general, factors are found to influence themselves as a form of conjunctural causation [62].

Table 1 summarises the research model we developed in [12] and provides an overview of the suggested sub-systems, factors to capture LCDP adoption, the factors' descriptions, and the effect on the adoption found in previous research. If both positive and negative effects were found by previous research, the term "mixed" is used in Table 1 in addition to the terms "positive" and "negative". However, since prior studies on CC and agile SDM adoption lack a combinatorial approach, we suggest in [12] to limit ourselves to considering these factors as essential for explaining LCDP adoption without specifying their combined effect on LCDP adoption.

III. METHODOLOGY

A. Overview

For this paper, we followed a multiple case study approach to answer our research questions. This approach enables us to investigate the contemporary phenomenon of LCDP adoption in

its real-world context [38]. Since our research has a relatively high number of cases (i.e., a high breadth), with each case only having one semi-structured interview as a source of evidence, we refer to it as a multiple mini case study. The rationale for choosing a multiple mini case study is the complex research model (13 factors) requiring many cases to investigate combinatorial effects. A benefit of this high number of cases is that it allows us to identify archetypes of cases with similar characteristics, across which we can triangulate findings. To answer the research questions, we followed a four-stepped approach for conducting this case study, (1) designing the research, (2) collecting the data, (3) analysing the data, and (4) composing the case study report as proposed by [38]. In the following sections, we elaborate on our methodological decisions and the measures taken to ensure the result's validity.

B. Designing the Research

As a first step, we formulated the research questions (i.e., RQ1 and RQ2) and clarified the unit of analysis (i.e., the LCDP adoption decision in work systems) [38]. This work systems perspective allows us to research the adoption on a level where it usually occurs – between the individual developer and an organisational level. We used our research model [12] as an a-priori theoretical framework to guide research design, data collection, and data analysis.

As case study research builds on analytical generalisation from cases, the case replication logic is crucial to define the underlying rationale for selecting cases [38], [63]. We combined literal and theoretical replication to select our cases and strengthen our research's robustness and generalisability [38], [63]. Literal replication allows us to predict similar results for cases with similar characteristics [38]. Theoretical replication allows us to identify cases with varying outcomes based on theoretically predictable reasons [63]. Literal replication was ensured by selecting homogenous work systems from large organisations. For theoretical replication, we selected informants with a background from the business and IT units, as we found in our previous paper (i.e., [11]) that informants from the business and IT units have a different view on LCDP adoption. Moreover, all informants had to be decision-makers while deciding on LCDP adoption within a work system (e.g., an LCDP owner) or actively consulting in these decision situations (e.g., a consultant). We identified the experts through our personal professional network and the social network LinkedIn. For LinkedIn, we searched for informants from German Dax, MDax or SDax companies in combination with the search string "Low Code". These indices include the stocks of the 40 largest German companies (Dax), but also 50 mid-cap (MDax), and 70 small-cap (SDax) companies.

For all informants identified, we evaluated if the informants were decision-makers or consultants in the described decision situations. We found 23 informants, of which nine had a background in consulting, and 14 were line managers. An overview of the informant's characteristics can be found in Appendix A. Furthermore, all informants gave their consent to publish the collected data in this research article.

C. Data Collection

We first sent each informant an online questionnaire with context questions (based on the contexts found in [11]). This approach allows us to obtain standardised context information to increase comparability. We then invited the informants to participate in

semi-structured interviews [38] as our primary data collection method to gather in-depth information about LCDP adoption. The interview guide was derived from the research questions and the theoretical framework in [12] to ensure a full chain of evidence from the research questions to the results. All interviews were structured in six parts: (1) introduction, (2) current state of LCDP adoption on an organisational level, (3) process to adopt LCDPs in work systems, (4) factors for LCDP adoption in work systems, (5) key learnings for LCDP adoption in work systems, (6) closing remarks. The semi-structured questionnaire can be found in Appendix B. Data collection took place via video conferences between February to April 2023. The interviews had an average duration of 45 minutes, were recorded, transcribed, anonymised, and analysed. We captured 36 cases, with 26 adoption cases and ten non-adoption cases. A description of the cases and the context variables can be found in Appendix C and D. Furthermore, to increase the case studies' reliability, we used a case study protocol and a case study database (MAXQDA) [38], [63]. MAXQDA is a software solution to store, categorise, and analyse qualitative data [64].

D. Data Analysis

We analysed the collected data from the context questionnaire and interviews (i.e., used multiple sources of evidence [38]) in several iterations. We started with a within-case analysis by first coding our data along the 13 factors of the analytical framework, which we developed in a previous paper [12]. In the second iteration, we used open coding to identify potential new factors from the case data. For each case, we extracted the output (adoption or non-adoption decision), the factors leading to the output, the value of the factors, and the factors' effects. Based on these results, we analysed whether a factor was relevant to the LCDP (non-)adoption decision. We then identified archetypes and conducted a pattern matching, i.e., matching the factors and their effect from the empirical data to theoretically predicted effects [38]. This pattern-matching approach increased our findings' robustness [38]. We conducted the cross-case analysis based on our identified archetypes with the goal to understand the underlying effects of LCDP adoption better.

E. Composition

We only reported findings found in at least two cases to ensure robustness and mitigate the risk of reporting spurious results [38], [63]. Moreover, we also followed the recommendations for a convincing composition, e.g., a clear transferability of results, suitable rhetoric, and the empowerment of readers [63].

F. Validity

A crucial part of multiple case study research is ensuring its rigour. Four criteria are used for this purpose: *Construct validity*, *internal validity*, *external validity*, and *reliability* [38], [65], [66]. As multiple mini case studies have fewer sources of evidence per case, and because of the general criticism that many case study papers do not show how rigour is ensured [65], [66], we elaborate in the following section on our methodological steps taken to ensure rigour.

Construct validity is about ensuring the development of correct operational measures for the researched concepts [38]. To ensure construct validity, we triangulated across multiple sources of evidence (interview and context questionnaire), and established a chain of evidence from the a priori framework to the results [38]. *Internal validity* is about establishing a causal

relationship [38]. We, therefore, applied pattern matching [38] and formed archetypes of cases between which results can be triangulated. *External validity* defines the domain to which a study's findings can be generalised [38]. To ensure external validity, we used a multiple case (36 cases) design, with a combination of theoretical and literal replication, to allow the generalisability of results [38]. Furthermore, we clearly defined the unit of analysis. The aim of *reliability* is to ensure that the steps of a study can be repeated and lead to the same results [38]. To ensure reliability, we followed the methodological guidance of [38] and built a case study database (MAXQDA) and a case study protocol to allow replication of our research.

IV. RESULTS

A. Overarching

In the following sections, we first outline overarching results from the context questions, second identify factors for the LCDP (non-)adoption based on the research model and open coding, and third introduce archetypes of LCDP (non-)adoption. In the context of this paper, archetypes are typical examples of LCDP (non-)adoption cases and are defined by the occurrence of a set of factors. All results of the context questions can be found in Appendix D.

We captured 36 cases, with 26 being adoption and ten non-adoption cases. The organisations to which the work systems belong, have a background from various industries, with industrial manufacturing (22%), financial services (17%), healthcare and life sciences (17%), and conglomerates (11%) being the most frequent. All organisations can be considered large with a global footprint. This is also confirmed by the results of the context questionnaire, where in only 21% of the cases, the IT department provides IT services to only one continent. Moreover, the informants consider their IT units rather centralised, whereas the business units are rather de-centralised. In most cases, the LCDP is Mendix (39%), Microsoft Power Platform (25%), or OutSystems (14%). Moreover, we also have fewer cases where the LCDP is Appian, Oracle APEX, or RCP. Many case organisations claim to be in an early stage of LCDP adoption, as they only have a few low code applications (i.e., only 26 % have more than 100 low code applications). When analysing who develops on the LCDP, in the majority (32%) of the cases, the IT unit and external developers develop on the LCDP, followed by only the IT unit (16%), business and IT units (16%), business unit only (11%), and IT unit, business unit, and external developers (11%). Hence, in most cases, the IT unit is involved in the (non-)adoption decision. The informants explain the lower usage of only citizen development with a lack of governance concepts, the fear of building an island application landscape, and security concerns from citizen development. As I-09 put it: "We do not push the topic citizen development [...] as we as IT unit or organisation, in general, do not have a concept on how we want to deal with it." However, a larger-scale introduction of citizen development is piloted or planned to be piloted in multiple organisations.

The process for adopting an LCDP in a work system is primarily project-specific (i.e., the business unit wants an application to be developed, approaches the IT unit, a joint evaluation, if an LCDP is the right solution is done, and then the adoption decision is taken). Therefore, in the following, each case represents the adoption decision for a to-be-developed application in a work system.

B. Empirically Found Factors for LCDP Adoption

In the following section, we present the empirically found factors for LCDP post-adoption. As shown in Table 2, we identified ten of the 13 factors from the research model (cf. Table 1) and two additional factors from open coding during the within-case analysis. In the adoption cases, the most often found factors are *expected efficiency improvements* (26 of 26), *sophistication of the to-be-developed application* (15 of 26), *compatibility* (14 of 26), and *internal IT capabilities* (12 of 26). For the non-adoption, the most often found factors are the same, with *sophistication of the to-be-developed application* (ten of ten), *expected efficiency improvements* (seven of ten), *compatibility* (six of ten), and *internal IT capabilities* (six of ten). The two newly identified factors are *sophistication of the to-be-developed application* and *replacement of shadow IT*. The factor *sophistication of the to-be-developed application* refers to the fact that the LCDPs are better suitable to develop small, simple applications and not large, complex ones. Multiple informants highlighted the importance of this factor, e.g., “[The evaluation] also contains soft factors, like low complexity [of the to-be-developed application]” (I-15) or “[The] reporting functionality ended up not being at the same level as [in specialised software], and there would simply have been no advantage to using [a LCDP] in these use cases. Another point is performance, especially when you must process large amount of data” (I-21). As this factor refers to the *sophistication of to-be-developed application* in LCDPs, and all development-related factors are in the task sub-system, we propose to allocate this factor to the task sub-system. The factor *replacement of shadow IT* is the adoption of an LCDP to provide (citizen) developers a platform where they can develop applications in a way compliant with organisational guidelines, as illustrated by I-03: “There is this issue of shadow

IT. Many applications are not apps, but it is still an application because people work with them daily. [...] And we lift it out of the shadows and into the official path that conforms to corporate guidelines. The person in charge of something like this can be sure: Okay, I am compliant.” As this factor’s underlying goal is to comply with the organisation’s guidelines (i.e., authority), we propose to allocate this factor to the structure sub-system.

The factors of *vendor lock-in*, *previous experience*, and *training opportunities* from the research model are not found in our cases. The non-occurrence of the factor *vendor lock-in* can be explained by the work system (i.e., post-adoption) view taken for this paper. The informants explained that *vendor lock-in* is a factor to consider during the adoption decision on an organisational level. However, for the adoption on a work system level (post-adoption), “*this does not play any role*” (I-03). In our cases, *security and data privacy concerns* often result from *external pressure* (e.g., *security and data privacy concerns* as the regulator requires a full auditability of the applications).

C. Archetypes for LCDP (Non-)Adoption

I. GENERAL

A combination of factors from the previous section lead to the (non-)adoption of LCDPs. Each factor has a value (e.g., low or high) and an effect (negative or positive) on the LCDP (non-) adoption. The same combination of factors, values, and effects always lead to the same outcome. The coding table for all cases can be found in Appendix E. In the next sections, we present three archetypes for LCDP adoption and one archetype of LCDP non-adoption. The archetypes were created based on the frequency of occurrence of the factors and the interpretation of these cases.

TOE	STS	Factor	# of occurrences			Source	
			Adoption	Non-adoption	Total	Research model	Open coding
Technology	Technology sub-system	Security and data privacy concerns	1	1	2	X	
		Compatibility	14	6	20	X	
		Vendor lock-in	0	0	0	X	
	Task sub-system	Expected efficiency improvements	26	7	33	X	
		Sophistication of the to-be-developed application	15	10	25		X
Organisation	People sub-system	Complexity	5	1	6	X	
		Previous experience	0	0	0	X	
		Training opportunities	0	0	0	X	
		Organisational culture	7	1	8	X	
		Usefulness	0	1	1	X	
	Structure sub-system	Expected working mode improvements	8	1	9	X	
		Top management support	7	2	9	X	
		Internal IT capabilities	12	6	18	X	
		Replacement of shadow IT	3	0	3		X
		External pressure	3	2	5	X	
Environment	n/a	External pressure	3	2	5	X	

Table 2: Occurrence of factors in analysed cases; (x) indicates source of factor

II. ADOPTION

In all adoption cases the factor *expected efficiency improvements* with the value high and a positive effect occurs. Moreover, in all adoption cases the majority of factors positively affect LCDP adoption and two to six factors occur in our analysed adoption cases. As shown in Table 3, we identified three archetypes for LCDP adoption based on the interpretation of the cases: (1) *IT Resource Shortage Mitigators*, (2) *Application Development Democratisers*, and (3) *Synergy Realisers*. The following sections present the archetypes and the factors defining them.

The first archetype is *IT Resource Shortage Mitigators* with the underlying goal of building applications efficiently with limited traditional IT resources. Therefore, this archetype comprises the factors *expected efficiency improvements*, with the value high having a positive effect, and *internal IT capabilities*, with a positive effect. For case three, the informant pointed out that *expected efficiency improvements* were one of the key reasons for the LCDP adoption: “*The expectation is, in any case, that this [the development] is done very, very fast*” (I-02). Furthermore, in this case, there was a lack of IT resources (i.e., low *internal IT capabilities*) that promoted the usage of LCDPs to develop the required application (“*It is key that we have this application*” I-02). Another informant elaborated that when using an LCDP, the development of “*the application [could be done] in a tenth of the time*” (I-20) and also use “*external resources*” (I-20) for the development (i.e., solving the issue of low *internal IT capabilities*). Moreover, the factor *internal IT capabilities* can have the value high with a positive effect, as i.e., “*the people know a lot*” (I-14) about LCDPs. Interestingly, in these cases, the development is typically done by the IT units and external

developers. Therefore, in this archetype LCDPs are adopted by the IT units to become more efficient and solve the challenge of limited resources for traditional development.

The second archetype is *Application Development Democratisers* with the goal of broadening the developer base for less sophisticated applications through LCDPs. The archetype is defined by the factors *expected efficiency improvements* with the value high and a positive effect and low *sophistication of the to-be-developed application* with a positive effect. In this archetype, *expected efficiency improvements* are a key decision criterion, e.g., “*cost efficiency and speed – those are the two things that really matter*” (I-04). Interestingly, for the *expected efficiency improvements*, not only the initial development cost are considered but also the cost to operate and maintain the application. Here LCDPs were also seen of having a competitive advantage as also non-experts can support the operations: “*This is a topic, where also student workers can support*” (I-16). Moreover, when deciding to adopt an LCDP, the development and subsequent operations and maintenance must be considered. A crucial factor that was stated by multiple informants is the lower *sophistication of the to-be-developed application*. They argued that a lower *sophistication of the to-be-developed application* affected LCDP adoption positively. As I-15 put it: “*[The evaluation] also contains soft factors, like low complexity [of the to-be-developed application].*” The same argument was relevant in case eight: “*Because I can build an app, but I cannot offer support for a big app, right? [...] That is the main factor when deciding to use low code*” (I-04). The overarching goal of this archetype is to broaden the developer base with citizen developers from the business units. Moreover, the IT unit and external developers support the citizen developers i.e., by offering Q&A sessions to answer open questions.

Archetype	IT Resource Shortage Mitigators	Application Development Democratisers	Synergy Realisers
Rationale for the adoption	Efficiently develop applications with limited traditional IT resources.	Broaden developer base for less sophisticated applications.	Efficiently develop applications using already included licenses for LCDPs from existing enterprise systems or already developed components.
Used LCDPs	<ul style="list-style-type: none"> Mendix 	<ul style="list-style-type: none"> Appian Mendix Microsoft Power Platform Oracle APEX OutSystems 	<ul style="list-style-type: none"> Mendix Microsoft Power Platform OutSystems
Typical types of developer	<ul style="list-style-type: none"> External developers IT unit 	<ul style="list-style-type: none"> Business unit External developers IT unit 	<ul style="list-style-type: none"> Business unit External developers IT unit

Table 3: Archetypes of LCDP adoption cases

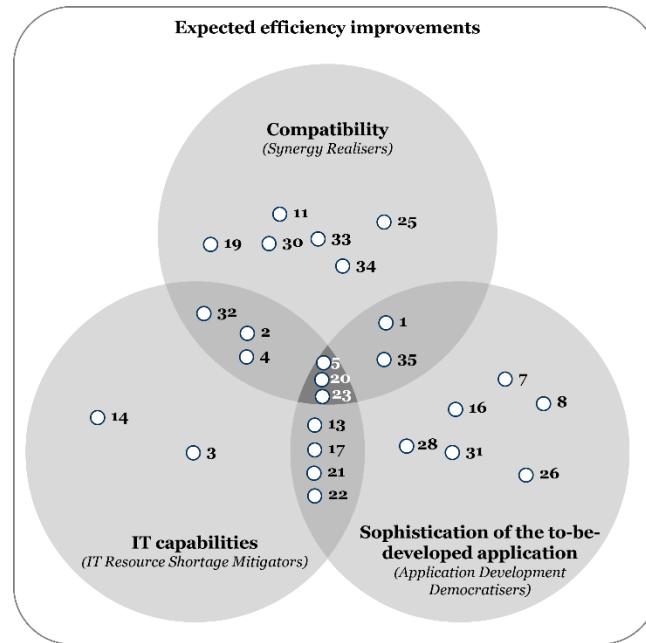


Figure 1: Venn diagram of all LCDP adoption cases

The third archetype is *Synergy Realisers* with the goal to realise efficiencies by using already included licences for LCDPs from existing enterprise systems or already developed components (e.g., authentication modules). Therefore, this archetype is defined by *expected efficiency improvements* with the value high and a positive effect and high *compatibility* with a positive effect. As for the previous two archetypes, the *expected efficiency improvements* are a key reason for LCDP adoption, e.g., “this was cost-driven” (I-14), or “the key reason was the budget” (I-22). Moreover, high *compatibility* with the already existing enterprise systems of the work system and their licences is a key decision factor. For instance, in case 34, the informant pointed out: “The licences environment – it [the LCDP] was already part of licences environment” (I-22). Moreover, as in this archetype the licences are included in already existing enterprise systems, the LCDPs are well integrated in these systems making the development even easier. Therefore, the efficiency in application development is achieved by realising synergies through using LCDP licences from existing enterprise systems or pre-configured components from organisational marketplaces.

Figure 1 shows a Venn diagram of the archetypes, where each circle represents the occurrence of a factor (e.g., *compatibility*). Furthermore, in brackets, the archetype is shown, and the numbers show the case IDs. Based on this visualisation, we can highlight two results. First, all LCDP adoption cases can be sorted into these three archetypes. This indicates that for an LCDP adoption to occur, at least the factor *expected efficiency improvements* and one archetype-specific factor (*compatibility*, *internal IT capabilities*, or *sophistication of the to-be-developed application*) must occur. Besides the factors that define the archetypes, additional factors (e.g., *top management support*) can occur. However, we see these cases as a more specific version of the archetypes, as the adoption already occurs based on the archetype-defining factors.

Second, multiple cases (e.g., case 35) combine two archetypes (*Synergy Realisers* and *Application Development Democratisers*). Therefore, we conclude that archetypes can occur combined.

Except from a high *external pressure* (cases seven and 26), high *security and data privacy concerns* (case 21), and a closed *organisational culture* (cases two, four, and 33), all factors positively affect LCDP adoption. As the output for these cases is nevertheless LCDP adoption, we argue that the positive effect of the other factors must have equalised or even dominated the negative effects. This argument is also supported by our empirical data, e.g., case four, where there “were a few colleagues from the IT, that were against it [the LCDP]” (closed *organisational culture*), but none of them was a “decision maker, who could push against it” (I-02). However, in this case, the decision-makers were “fascinated by the speed” and the “integration [of the LCDP] in the [software name]” (I-02). Therefore, the LCDP adoption decision was taken. The same pattern also occurred in case two, where the “IT has not really shown that much interest in engaging with [the LCDP]” (I-02) due to closed *organisational culture*. However, the high *compatibility* with existing systems, high *expected efficiency improvements*, the lack of IT resources (low *internal IT capabilities*), and the support from *top management* resulted in LCDP adoption.

In summary, first, the overarching goal of all LCDP adoption cases is to realise efficiencies (i.e., *expected efficiency improvements*). Second, there are three archetypes for LCDP adoption (*IT Resource Shortage Mitigators*, *Application Development Democratisers*, and *Synergy Realisers*). Third, all LCDP adoption cases can be allocated to these archetypes. Fourth, the majority of factors have a positive effect for LCDP adoption; however, negative effects (e.g., from a closed *organisational culture*) can be compensated by positive effects of other factors.

III. NON-ADOPTION

Besides the 26 adoption cases, we also captured ten non-adoption cases. For the non-adoption cases, high *privacy and data security concerns*, low *compatibility*, high *sophistication of the to-be-developed application*, high *complexity*, closed *organisational culture*, low *usefulness*, low *internal IT capabilities* and high *external pressure* have a negative effect on LCDP adoption. However, high *compatibility*, high *expected efficiency improvements*, low *internal IT capabilities*, and high *top management support* have a positive effect. A full coding table can be found in Appendix E.

When analysing the underlying reason for the LCDP non-adoption, we found one archetype across all non-adoption cases. The archetype is defined by a too high *sophistication of the to-be-developed application*, which has a negative effect on LCDP adoption. We label this archetype *Intricacy Adversaries* and describe it in Table 4. Multiple informants illustrated the too high *sophistication of the to-be-developed application* resulting in a non-adoption. For instance, in case 15, it was decided not to develop a new front-end for an existing application, as a “*high amount of data*” (I-15) had to be processed (i.e., high *sophistication of the to-be developed application*). In case 29, the too high *sophistication of the to-be-developed application* was also the reason for non-adoption, i.e., “[*the*] *reporting functionality ended up not being at the same level as [in specialised software], and there would simply have been no advantage to using [a LCDP] in these use cases*” (I-21). As multiple LCDPs (Appian, Mendix, Microsoft Power Platform, and Oracle APEX) are found for this archetype, we argue that it is independent of the LCDP. Moreover, it also is independent of the type of developer, as cases with developers from business units, IT units and external developers were found.

Archetype	Intricacy Adversaries
Rationale for the non-adoption	The sophistication of the to-be-developed application is too high for the LCDP.
Considered LCDPs	<ul style="list-style-type: none"> • Appian • Mendix • Microsoft Power Platform • Oracle APEX
Typical types of developer	<ul style="list-style-type: none"> • Business unit • External developers • IT unit

Table 4: Archetype of LCDP non-adoption cases

The *Intricacy Adversaries* archetype can be further specified by adding additional factors. For instance, in case 15, the key decision criterium for non-adoption was a too high *sophistication of the to-be-developed application*. However, the informant further added that the *internal IT capabilities* were too low to develop an application outside the LCDP standard, which he highlighted by stating, “*because we have this skillset to work in the standard [...] So that means what does the [LCDP] framework offer me? I am not just going to use some plugin or JavaScript library*” (I-15). Therefore, due

to the too high (out of standard) *sophistication of the to-be-developed application* and low *internal IT capabilities*, a decision was taken not to adopt the LCDP.

As outlined above, the factors high *compatibility*, high *expected efficiency improvements*, low *internal IT capabilities*, and high *top management support* positively affect the LCDP adoption. However, in combination with a high *sophistication of the to-be-developed application*, the result is non-adoption. In the following sections, we present two cases where the positive factors occurred, but still, the decision was taken not to adopt the LCDP, (i.e., case ten and case 24).

In case ten, *expected efficiency improvements*, and *top management support* were high and positively affected LCDP adoption. Moreover, *internal IT capabilities* were low which also had a positive effect on the LCDP adoption. However, the *sophistication of the to-be-developed application* was high and the *usefulness* was seen as low, both with a negative effect, resulting in the LCDP non-adoption. In this case, the adoption of an LCDP to develop a specialised insurance application was evaluated. There was high *top management support*, as “*low code is modern, and [the LCDP] was the strategic low code tool*” (I-12). Furthermore, in this case, the goal was to realise expected efficiency improvements for an internal project, as “*custom development is done in profit centres and not in cost centres*” (I-12). However, the *sophistication of the to-be-developed application* was too high for the LCDPs, i.e., “*there was no fit between the platform and the requirements in the end*” (I-12). This resulted in LCDP non-adoption.

In case 24, *compatibility* and *expected efficiency improvements* were high and positively affected LCDP adoption. However, the *sophistication of the to-be-developed application (internal IT capabilities)* were high (low), and both had a negative effect on LCDP adoption, which in the end resulted in the LCDP non-adoption. In this case, the objective was to develop a tool for project management; however, “*from the beginning, the demand was fuzzy. It was just the main topic that we want to manage our projects better*” (I-16). At the beginning of the evaluation, the goal was to get “*as fast as possible, a cheap solution [an application]*” (I-16), and as there was a “*strong organisational drive*” (I-16) towards adopting LCDPs, and therefore, an LCDP was the first choice. However, upon further evaluation of the sophistication of the to-be-developed application “*that made no sense at all in the end because it would be extremely difficult to maintain and these functionalities would in any case not be as easy to reproduce as they are implemented in these professional tools*” (I-16). Moreover, the internal IT capabilities were low, i.e., “*I think this was also a skill problem*” (I-16). Therefore, the decision was taken to not adopt an LCDP for this application development project.

As a summary of the previous sections, first, a too high *sophistication of the to-be-developed application* is the core factor hindering an LCDP adoption. Second, this factor is applicable across platforms and types of developers. Third, even if other factors have a positive effect on LCDP adoption (e.g., *expected efficiency improvements* or *top management*

support), the effect of the high *sophistication of the to-be-developed application* is still stronger (cf. case ten).

V. DISCUSSION

A. Overarching

We empirically researched combinations of factors that result in LCDP adoption in work systems with 36 cases. We captured 26 adoption cases and ten non-adoption cases. From these cases, we found three archetypes for LCDP adoption (i.e., *IT Resource Shortage Mitigators*, *Application Development Democratisers*, and *Synergy Realisers*) and one archetype for LCDP non-adoption (i.e., *Intricacy Adversaries*). In the following sections, we first discuss the theoretical implications of our results and, second, derive practical starting points for LCDP adoption.

B. Theoretical Implication

I. COMPARISON OF EFFECTS TO LITERATURE

First, we compare the effects found in our cases with those effects discussed in CC and SDM adoption literature based on our research model published in [12]. As shown in Table 5, for seven factors, we found the same effect as discussed in literature. We did not find three factors that are discussed in the literature in our cases. For two factors, we found the effect partly (i.e., literature found mixed effects, but we found only positive or negative effect). Further, we found two additional factors and for one factor we could extend the effects from literature. The remainder of this section focuses on situations where the empirically found effects differ from the literature.

We did not find three factors which are found to be relevant in CC and SDM adoption literature in our cases: First, *vendor lock-in* is not relevant for adopting LCDPs in work systems. I-03 pointed out that it is essential to consider vendor lock-in on an organisational level but, in a post-adoption situation in a work system, *“this does not play any role”* (I-03). The next factor found by academic literature, but not in our cases, is *training opportunities* [12]. The informants argued that there are comprehensive training materials available (*“Training materials are openly available”* and *“we try to link to [LCDP vendors’] resources, so to offer [LCDP vendor] learning courses”* (I-14)). Therefore, they did not see it as an important decision factor for LCDP (non-)adoption. Another reason why this factor was not found could be that the decision to adopt or not adopt an LCDP is made by people who are already trained. Hence, they do not see this factor as important. Moreover, we could not find the factors *previous experience* in our analysed cases. As the informants did not mention these factors, the factors might have no effect on the LCDP adoption decision.

There are two factors (*Security and data privacy concerns* and *expected working mode improvements*) where we found only parts of the effects previously discussed in the literature. The factor *security and data privacy concerns* was only found empirically in two cases, indicating a lower importance of the factor. In these cases, the effect is also in line with the majority of the literature (i.e., higher concerns have a negative effect on the adoption) [12]. Compared to the literature, only one paper reports a different effect (i.e., high concerns with a positive effect on the adoption [61]). We argue that the paper’s result is not transferable to our findings due to the different contexts of the research.

Factor	Effects on adoption decision found in literature [12]	Comparison to CC and SDM literature
Security and data privacy concerns	Mixed	Effect found partly: High security and data privacy concerns have a negative effect
Compatibility	Positive	Same effect
Vendor lock-in	Negative	Factor not found
Expected efficiency improvements	Positive	Same effect
Sophistication of the to-be-developed application*	n/a	New: Factor is not discussed in literature
Complexity	Negative	Same effect
Previous experience	Positive	Factor not found
Training opportunities	Positive	Factor not found
Organisational culture	Positive	Same effect
Usefulness	Positive	Same effect
Expected working mode improvements	Mixed	Effect found partly: High expected working mode improvements have a positive effect
Top management support	Positive	Same effect
Internal IT capabilities	Positive	Extension to literature: High and low internal IT capabilities have a positive effect
Replacement of shadow IT*	n/a	New: Factor is not discussed in literature
External pressure	Mixed	Same effect

Table 5: Comparison of effect found in literature with case results; (*) additional factor found through open coding

Moreover, we are in line with most papers reporting a positive influence of *expected working mode improvements* on technology adoption. For LCDPs, this refers to the quick, tangible results, as I-21 outlined: "You could really give them something tangible, [...] within one or two weeks, that you could really test. [...]. And that simply helped enormously to accelerate this [...] [development] process." Our finding is unsurprising, as LCDPs are advertised to increase alignment and understanding between business and IT units during application development [11].

We found the two new factors *replacement of shadow IT* and the *sophistication of the to-be-developed application*. The factor *replacement of shadow IT* is not yet discussed in academic CC and SDM adoption literature. We argue that it is not discussed in this literature because these research streams do not focus on replacing shadow IT. Moreover, some practitioners see the adoption of CC as a mechanism to promote shadow IT and not as a way to reduce it, e.g., [67]. However, it is unsurprising that we found this factor in our cases, as practitioners, e.g., [68] and academic research recently discussed this factor, e.g., [11] for LCDPs. Also, the literature on shadow IT considers LCDPs as a technology to support organisations in transforming shadow IT development into a positive or legitimate way of developing applications, also termed as business-managed IT [20]–[22], [69].

Finally, we found the additional factor of *sophistication of the to-be-developed application*, which is not discussed in CC and SDM adoption literature. However, previous research on challenges with LCDPs highlight the limited functionality, flexibility, and customisation potentials [35] of LCDPs, especially for more experienced developers. We argue that these challenges lead to LCDPs being only adopted for less sophisticated applications.

We extended the literature for the factor of *internal IT capabilities*, as we found a new effect, i.e., both high and low *internal IT capabilities* positively influence LCDP adoption. High *internal IT capabilities* with a positive effect on the adoption are already well discussed in the literature, e.g., [41], [56], [70]. However, for LCDPs, we also found that low *internal IT capabilities* positively influence LCDP adoption. I-02 argued that if they did "not have the resources" to develop an application with traditional development methods (i.e., low *internal IT capabilities*), the decision was made to adopt an LCDP and develop the application without traditional IT resources. This result is specific to LCDPs, as they are promoted to reduce entry barriers and required knowledge for application development, thereby empowering citizen developers to develop applications [11] and closing the *internal IT capabilities* gap.

II. ADOPTION AND NON-ADOPTION ARCHETYPES

Generally, we see that combinations of factors lead to the LCDP adoption decision. There are three archetypes for LCDP adoption and one archetype for LCDP non-adoption. All adoption archetypes have high *expected efficiency improvements* with a positive effect in common. Therefore, we see this factor as a necessary factor for LCDP adoption. Both

our informants (e.g., "good IT groups have always tried to adopt the most productive tool-sets they can and a lot of them are very willing to adopt" I-08), but also LCDP vendors (e.g., "accelerate all phases of the app development" [71]) see the *expected efficiency improvements* as a key factor for LCDP adoption. However, as the same factor, with the same value and effect, also occurs in the non-adoption cases, it is not sufficient to explain LCDP adoption. In contrast, in our non-adoption cases, a high *sophistication of the to-be-developed application* with a negative effect only occurs in all non-adoption cases. Therefore, we suggest to see this factor as sufficient for LCDP non-adoption.

For LCDP adoption cases, we found that cases can be part of multiple archetypes (e.g., case two is part of the *IT Resource Shortage Mitigators* and *Synergy Realisers*, cf. Figure 1). Moreover, a negative effect of one factor (e.g., a closed organisational culture, case 33) can be equalised by the other factors' positive effect, thus, resulting in an adoption. In the following sections, we link our archetypes and their goals to previously discussed reasons why LCDPs are adopted.

First, adoption cases belonging to the *IT Resource Shortage Mitigators* archetype aim to efficiently develop applications with limited traditional IT resources. These results corroborate with previous research, e.g., [1], [11] which find that LCDPs are often used to fill the gap of skilled application developers. Also, practitioners highlight that LCDPs are an option to allow lesser skilled developers (e.g., student workers) to develop applications [5]. Therefore, LCDPs must be easy to use and require little time to learn the development. Second, adoption cases belonging to the *Application Development Democratisers* archetype aim to broaden the developer base for less sophisticated applications. For this archetype, the concept of citizen developers (i.e., persons from the business units with little programming experience) is essential, as the citizen developers can now develop less sophisticated applications using the LCDP [5]. The key benefit of citizen developers is that they have a significantly deeper understanding of the overall business process [72], which the to-be-developed application should support. Therefore, this archetype resonates with the marketing statement of LCDPs to enable citizen developers to develop less sophisticated applications [35]. It is also in line with the literature on business-managed IT that promotes the democratisation of IT [20]–[22], [69]. However, governance, compliance, and security guidelines must be established when allowing citizen development, to avoid security breaches [5], [21]. Third, adoption cases belonging to the *Synergy Realisers* archetype aim to efficiently develop applications using already included licenses for LCDPs from existing enterprise systems or already developed components. For Microsoft's Power Platform, the informants highlighted that the LCDP was already part of the existing licences and well-integrated with Microsoft's overall ecosystem. I-18 explained it "Microsoft did a good job, faded in all these [low code] components everywhere and so that you could not get around it." Therefore, citizen developers could easily use the Microsoft Power Platform to develop applications. When

researching the broader enterprise software vendors space, multiple other companies announced Low-Code functionalities in their products in recent years (e.g., SAP [23], ServiceNow [24], Salesforce [25], and Oracle [26]). Finally, the key reason for the non-adoption archetype is the high *sophistication of the to-be-developed application*. From a research perspective, this result can be explained by the lack of functionality, flexibility, and customisation of the LCDPs [35]. Practitioners have the same perception, as they highlight that LCDPs should only be used for simpler applications. They even highlight that the development can be impossible for more sophisticated applications or required a significant amount of traditional coding [5]. However, the high amount of traditional development contradicts the core idea of LCD.

III. LINK TO THEORY

As discussed in section two, we used a combination of the TOE model and the STS theory as the theoretical lenses for this paper. Based on these theoretical lenses, we derived three tentative propositions, which we will elaborate on in the following sections.

The first tentative proposition was that a combination of factors must be considered when researching LCDP adoption in work systems. Our research confirms this proposition, as all our adoption cases required at least two different factors. This finding is also in line with previous combinatorial technology adoption research, which claims that only a combination of factors leads to the adoption, e.g., Artificial Intelligence adoption [62] or blockchain adoption [73].

The second tentative proposition was that the *social sub-system* (i.e., *structure and people sub-systems*) and *technical sub-system* (i.e., *technology and task sub-systems*) must be balanced and jointly optimised [50] for LCDP adoption [12]. For this paper, the proposition can be interpreted as follows: First, in each adoption case, at least one factor from the *social sub-system* and at least one factor from the *technical sub-system* must occur (to allow a balancing of the sub-systems). Second, to optimise the sub-systems, the majority of factors should have a positive effect. When analysing the three archetypes for LCDP adoption, we see that only for the *IT Resource Shortage Mitigators* archetype, at least one factor from the *social sub-system* and one factor from the *technical sub-system* occurred. For the other two archetypes, only factors from the technical sub-system occurred. This finding indicates that the balancing of sub-systems is less relevant when deciding on LCDP adoption. Moreover, the *technical sub-system* seems to be more important, as illustrated by I-15: "You are right – this list is rather technical." The optimisation of sub-systems, however, is core to all LCDP adoption cases, as we found negative effects only in few cases. However, for these cases, the positive effect of the other factors of the respective sub-system outweighed and equalised the negative effect. When applying this theoretical lens to the non-adoption cases, we see imbalanced sub-systems, as only factors from one sub-system occurred in the many non-adoption cases. Moreover, since most factors had a negative effect, the sub-systems are

not optimised. As a summary of the previous section, we can see that the balancing of sub-systems is less important in explaining LCDP adoption (as adoption also occurs with only technical factors). However, optimising sub-systems (i.e., factors have a positive effect on the sub-systems) is more important for LCDP adoption.

Based on the TOE model, the third proposition was that the environment is important for LCDP adoption [12]. We found few cases (i.e., case six, seven, 11, 26, and 29) where external pressure (i.e., an environmental factor) was required for the (non-)adoption decision. However, as this is not the case for all archetypes, we can only confirm this tentative proposition partly. We explain this finding by the post-adoption focus of our research, i.e., the LCDP had already been introduced at an organisational level, and as the primary decision is to adopt the LCDP for a specific project, the environment may be less important. This finding is also in line with the results of [11], who found that environmental inhibitors are rather unimportant for the decision to adopt a LCDP in work systems.

As a summary of the previous sections, the LCDP adoption can be explained by a combination of different factors. As we found cases where an LCDP adoption occurs without balancing the *technical* and *social sub-system*, it does not seem to be necessary. However, in all adoption cases, the sub-systems are optimised, and therefore, we conclude that the optimisation is important for the LCDP adoption decision. Finally, the environmental factors are less important for the LCDP adoption decision, as they only occur in few cases.

C. Practical Implications

Based on our results of the three archetypes for LCDP adoption and one archetype for LCDP non-adoption, we can derive seven practical starting points to foster an LCDP adoption in work systems. These practical implications are shown in Table 6 and are discussed in the next sections.

First, when organisations already have LCDP licences included in existing enterprise systems, they should offer them to a broader developer base (as indicated by the *Synergy Realisers* archetype). Especially the Power Platform seems to be well integrated in Microsoft's (licences) environment and thus, it is easy for professional and citizen developers to start developing applications. However, when offering the existing LCDPs licences to a broader developer base, organisations must have strong governance of the LCDPs to avoid the development of shadow IT [11], [21].

Second, marketplaces where organisation-specific low code components (e.g., authentication modules, user interfaces) can be found and re-used, should be established. I-01 pointed out that they have a marketplace with "pre-configured interfaces, modules for the authentication, all in all, that you can use instantly for the low code development. This is a key benefit for LCDPs, that [...] you can use these modules directly." To ensure compliance with security guidelines and a high quality of the marketplace components, a review process for the component must be in place.

Rationale for the adoption	Practical starting points
Efficiently develop applications with limited traditional IT resources. Broaden developer base for less sophisticated applications. Efficiently develop applications using already included licenses for LCDPs from existing enterprise systems or already developed components.	#1: Offer LCDP licences from existing enterprise systems to a broad developer base.
	#2: Build an internal marketplace for organisation specific pre-configured components (e.g., interfaces modules).
	#3: Ensure a low sophistication of the to-be-developed application.
	#4: Ensure a low complexity of adopted LCDP.
	#5: Adopt the LCDP jointly with business and IT units.
	#6: Create an open organisational culture with LCDP champions.
	#7: Create top management support for LCDP adoption.

Table 6: Practical starting points for LCDP adoption in work systems

Third, as indicated by the *Application Development Democratisers* archetype and the *Intricacy Adversaries* archetype, practitioners must evaluate whether the to-be-developed application is of low sophistication. An in-depth analysis of an application’s requirements must be done for this evaluation. It is crucial to start with this evaluation, as otherwise, the decision to adopt an LCDP could be taken, and the development must be stopped due to unsatisfying results. For instance, I-12 illustrated this by stating: “*There was a project that was stopped [...], it was stopped because the implementation has to be done as in custom development. Large data models were built; things were made persistent, which didn’t have to be persistent, both from the requirements and the low code solutions.*” In this case, the problem was that the *sophistication of the to-be-developed application* was too high for the LCDP, and therefore, the development had to be stopped.

Fourth, primarily when focusing on citizen development, the complexity of the adopted LCDP should be low. This factor should be considered when selecting the LCDP, as the complexity and capability of different LCDPs differ significantly. I-03 pointed out that: “*the [Microsoft] Power Platform is just a bit different. In my opinion, much more easy to use.*”

Fifth, as LCDPs change the application development process, the LCDP adoption must be jointly shaped by business and IT units, as otherwise developers in the IT unit could feel “*threatened. If someone comes along now, like me, and I tell them: Guys, I can do exactly what you’re doing, but I do it ten times faster than you. They automatically felt threatened*” (I-20). Research on critical success factors for business-managed IT finds similar result and proposes to establish a mutual approach (with people from business and IT units) towards business-managed IT adoption [21]. Moreover, multiple informants pointed out that this fear resulted from an unclarity on the capabilities and scenarios of use. I-01 highlighted that by stating that “*The traditional software developers often have an aversion to [the LCDP] [...]. However, there are also some colleagues, really traditional software developers, who at some point slip into low-code development with us or work with us once, and then after, let’s say, a few beginnings, they appreciate it or actually enjoy working with it.*” Therefore, organisations should educate professional and citizen developers on the capabilities and scenarios of use for LCDPs.

Sixth, organisations should strive for an open *organisational culture* with LCDP champions. This starting point is closely interlinked with #5, as an open *organisational culture* also promotes the alignment between business and IT units for LCDP usage. Moreover, LCDP champions (i.e., people who have already successfully developed LCDP applications) can be the first points of contact for citizen developers to answer potential questions and enable them to use LCDPs. I-16 highlighted the benefits of this model: “*I organised a lot of them [Q&A sessions] to real enable the use case owner, with the real problems and app ideas.*” This approach to create LCDP champions who can easily be asked in case of questions, reduces the entry barriers for LCDPs significantly.

Seventh, work systems should get *top management support* to promote the LCDP’s adoption. The top management can help to allocate resources for LCDP development, overcome resistance, and positively influence the organisational culture.

VI. CONCLUSION

In this paper, we captured 36 cases (26 adoption and ten non-adoption) and identified three archetypes for LCDP adoption and one for the non-adoption. The paper has four main contributions that are beneficial for researchers and practitioners.

First, based on our research model published in [12], we empirically confirmed ten out of 13 factors leading to LCDP adoption. In addition to the research model, we found two additional factors (i.e., *the sophistication of the to-be-developed application* and the *replacement of shadow IT*). We showed that a high sophistication of the to-be-developed application is a sufficient factor for LCDP non-adoption. This finding contributes to a better understanding of the adoption of LCDPs and explicitly addresses the calls for empirical research on LCDPs [14] and their adoption [35]. Practitioners evaluating LCDP adoption can use these results to develop decision criteria for the adoption.

Second, we showed that the decision to adopt LCDPs results from a combination of multiple factors and is a multifaceted phenomenon. Based on these combinations, we derived three archetypes for LCDP adoption (i.e., *IT Resource Shortage Mitigators*, *Application Development Democratisers*, and *Synergy Realises*) and one archetype for LCDP non-adoption (i.e., *Intricacy Adversaries*). Each

archetype can be described with a distinct set of factors. Since combinations of these factors are the most important for LCDP adoption, researchers and practitioners should focus on these aspects when analysing LCDP adoption in order to save resources. Moreover, with this paper we have confirmed the potential to further explore this topic with other methodologies (e.g., Qualitative Comparative Analysis).

Third, by combining the TOE model and STS theory, this paper is the first to apply a theoretical lens for LCDP adoption in work systems. By applying these theoretical lenses, we showed that social and technical aspects should be jointly optimised for LCDP adoption.

Fourth, based on the identified LCDP adoption archetypes, we derived seven starting points for practitioners for LCDP adoption. These starting points and the theoretical lenses also highlight that LCDP adoption is not a traditional technology adoption, as the application development approach is entirely transformed. Therefore, technical and social aspects must be considered when making the LCDP adoption decision.

Like all research, this paper has limitations. Firstly, we were only able to identify ten cases of non-adoption, which is relatively few. It is much more difficult to identify cases of non-adoption of LCDPs for a variety of reasons. As several informants explained, work systems are still in the early stages of adopting LCDPs and therefore started with simple, clear adoption cases (i.e., there are significantly fewer non-adoption decisions than adoption decisions). In addition, informants may be biased towards talking about positive outcomes (i.e., adoption) and were therefore reluctant to talk about non-adoption. Second, we did not triangulate our findings within cases (e.g., through source triangulation); instead, we identified archetypes of cases across which we triangulated our findings. The rationale for this is that we wanted to have a greater breadth of cases and explore combinatorial effects, while still keeping the research project operationally manageable. Third, because we have explored LCDP adoption qualitatively through mini cases, we cannot quantify the strength of the effects (e.g., the strength of the positive effect of low sophistication of the application to be developed on LCDP adoption). Therefore, we cannot determine to what extent the positive and negative effects balance each other out.

Based on our findings, we can derive multiple areas for further research. First, as we found that a high *sophistication of the to-be-developed application* is a sufficient factor for the LCDP non-adoption, one could research what drives the *sophistication of the to-be-developed application* for LCDPs. In this vein, one could also apply a task-technology-fit theory; as for the non-adoption cases, there seems to be a misfit between the task (i.e., developing a sophisticated application) and the technology (i.e., the LCDP). Second, researchers could also focus on small and medium (SME) sized organisations, as this study focused on large organisations. SMEs might be more interested in adopting LCDPs, as they usually have fewer developer resources than large organisations. Third, we explicitly decided not to conduct a Qualitative Comparative Analysis in this study as we wanted to use an exploratory

approach and be open to further factors besides the model of [12]. Furthermore, using our research model in [12] with 13 factors would have required 8,192 cases for validation, which is not feasible from a resource perspective. However, as we were able to reduce the number of relevant factors for LCDP adoption significantly, a Qualitative Comparative Analysis could be an interesting research avenue to follow. An example for business-managed IT and Qualitative Comparative Analysis is the publication of [22].

VII. BIBLIOGRAPHY

- [1] S. A. A. Naqvi, M. P. Zimmer, P. Syed, and P. Drews, "Understanding the Socio-Technical Aspects of Low-Code Adoption for Software Development," in *Proc. ECIS*, 2023, pp. 1–17.
- [2] C. Richardson and J. Rymer, "New Development Platforms Emerge for Customer-Facing Applications," 2014. <https://www.forrester.com/report/New-Development-Platforms-Emerge-For-CustomerFacing-Applications/RES113411> (accessed Feb. 22, 2022).
- [3] E. Sahinaslan, O. Sahinaslan, and M. Sabancioglu, "Low-Code Application Platform in Meeting Increasing Software Demands Quickly: SetXRM," in *Proc. AIP Conf.*, 2021, vol. 2334, pp. 1–4. doi: 10.1063/5.0042213.
- [4] A. Streim and L. Erdmann, "Presseinformation - Trotz Krieg und Krisen: In Deutschland fehlen 137.000 IT-Fachkräfte," *Bitkom e.V. - Der Arbeitsmarkt für IT-Fachkräfte*, 2022. <https://www.bitkom.org/Presse/Presseinformation/Deutschland-fehlen-137000-IT-Fachkraefte> (accessed Jul. 20, 2023).
- [5] A. Gärtner and K. Wossidlo, "Löst Low Code das Personalproblem der IT?," *CIO Magazine*, 2023. Accessed: Jul. 20, 2023. [Online]. Available: <https://www.cio.de/a/print/noimages/loest-low-code-das-personalproblem-der-it,3713678>
- [6] M. A. Al Alamin, S. Malakar, G. Uddin, S. Afroz, T. Bin Haider, and A. Iqbal, "An Empirical Study of Developer Discussions on Low-Code Software Development Challenges," in *Proc. IEEE/ACM Int. Conf. on Mining Software Repositories*, 2021, pp. 46–57. doi: <https://doi.org/10.48550/arXiv.2103.11429>.
- [7] A. C. Bock and U. Frank, "Low-Code Platform," *Bus. Inf. Syst. Eng.*, vol. 63, no. 6, pp. 733–740, 2021, doi: 10.1007/s12599-021-00726-8.
- [8] S. S. Bhattacharyya and S. Kumar, "Study of Deployment of 'Low Code No Code' Applications Toward Improving Digitization of Supply Chain Management," *J. Sci. Technol. Policy Manag.*, vol. 14, no. 2, pp. 271–287, 2023, doi: 10.1108/JSTPM-06-2021-0084.
- [9] J. Rymer, "Why you Need to Know about Low-Code, even if you're not Responsible for Software Delivery," 2018. <https://www.forrester.com/blogs/why-you-need-to-know-about-low-code-even-if-youre-not-responsible-for-software-delivery/> (accessed Jul. 20, 2023).
- [10] MarketsandMarkets, "Low-Code Development Platform Market by Component, Application Type, Deployment Type, Organization Size, Industry and Region - Global Forecast to 2025," 2020. [Online]. Available: https://www.reportlinker.com/p05286710/Low-Code-Development-Platform-Market-by-Component-Deployment-Mode-Organization-Size-Vertical-And-Region-Global-Forecast-to.html?utm_source=GNW
- [11] Käss, S., Strahringer, S. and Westner, M. (2023) "Practitioners' Perceptions on the Adoption of Low Code Development Platforms", *IEEE Access*, 11, pp. 29009–29034. doi:10.1109/ACCESS.2023.3258539.
- [12] Käss, S. (2022) "Low Code Development Platform Adoption : A Research Model", in *Proc. ACIS 2022*, pp. 1–8.

- [13] J. Rymer, "The Forrester Wave: Low-Code Development Platforms for AD&D," Cambridge, MA, 2017.
- [14] N. Prinz, C. Rentrop, and M. Huber, "Low-Code Development Platforms: A Literature Review," in *Proc. AMCIS 2021*, 2021, pp. 1–10.
- [15] S. Alter, "Work System Theory: Overview of Core Concepts, Extensions, and Challenges for the Future," *J. Assoc. Inf. Syst.*, vol. 14, no. 2, pp. 72–121, 2013, doi: 10.17705/1jais.00323.
- [16] F. Wulf, M. Westner, and S. Strahringer, "We Have a Platform, but Nobody Builds on it: What Influences Platform-as-a-Service Post-Adoption?," *Int. J. Inf. Syst. Proj. Manag.*, vol. 10, no. 1, pp. 49–70, 2022, doi: 10.12821/ijispm100103.
- [17] I. van de Weerd, I. S. Mangula, and S. Brinkkemper, "Adoption of Software as a Service in Indonesia: Examining the Influence of Organizational Factors," *Inf. Manag.*, vol. 53, no. 7, pp. 915–928, 2016, doi: 10.1016/j.im.2016.05.008.
- [18] M. Manhai Li, C. Peters, M. Poser, K. Eilers, and E. Elshan, "ICT-enabled Job crafting: How Business Unit Developers Use Low-Code Development Platforms to Craft Jobs," in *Proc. ICIS*, 2022, pp. 1–17.
- [19] A. Sahay, D. Di Ruscio, and A. Pierantonio, "Understanding the Role of Model Transformation Compositions in Low-Code Development Platforms," in *Proc. 23rd ACM/IEEE Int. Conf. on Model Driven Eng. Lang. and Syst.*, 2020, pp. 1–5. doi: 10.1145/3417990.3420197.
- [20] S. Klotz, M. Westner, and S. Strahringer, "From Shadow IT to Business-Managed IT and back again: How Responsibility for IT Instances evolve over Time," in *Proc. PACIS*, 2020, pp. 1–14.
- [21] S. Klotz, M. Westner, and S. Strahringer, "Critical Success Factors of Business-Managed IT: It takes two to tango," *Inf. Syst. Manag.*, vol. 29, no. 3, pp. 220–240, 2022, doi: 10.1080/10580530.2021.1938300.
- [22] A. Kopper, M. Westner, and S. Strahringer, "From Shadow IT to Business-managed IT: a Qualitative Comparative Analysis to Determine Configurations for Successful Management of IT by Business Entities," *Inf. Syst. E-bus. Manag.*, vol. 18, no. 2, pp. 209–257, 2020, doi: 10.1007/s10257-020-00472-6.
- [23] M. Weingartner, "SAP stellt einheitliches Low-Code-/No-Code-Angebot vor," 2021. <https://news.sap.com/germany/2021/12/erp-low-code-no-code-loesung-btp-innovation/> (accessed Jul. 20, 2023).
- [24] S. Hamiedha, "Low-Code: Der Baukasten der Digitalisierung," 2022. <https://www.servicenow.com/de/blogs/2022/low-code-baukasten-der-digitalisierung.html> (accessed Jul. 20, 2023).
- [25] Salesforce.com, "Salesforce Announces Lightning Web Components, a Modern JavaScript Programming Model for Building Apps on the Lightning Platform," 2018. <https://www.salesforce.com/news/press-releases/2018/12/13/salesforce-announces-lightning-web-components-a-modern-javascript-programming-model-for-building-apps-on-the-lightning-platform/> (accessed Jul. 20, 2023).
- [26] S. Shmeltzer, "Announcing Oracle Visual Builder Platform," 2020. <https://blogs.oracle.com/developers/post/announcing-oracle-visual-builder-platform> (accessed Jul. 20, 2023).
- [27] J. Cabot, "Positioning of the Low-Code Movement within the Field of Model-Driven Engineering," in *Proc. 23rd ACM/IEEE Int. Conf. Model Driven Eng. Lang. Syst.*, 2020.
- [28] C. Di Sipio, D. Di Ruscio, and P. T. Nguyen, "Democratizing the Development of Recommender Systems by Means of Low-Code Platforms," in *Proc. 23rd ACM/IEEE Int. Conf. on Model Driven Eng. Lang. and Syst.*, 2020, pp. 1–9.
- [29] A. C. Bock and F. Ulrich, "In Search of the Essence of Low-Code: An Exploratory Study of Seven Development Platforms," in *Proc. 23rd ACM/IEEE Int. Conf. on Model Driven Eng. Lang. and Syst.*, 2021, pp. 57–66. doi: 10.1109/MODELS-C53483.2021.00016.
- [30] R. L. Totterdale, "Case Study: The Utilization of Low-Code Development Technology to Support Research Data Collection," *Issues Inf. Syst.*, vol. 19, no. 2, pp. 132–139, 2018, doi: https://doi.org/10.48009/2_iis_2018_132-139.
- [31] P. Vincent *et al.*, "Magic Quadrant for Enterprise Low-Code Application Platforms," 2019. Accessed: Jul. 20, 2023. [Online]. Available: <https://www.gartner.com/en/documents/3956079>
- [32] Mendix, "Gartner Magic Quadrant for Enterprise Low-Code Application Platforms," 2022. <https://www.mendix.com/de/resources/gartner-magic-quadrant-for-low-code-application-platforms/> (accessed Jul. 20, 2023).
- [33] E. Karahanna, D. W. Straub, and N. L. Chervany, "Information Technology Adoption across Time: A Cross-Sectional Comparison of Pre-adoption and Post-Adoption Beliefs," *MIS Q.*, vol. 23, no. 2, pp. 183–213, 1999, doi: 10.2307/249751.
- [34] F. Wulf, "Platform-As-A-Service Post-Adoption by Large-Scale Organizations: A Research Model Design," in *Proc. ECIS 2020*, 2020.
- [35] S. Käss, S. Strahringer, and M. Westner, "Drivers and Inhibitors of Low Code Development Platform Adoption," in *Proc. IEEE Conf. on Business Informatics*, 2022, pp. 196–205. doi: 10.1109/CBI54897.2022.00028.
- [36] Y. Luo, P. Liang, C. Wang, M. Shahin, and J. Zhan, "Characteristics and Challenges of Low-Code Development," in *Proc. 15th ACM / IEEE Int. Symposium on Empirical Software Engineering and Measurement*, 2021, pp. 1–11. doi: 10.1145/3475716.3475782.
- [37] R. Sanchis, Ó. García-Perales, F. Fraile, and R. Poler, "Low Code as Enabler of Digital Transformation in Manufacturing Industry," *Appl. Sci.*, vol. 10, no. 1, pp. 1–17, 2020, doi: <https://doi.org/10.3390/app10010012>.
- [38] R. K. Yin, *Case Study Research: Design and Methods*, 6th ed. Thousand Oaks, London: SAGE, 2018. doi: 10.1111/j.1600-0447.1994.tb05843.x.
- [39] C. L. Hsu and J. C. C. Lin, "Factors Affecting the Adoption of Cloud Services in Enterprises," *Inf. Syst. E-bus. Manag.*, vol. 14, no. 4, pp. 791–822, 2016.
- [40] R. Depietro, E. Wiarda, and M. Fleischer, "The Context for Change: Organization, Technology and Environment," *Process. Technol. Innov.*, vol. 199, pp. 151–175, 1990.
- [41] P. F. Hsu, S. Ray, and Y. Y. Li-Hsieh, "Examining Cloud Computing Adoption, Intention, Pricing Mechanism, and Deployment Model," *Int. J. Inf. Manage.*, vol. 34, no. 4, pp. 474–488, 2014, doi: <https://doi.org/10.1016/j.ijinfomgt.2014.04.006>.
- [42] C. Low, Y. Chen, and M. Wu, "Understanding the Determinants of Cloud Computing Adoption," *Ind. Manag. Data Syst.*, vol. 111, no. 7, pp. 1006–1023, 2011, doi: 10.1108/02635571111161262.
- [43] A. N. Tashkandi and I. M. Al-Jabri, "Cloud Computing Adoption by Higher Education Institutions in Saudi Arabia: an Exploratory Study," *Cluster Comput.*, vol. 18, no. 4, pp. 1527–1537, 2015, doi: 10.1007/s10586-015-0490-4.
- [44] M. J. Pan and W. Y. Jang, "Determinants of the Adoption of Enterprise Resource Planning within the Technology-Organization-Environment Framework: Taiwan's Communications Industry," *J. Comput. Inf. Syst.*, vol. 48, no. 3, pp. 94–102, 2008, doi: 10.1080/08874417.2008.11646025.
- [45] T. Oliveira and M. F. Martins, "Understanding e-Business Adoption Across Industries in European Countries," *Ind. Manag. Data Syst.*, vol. 110, no. 9, pp. 1337–1354, 2010, doi: 10.1108/02635571011087428.
- [46] J. Baker, "The Technology–Organization–Environment Framework," in *Springer*, 2012, pp. 231–245. doi: 10.1007/978-1-4419-6108-2_12.
- [47] G. Friedrich-Baasner, M. Fischer, and A. Winkelmann, "Cloud Computing in SMEs: A Qualitative Approach to Identify and Evaluate Influential Factors," in *Proc. HICSS*, 2018, pp. 4681–4690. doi: 10.24251/HICSS.2018.590.
- [48] K. Zhu and K. L. Kraemer, "Post-adoption Variations in Usage and Value of E-business by Organizations: Cross-Country Evidence from the Retail Industry," *Inf. Syst. Res.*, vol. 16, no. 1, pp. 61–84, 2005, doi: 10.1287/isre.1050.0045.
- [49] M. Sumner and T. Ryan, "The Impact of CASE: Can it Achieve Critical Success Factors?," *J. Syst. Manag.*, vol. 45, no. 6, pp. 1–7, 1994.

- [50] R. P. Bostrom and J. S. Heinen, "MIS Problems and Failures: A Socio-Technical Perspective Part I: The Causes," *MIS Q.*, vol. 1, no. 3, pp. 17–32, 1977, doi: <https://doi.org/10.2307/248710>.
- [51] S. K. Parker, A. Van Den Broeck, and D. Holman, "Work Design Influences: A Synthesis of Multilevel Factors that Affect the Design of Jobs," *Acad. Manag. Ann.*, vol. 11, no. 1, pp. 267–308, 2017, doi: [10.5465/annals.2014.0054](https://doi.org/10.5465/annals.2014.0054).
- [52] F. Schuch, D. Gerster, D. Hein, and A. Benlian, "Implementing Scaled-Agile Frameworks at Non-Digital Born Companies: A Multiple Case Study," in *Proc. HICSS*, 2020, pp. 4336–4345. doi: [10.24251/HICSS.2020.530](https://doi.org/10.24251/HICSS.2020.530).
- [53] A. Loske, T. Widjaja, A. Benlian, and P. Buxmann, "Perceived IT Security Risks in Cloud Adoption: The Role of Perceptual Incongruence Between Users and Providers," in *Proc. ECIS*, 2014, pp. 1-16.
- [54] E. M. Rogers, *Diffusion of Innovations*, 4th ed. Free Press, 2010.
- [55] M. Siepermann, A. Sutaj, P. Lübbecke, and R. Lackes, "Refuse to Walk on Clouds: Why Firms Still do not use Cloud Computing: An Empirical Study of Barriers and Enhancers," in *Proc. HICSS*, 2016, pp. 3969–3978. doi: [10.1109/HICSS.2016.492](https://doi.org/10.1109/HICSS.2016.492).
- [56] F. Wulf, M. Westner, and S. Strahringer, "Cloud Computing Adoption: A Literature Review on What is New and What Still Needs to be Addressed," *Commun. Assoc. Inf. Syst.*, vol. 48, no. 1, pp. 523–561, 2021, doi: [10.17705/1CAIS.04843](https://doi.org/10.17705/1CAIS.04843).
- [57] E. H. Schein, "Organizational Culture," *Am. Psychol.*, vol. 45, no. 2, pp. 109–119, 1990.
- [58] F. K. Y. Chan and J. Y. L. Thong, "Acceptance of Agile Methodologies: A Critical Review and Conceptual Framework," *Decis. Support Syst.*, vol. 46, no. 4, pp. 803–814, 2009, doi: [10.1016/j.dss.2008.11.009](https://doi.org/10.1016/j.dss.2008.11.009).
- [59] M. Malik and S. Orr, "A Configurational Examination of Agile Development as a Sociotechnical System," *Ind. Mark. Manag.*, vol. 104, pp. 325–339, 2022, doi: <https://doi.org/10.1016/j.indmarman.2022.05.003>.
- [60] T. Beranic, P. Rek, and M. Hericko, "Adoption and Usability of Low-Code / No-Code Development Tools," in *Proc. of the Central European Conf. on Information and Intelligent Systems*, 2020, pp. 97–103.
- [61] J. N. Kinuthia, "Technological, Organizational, and Environmental Factors Affecting the Adoption of Cloud Enterprise Resource Planning (ERP) Systems," in *Proc. AMCIS*, 2015, pp. 1–15.
- [62] P. Laut, P. Dumbach, and B. M. Eskofier, "Integration of Artificial Intelligence in the Organizational Adoption: A Configurational Perspective," in *Proc. ICIS*, 2021, pp. 1–17.
- [63] G. Paré, "Investigating Information Systems with Positivist Case Research," *Commun. Assoc. Inf. Syst.*, vol. 13, 2004, doi: [10.17705/1cais.01318](https://doi.org/10.17705/1cais.01318).
- [64] "MAXQDA Official Website," 2023. <https://www.maxqda.com/de/> (accessed Jul. 20, 2023).
- [65] L. Dubé and G. Paré, "Rigor in Information Systems Positivist Case Research: Current Practices, Trends, and Recommendations," *MIS Q.*, vol. 27, no. 4, pp. 597–635, 2003, doi: [10.2307/30036550](https://doi.org/10.2307/30036550).
- [66] M. Gibbert, W. Ruigrok, and B. Wicki, "What Passes as a Rigorous Case Study?," *Strateg. Manag. J.*, vol. 29, no. 13, pp. 1465–1474, 2008, doi: [10.1002/smj.722](https://doi.org/10.1002/smj.722).
- [67] M. Walters, "Four Practical Steps to Eliminate Shadow IT Permanently," 2022. <https://www.forbes.com/sites/forbestechcouncil/2022/02/25/four-practical-steps-to-eliminate-shadow-it-permanently/> (accessed Jul. 20, 2023).
- [68] D. Begonha, A. Kopper, and T. Thirakul, "Low-code/No-code: A Way to Transform Shadow IT into a Next-gen Technology Asset," 2022. <https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/tech-forward/low-code-no-code-a-way-to-transform-shadow-it-into-a-next-gen-technology-asset>
- [69] A. Kopper *et al.*, "Shadow IT and Business-Managed IT: A Conceptual Framework and Empirical Illustration," *Int. J. It/bus. Alignment Gov.*, vol. 9, no. 2, pp. 53–71, 2018, doi: [10.4018/IJITBAG.2018070104](https://doi.org/10.4018/IJITBAG.2018070104).
- [70] E. Loukis, S. Arvanitis, and N. Kyriakou, "An Empirical Investigation of the Effects of Firm Characteristics on the Propensity to Adopt Cloud Computing," *Inf. Syst. E-bus. Manag.*, vol. 15, no. 4, pp. 963–988, 2017, doi: [10.1007/s10257-017-0338-y](https://doi.org/10.1007/s10257-017-0338-y).
- [71] Mendix.com, "Mendix - Low Code Development." <https://www.mendix.com/> (accessed Jul. 20, 2023).
- [72] D. Hoogsteen and H. Borgman, "Empower the Workforce, Empower the Company? Citizen Development Adoption," in *Proc. HICSS*, 2022, pp. 4717–4726. doi: [10.24251/HICSS.2022.575](https://doi.org/10.24251/HICSS.2022.575).
- [73] M. Xia, Z. Xie, H. Lin, and X. He, "Synergistic Mechanism of the High-Quality Development of the Urban Digital Economy from Blockchain Adoption Perspective—A Configuration Approach," *J. Theor. Appl. Electron. Commer. Res.*, vol. 17, no. 2, pp. 704–721, 2022, doi: [10.3390/jtaer17020037](https://doi.org/10.3390/jtaer17020037).

VIII. Appendix

A. Informant Characteristics

ID	Role	Type of informant	Nb. of adoption cases	Nb. of non-adoption cases
I-01	LCDP roll-out champion	Line manager	1	
I-02	LCDP consultant	Consultant	3	
I-03	LCDP owner	Line manager	1	1
I-04	LCDP owner	Line manager	1	1
I-05	Project manager for LCDP	Consultant	Excluded as no case can be found	
I-06	Project manager for LCDP	Consultant	Excluded as no case can be found	
I-07	Project manager for LCDP	Consultant	Excluded as focus was on organisational adoption and not on work system.	
I-08	Project manager for LCDP	Consultant	1	1
I-09	LCDP owner	Line manager	1	
I-10	LCDP owner	Line manager	1	
I-11	Project manager for LCDP	Consultant	2	
I-12	LCDP Owner	Line manager	1	1
I-13	Project manager for LCDP	Consultant	1	1
I-14	LCDP roll-out champion	Line manager	2	
I-15	LCDP owner	Line manager	1	1
I-16	LCDP owner	Line manager	1	1
I-17	LCDP owner	Line manager	1	
I-18	LCDP owner	Line manager	1	1
I-19	Project manager for LCDP	Consultant	1	
I-20	LCDP owner	Line manager	2	
I-21	Project manager for LCDP	Consultant	2	1
I-22	LCDP developer	Line manager	1	
I-23	LCDP owner	Line Manager	1	1
Total			26	10

Table 7: Informant characteristics

B. Semi-structured Questionnaire for LCDP Adoption

Topic	Open questions to ask	Follow up questions (ask if necessary)
Introduction		
Current state of LCDP adoption on an organisational level	<p>To start the interview, would you mind giving me a short introduction about yourself?</p> <p>Line Manager: What is the current status of LCDP adoption in your organisation? Why did you adopt an LCDP in your organisation?</p> <p>Consultants: Which cases of LCDP adoption / non-adoption have you seen? Why was an LCDP adopted in this organisation?</p>	<p>Would you consider your background an IT background or a business background?</p> <p>What is the extent to which LCDPs are used in your organisation?</p>
Process to adoption LCDPs in work systems	<p>Line Managers: In hindsight, could you please describe the process for the adoption/non-adoption for LCDP [x] in work system [y]?</p> <p>Consultants: In hindsight, could you please describe the process for the adoption/non-adoption of a LCDP at case [previously outlined case]?</p>	<p>How did you decide to select LCDP [x]?</p> <p>In which stage of LCDP adoption would you consider the organisation [i.e., adoption vs. post-adoption]?</p> <p>Who are the members of this work system?</p>
Factors for LCDP adoption in work systems	<p>Line Manager: What were the most important factors for adopting / not adopting LCDP [x]?</p> <p>Consultants: At case [y], what were the most important factors for adopting / not adopting LCDP [x]?</p> <p>Multiple follow-up questions to the factors outlined, incl. how would you evaluate the effect of the [outlined factors]?</p>	<p>Technology factors</p> <p>Which technology related factors did you consider for the LCDP adoption / non-adoption?</p> <p>How would you evaluate the effect of [outlined factors] on LCDP adoption / non-adoption?</p> <p>Organisation factors</p> <p>Which organisation related factors did you consider for the LCDP adoption / non-adoption?</p> <p>How would you evaluate the effect of these factors on the LCDP adoption / non-adoption?</p> <p>Environment factors</p> <p>Which environmental factors did you consider for the LCDP adoption/non-adoption?</p> <p>How would you evaluate the effect of these factors on the LCDP adoption / non-adoption?</p>
Key learnings for LCDP adoption in work systems	<p>What are your key learnings you would recommend other organisations when adopting LCDPs?</p>	

Topic	Open questions to ask	Follow up questions (ask if necessary)
Closing remarks		

Table 8: Semi-structured interview guideline

C. Case Description

ID	Description	LCDP	Outcome	Industry	Informant	Archetype				
						IT Resource Shortage Mitigators	Application Development Democratisers	Synergy Realisers	Intricacy Adversaries	
1	Development of an HR low code application	Mendix	Adoption	Conglomerate (Multiple Industries)	I-01		X	X		
2	Development of an application to digitalise the process for occupational disability	Mendix	Adoption	Financial Services	I-02	X		X		
3	Development of a booking and invoice system	Mendix	Adoption	Aircraft Renting	I-02	X				
4	Development of an inventory tracking for tools	Mendix	Adoption	Financial Services	I-02	X		X		
5	Development of an application for construction work surveillance	Microsoft Power Platform	Adoption	Transportation	I-03	X	X	X		
6	Non-development of an application for time tracking	Microsoft Power Platform	Non-adoption	Transportation	I-03				X	
7	Development of a low code application for the automatic storage of mails	Appian	Adoption	Financial Services	I-12		X			
8	Development of an idea management tool for an organisation	Mendix	Adoption	Industrial Manufacturing (e.g., Automotive, Aerospace)	I-04		X			
9	Non-development of an application to track tools for laboratories	Mendix	Non-adoption	Industrial Manufacturing (e.g., Automotive, Aerospace)	I-04				X	
10	Non-development of a specialised insurance application	Appian	Non-adoption	Financial Services	I-12				X	
11	Development of a business efficiency application in SharePoint	Microsoft Power Platform	Adoption	Government and Public Services	I-08			X		
12	Non-development of an applications for Sharepoint	Microsoft Power Platform	Non-adoption	Government and Public Services	I-08				X	
13	Development of an application for a project database in Excel	Mendix	Adoption	Industrial Manufacturing	I-09	X	X			
14	Development of an HR tool to track the exit interview	Mendix	Adoption	Industrial Manufacturing	I-10	X				
15	Non-development of a low code front end for an existing application	APEX	Non-adoption	Mobility	I-15				X	
16	Development of an application to apply for a donation from the organisation	APEX	Adoption	Mobility	I-15		X			

ID	Description	LCDP	Outcome	Industry	Informant	Archetype			
						IT Resource Shortage Mitigators	Application Development Democratisers	Synergy Realisers	Intricacy Adversaries
17	Development of an internal COVID-19 tracking application	OutSystems	Adoption	Healthcare and Life Sciences	I-13	X	X		
18	Non-development of an application to control projects	Mendix	Non-adoption	Consulting	I-13				X
19	Development of an application to track the development of a gas turbine	Mendix	Adoption	Industrial Manufacturing	I-14			X	
20	Development of a competence management application for organisation	Mendix	Adoption	Industrial Manufacturing	I-14	X	X	X	
21	Development of an application to track construction work	RCP	Adoption	Technology, Media, and Telecommunications	I-11	X	X		
22	Development of an application for vehicle registration	Multiple	Adoption	Financial Services	I-11	X	X		
23	Development of an application to consolidate employee benefits in different countries	Microsoft Power Platform	Adoption	Healthcare and Life Sciences	I-16	X	X	X	
24	Non-development of an application for project management	Microsoft Power Platform	Non-adoption	Healthcare and Life Sciences	I-16				X
25	Development of an application for process improvements in the legal department	Microsoft Power Platform	Adoption	Industrial Manufacturing	I-17			X	
26	Development of an application for the IT project portfolio management	Microsoft Power Platform	Adoption	Retail and Consumer Goods	I-18		X		
27	Non-development of an application to document machine maintenance	Microsoft Power Platform	Non-adoption	Retail and Consumer Goods	I-18				X
28	Development of an application to streamline the procurement process	Mendix	Adoption	Conglomerate (Multiple Industries)	I-21		X		
29	Non-development of reporting dashboards	Mendix	Non-adoption	Conglomerate (Multiple Industries)	I-21				X
30	Development of a unified dashboard for sales representatives	Mendix	Adoption	Conglomerate (Multiple Industries)	I-21			X	
31	Development of an application for workflow management of laboratory orders	OutSystems	Adoption	#N/A	I-19		X		
32	Development of an B2C applications for patients	OutSystems	Adoption	Healthcare and Life Sciences	I-20	X		X	
33	Development of an integrated customer experience across multiple sub-systems	OutSystems	Adoption	Healthcare and Life Sciences	I-20			X	

ID	Description	LCDP	Outcome	Industry	Informant	Archetype			
						IT Resource Shortage Mitigators	Application Development Democratisers	Synergy Realisers	Intricacy Adversaries
34	Development of an application to track machine maintenance	OutSystems	Adoption	Chemicals	I-22			X	
35	Development of an application for engineering change management process	Appian	Adoption	Industrial Manufacturing	I-23		X	X	
36	Non-development of an application to visualise large amounts of data	Appian	Non-adoption	Industrial Manufacturing	I-23				X

Table 9: Case descriptions (“X” grouping to archetype) – Cases collected from February to April 2023.

D. Overview of Context Variables

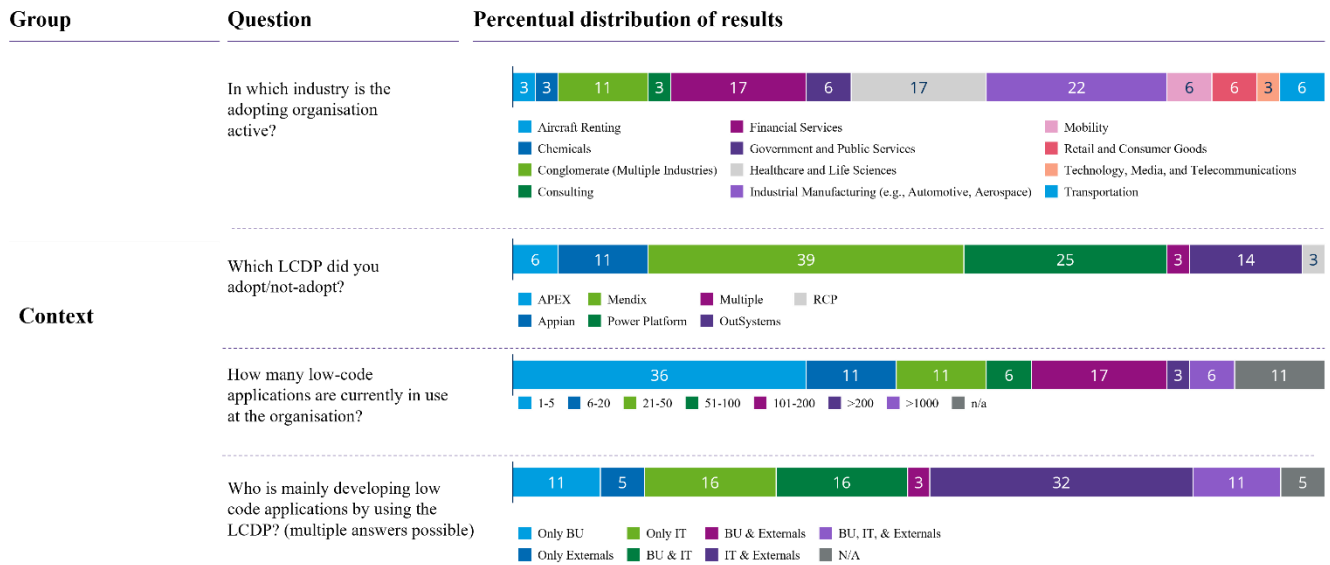
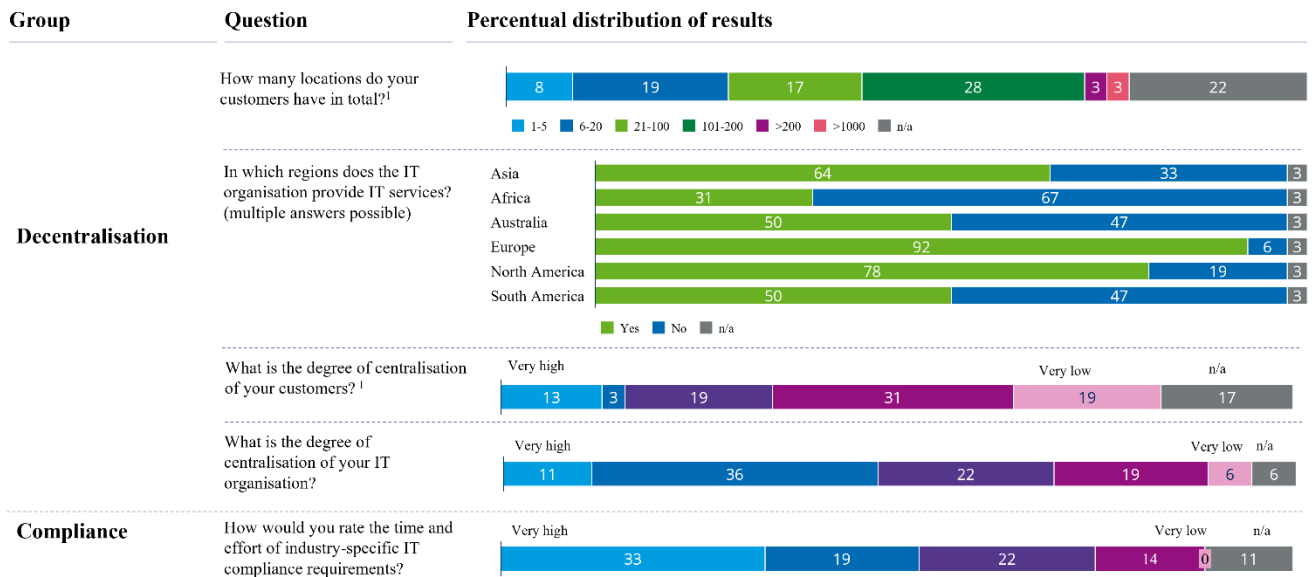


Figure 2: Overall context information of the cases



1. "Customers" = customers of the IT organisation - i.e., all organisational units (e.g., companies and divisions) that "purchase" IT services from the IT organisation and/or obtain them as users.

Figure 3: Decentralisation and compliance information

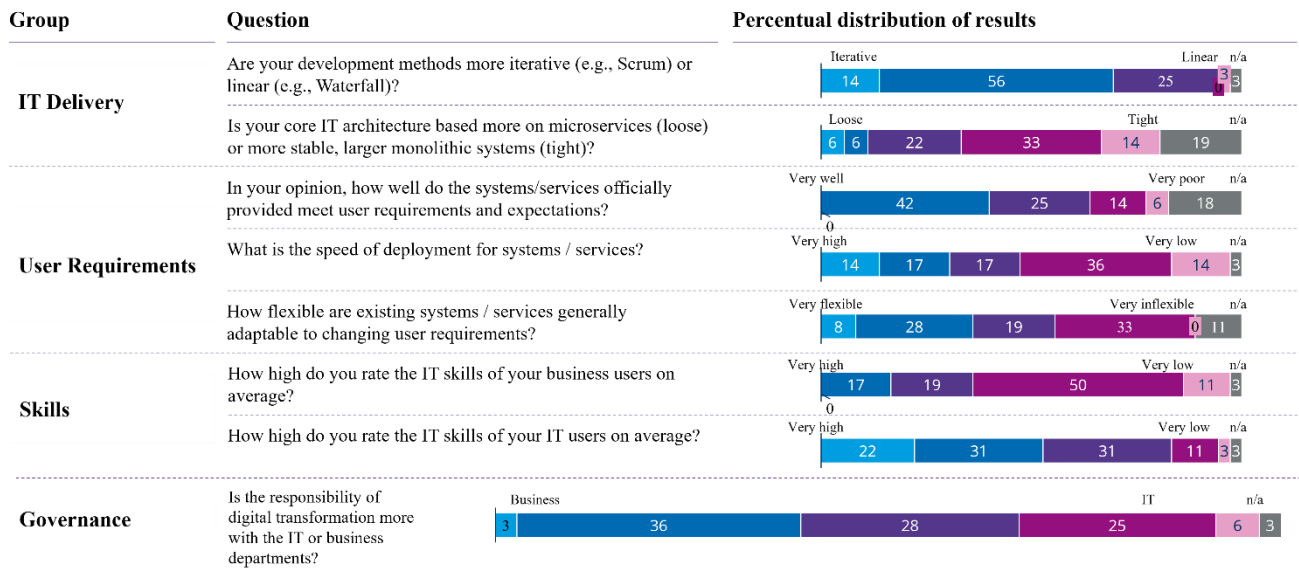


Figure 4: IT-delivery, user requirements, skills, and governance information

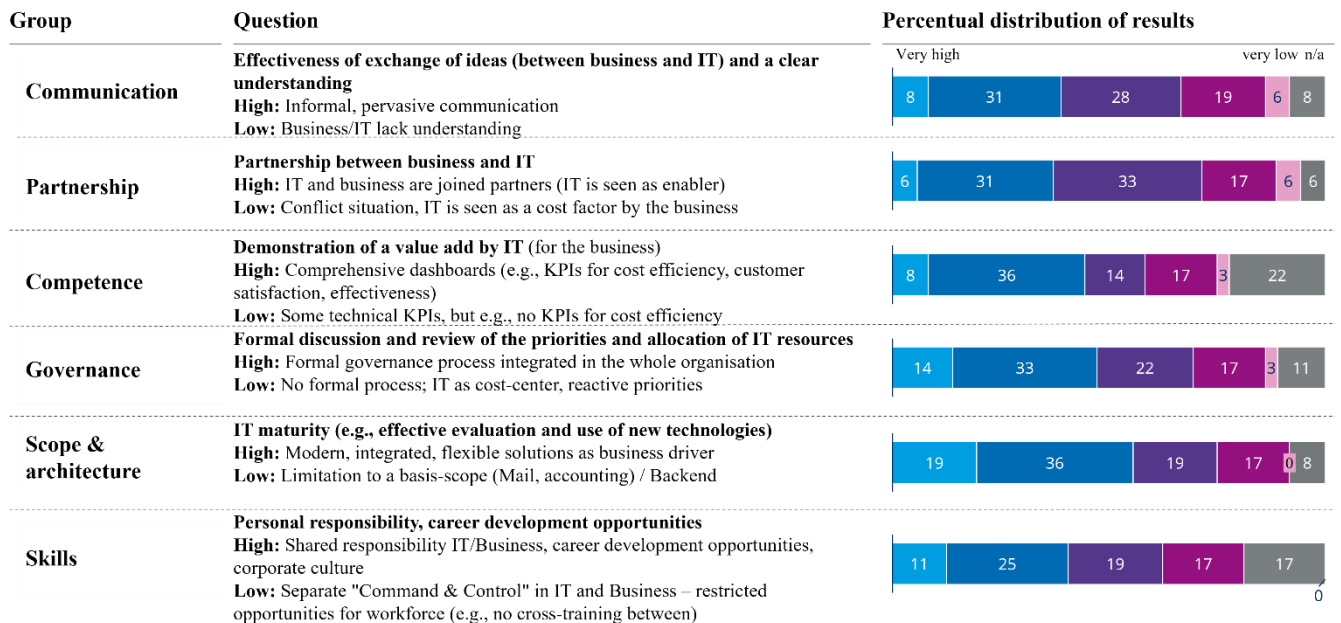


Figure 5: Communication, partnership, competence, governance, scope & architecture, and skills

E. Coding Table

ID	Outcome	Security and data privacy concerns		Compa-tibility		Expected ef-ficiency improve-ments		Sophisti-cation of to be devel-oped app-lication		Compl-exity		Organi-sational culture		Useful-ness		Expected working mode improve-ments		Top-manage-ment support		Internal IT capa-bilities		Repla-cement of shadow IT		External pressure	
		Value	Effect	Value	Effect	Value	Effect	Value	Effect	Value	Effect	Value	Effect	Value	Effect	Value	Effect	Value	Effect	Value	Effect	Value	Effect	Value	Effect
1	Adoption			H (+)	H (+)	L (+)																			
2	Adoption			H (+)	H (+)							C (-)						H (+)	L (+)						
3	Adoption				H (+)											H (+)			L (+)						
4	Adoption			H (+)	H (+)							C (-)						H (+)	L (+)						
5	Adoption			H (+)	H (+)	L (+)										H (+)			H (+)	H (+)					
6	Non-adoption	H (-)	L (-)				H (-)												L (-)				H (-)		
7	Adoption				H (+)	L (+)												H (+)					H (-)		
8	Adoption				H (+)	L (+)																			
9	Non-adoption				H (+)	H (-)	H (-)												L (-)						
10	Non-adoption				H (+)	H (-)							L (-)					H (+)	L (+)						
11	Adoption			H (+)	H (+)							O (+)						H (+)					H (+)		
12	Non-adoption			L (-)	H (+)	H (-)						C (-)						H (+)	L (-)						
13	Adoption				H (+)	L (+)													L (+)						
14	Adoption				H (+)					L (+)	O (+)								L (+)						
15	Non-adoption						H (-)												L (-)						
16	Adoption				H (+)	L (+)										H (+)									
17	Adoption				H (+)	L (+)						O (+)							L (+)						
18	Non-adoption			L (-)	H (+)	H (-)											H (+)								
19	Adoption			H (+)	H (+)																		H (+)		
20	Adoption			H (+)	H (+)	L (+)													H (+)						
21	Adoption	H (-)			H (+)	L (+)													L (+)						
22	Adoption				H (+)	L (+)													L (+)						
23	Adoption			H (+)	H (+)	L (+)													H (+)						
24	Non-adoption			H (+)	H (+)	H (-)													L (-)						
25	Adoption			H (+)	H (+)					L (+)						H (+)	H (+)								
26	Adoption				H (+)	L (+)	L (+)											H (+)					H (-)		
27	Non-adoption			L (-)	H (+)	H (-)																			
28	Adoption				H (+)	L (+)						O (+)						H (+)							
29	Non-adoption						H (-)																	H (-)	
30	Adoption			H (+)	H (+)													H (+)							
31	Adoption				H (+)	L (+)	L (+)																H (+)		
32	Adoption			H (+)	H (+)													H (+)	H (+)	L (+)					
33	Adoption			H (+)	H (+)					L (+)	C (-)							H (+)							
34	Adoption			H (+)	H (+)																				
35	Adoption			H (+)	H (+)	L (+)																			
36	Non-adoption			L (-)	H (+)	H (-)																			
Sum		2	2	20	20	33	25	25	6	6	8	8	1	1	9	9	9	9	18	18	3	3	5	5	

Table 10: Coding of all 36 LCDP (non-)adoption cases

H = High
L = Low
C= Closed
O = Open
(+) = Positive
(-) = Negative



Sebastian Käss received a BSc. and MSc. in industrial engineering and management from the Karlsruhe Institute of Technology, Karlsruhe, Germany.

He is currently a PhD candidate at TU Dresden and OTH Regensburg, with a research focus on Low Code Development Platforms. Before starting in academia, he worked as a management consultant at the largest strategy consultancy.

Mr. Käss already has several Conference publications at ACIS and CBI, and a journal publication in IEEE Access, and is currently writing his PhD thesis.



Susanne Strahinger received a Diploma in business informatics from Darmstadt University of Technology, Darmstadt, Germany where she also obtained her PhD and completed her habilitation thesis.

She is a Professor of Business Information Systems, especially IS in Manufacturing and Commerce, at TU Dresden, Dresden, Germany. Before joining TU Dresden, she held positions at the University of Augsburg and the European

Business School.

Prof. Dr. Strahinger has published in Information & Management, Journal of Information Technology Theory and Application, Information Systems Management, and Journal of Information Systems Education, IEEE Access, among others.



Markus Westner received a business administration and management diploma from the European Business School, Oestrich-Winkel, Germany. Further, he received a Master of Computing from Unitec, Auckland, New Zealand.

He is a Professor of IT Management at OTH Regensburg, Regensburg, Germany. Before he started his academic career, he worked as a management consultant in a project manager position for one of the world's largest management consultancies.

Prof. Dr. Westner is the author of several journal articles and conference papers. His work focuses on IT strategy and IT sourcing. He acts as an Associate Editor for Information & Management.