

Article

The Relation of Shadow Systems and ERP Systems—Insights from a Multiple-Case Study

Melanie Huber ^{1,*}, Stephan Zimmermann ¹, Christopher Rentrop ¹ and Carsten Felden ²

¹ Institute for Process Control (kips), Konstanz University of Applied Sciences, Brauneggerstr. 55, Konstanz D-78462, Germany; stephan.zimmermann@htwg-konstanz.de (S.Z.); christopher.rentrop@htwg-konstanz.de (C.R.)

² Department of Information Systems, TU Bergakademie Freiberg, Silbermannstr. 2, Freiberg D-09599, Germany; carsten.felden@bwl.tu-freiberg.de

* Correspondence: melanie.huber@htwg-konstanz.de; Tel.: +49-7531-206-328

Academic Editor: Donald Kerr

Received: 15 December 2015; Accepted: 25 January 2016; Published: 29 January 2016

Abstract: ERP systems integrate a major part of all business processes and organizations include them in their IT service management. Besides these formal systems, there are additional systems that are rather stand-alone and not included in the IT management tasks. These so-called ‘shadow systems’ also support business processes but hinder a high enterprise integration. Shadow systems appear during their explicit detection or during software maintenance projects such as enhancements or release changes of enterprise systems. Organizations then have to decide if and to what extent they integrate the identified shadow systems into their ERP systems. For this decision, organizations have to compare the capabilities of each identified shadow system with their ERP systems. Based on multiple-case studies, we provide a dependency approach to enable their comparison. We derive categories for different stages of the dependency and base insights into integration possibilities on these stages. Our results show that 64% of the shadow systems in our case studies are related to ERP systems. This means that they share parts or all of their data and/or functionality with the ERP system. Our research contributes to the field of integration as well as to the discussion about shadow systems.

Keywords: shadow systems; shadow IT; ERP systems; enterprise systems; relation; dependency; integration

1. Introduction

Two-thirds of IT managers indicated in surveys that so-called shadow systems support business processes in their organizations [1,2]. Shadow systems are a result of the case that business workgroups implement additional systems, which exist outside of an organizational IT service management [3] and are often not integrated with other enterprise systems [4,5]. Shadow systems support daily activities sometimes even in the same processes as Enterprise Resource Planning (ERP) systems [6–8]. Legacy systems that were replaced by ERP systems and are still used by business workgroups also relate to this phenomenon [9,10]. ERP systems support a major part of the organization, such as the departments of finance, human resources, operations and logistics, sales, and marketing [11]. ERP systems and other formal enterprise systems [12] thereby offer a comprehensive functionality to support business processes. The goal of this IT support is a high enterprise integration [13]. However, there is evidence that shadow systems lead to redundancy [7,14,15] and a lack of integration in the enterprise architecture [4,5,16,17]. While one of the main goals of Enterprise Architecture Management (EAM) is to support these integration and standardization requirements of an organization [18,19], existing shadow systems preclude this goal [20]. Organizations can identify shadow systems explicitly [21].

However, shadow systems can also appear during software maintenance projects, such as release changes, enhancements of ERP systems [22]. Organizations then have to decide about their integration. To be able to decide about integration, detecting redundancies and comparing the different systems seems to be appropriate [23]. To proceed in this decision, the pursued objective of our research is to explore the different forms of the relation between shadow systems and ERP systems.

Although some contributions about shadow systems mention the topic of integration [4,21], there has been no detailed conjunction of the two research fields in literature yet. However, integration becomes more and more important for organizations, especially in the context of shadow systems [24]. Furthermore, prior research describes integration as under-researched in general [25]. Literature describes integration as an ongoing task that is not only relevant during the initial stages of a system [26,27]. However, existing research mostly focuses on problems with shadow systems during the implementation of an ERP system [28–30]. There are only few studies on shadow systems and established ERP systems [30,31]. These identify challenges and opportunities of shadow systems [30], explore the effect of shadow systems on organizational control [31], but do not analyze how a shadow system relates to an ERP system in general. Furthermore, most ERP researchers focus on single instances of shadow systems [9,29]. There are no studies with a broader empirical basis consisting of a variety of shadow systems from different organizations and processes and their relation to the ERP system. To close these gaps, the scope of our study is to give an insight into the existence of shadow systems in conjunction with established ERP systems. Our applied method is a multiple-case study throughout different industries and processes. Following the analysis of the shadow systems that appeared in these cases, we provide a description of various stages of their dependency on ERP systems. We can use these dependency stages to derive integration recommendations. The study contributes to research on management of single shadow systems with regard to their integration into existing ERP systems. This supports the goal of ERP systems of a high enterprise integration and integration procedures in general.

This paper proceeds as follows: first, we describe the background of integrating shadow systems into ERP systems. This consists of a literature review on commonalities and differences of ERP and shadow systems, followed by details about the similarity of single IT systems in terms of integration. Then we sum up this background to clarify the research problem and pose our research question. Afterwards, we present the research method following a multiple-case study. Section 4 explains and discusses our analysis results. The final part summarizes the findings of the paper, provides practical and theoretical implications, and gives an outlook on future research.

2. The Challenge of Integrating Shadow Systems into ERP Systems

Exploring the relation between a shadow system and its corresponding ERP system, we review prior research and provide an insight into the differences and commonalities of shadow systems and ERP systems in general. Afterwards, we introduce dependency levels discussed in literature based on the similarity of single IT systems in the context of integration. This background builds the basis for the derivation of our research question.

2.1. Literature on ERP Systems and Shadow Systems

The purpose of ERP systems is a high enterprise integration [13]. They achieve this by integrating discrete applications using one common database [32]. The integration includes all information in a company and business fields such as finance, accounting, human resources, operations, and logistics [11]. ERP systems are therefore highly centralized. Especially in the context of their implementation, research has shown that shadow systems may occur [33] and has started to focus on their connection to ERP systems (To identify existing peer-reviewed studies and illustrate the status quo of the research field [33] we at first identified relevant keywords. Those were “shadow system(s)”, “shadow IT”, “feral system(s)”, “grey IT”, “hidden IT”, “rogue IT”, “end user computing”, and “EUC” as they are all used as terms to describe the phenomenon of shadow systems. We used these words

in combination with the terms “ERP” and “ERP system(s)”. After the identification of the keywords, we queried several scientific databases that provide access to peer-reviewed journals and conference papers. We limited our search to the keywords, abstract, and the title of the publication and used the following databases: Ebscohost Business Source Complete, ScienceDirect, IEEE Xplore, ACM Digital Library, and AIS Electronic Library. After analyzing the resulting papers, removing duplicates, and identifying relevant papers, 19 papers remained. Afterwards, we conducted a forward—as well as a backward—search to be able to find additional relevant publications [34]). These studies illustrate that both shadow systems and ERP systems support business processes. However, shadow systems “are deployed autonomously within business departments and by IT users” [3]. Shadow systems are decentralized solutions with a low enterprise integration such as a locally installed application, a spreadsheet, database solution, cloud service, but also an end or peripheral device, a combined solution [16,20] or a legacy system that is no longer part of the IT service management [9,20].

Further differences exist between ERP and shadow systems. IT departments professionally manage and surveil ERP systems and their whole lifecycle [12]. They incorporate them into IT controlling and EAM tasks. In contrast, organizational IT service management does not include shadow systems [3]. Therefore, business departments develop, implement, and maintain shadow systems less professionally. This can lead to false decisions and can disrupt business processes [29].

Supporting a variety of different functions, ERP systems are much more complex than single applications [32]. Shadow systems are often solutions that business departments develop quickly to fulfill an emergent business need [6,16]. At their implementation, they are often easy solutions. In some cases, shadow systems can also be complex due to their initial design or become complex during their life cycle [29].

Due to the complexity and the vast integration, the implementation of an ERP system comes with certain challenges. Either the system or the organization have to change to match each other [35]. Often, the organization puts the system first and then has to adapt its processes to the ERP processes [36]. This is also due to the complexity of extending or changing such a system. In some cases, changes need only slight effort, for example, configuration or screen masks. However, most require moderate to heavy change effort, like workflow or ERP programming, code modifications, or interface development [32]. Shadow systems in contrast provide efficient tools for the daily work. Users know their business needs very well and this knowledge shapes the shadow systems [3,6]. When implemented, they therefore fit the existing processes. Additionally, shadow systems are very flexible and can be adapted to changing requirements easily. Business departments can perform these changes quickly [3,30].

As a result, shadow systems and ERP systems have some attributes in common. For example, they both support business activities in general [3,37] and sometimes even in the same processes [4,6,7]. However, several differences exist, which oppose the goal of ERP systems to achieve a high enterprise integration [32]. Their use in the same processes favors their integration. Therefore, we need a basis to enable integration recommendations for these two system types. To build that basis, we present prior research about the comparison of single IT systems by determining their dependency in the following.

2.2. The Dependency of IT Systems in Context of Integration

As a prerequisite for deciding about an integration of two systems, organizations have to understand the structure and capabilities of the systems they want to integrate. A common word to describe the relation of two systems before their integration in research is “dependency” [23,38,39]. To be able to determine the dependency of two systems, we have to compare their components.

For comparison, the concept of similarity of Lin (1998) is suitable as it consists of commonalities and differences. Lin defines that the more commonalities two systems have, the more similar they are. The more differences they have, the less similar they are. Two systems will be identical, if the similarity between them reaches its maximum [40]. We therefore derive that similarity is not a concept that only has two specifications, such as identical and different. Rather, it can take different states. Thus, the determination of similarity is not a binary decision but a continuum of various stages.

To determine the components of an IT system and to reach our goal of giving propositions about the integration, we focus on integration models that refer to the structure of systems. They help us to define how to integrate the applications. Researchers propose different perspectives on the dimensions of these models. A search for the latest review in the field of integration resulted in a literature review from 2012 [25]. In their study, Chovanetz et al. describe common levels of integration. These are, especially in the field of the integration of applications, data integration, and functional integration [25] (with reference to [41–46]). The goal of data integration is the unique existence of data in an organization [47]. Thereby, different applications share the same data [23,47]. Functional integration refers to the implementation of a functionality only once in an application combined with the opportunity to be usable in other applications, too [23]. A consolidation of similar functions enables this form of integration [46].

Following this concept, we use the data and functional level in our study to compare the systems and to determine their dependency. If a system shares all of its data with the other system [47] and has the same or similar functions [46], we will consider them as dependent. If the two systems share their data via an interface, some form of data integration already exists [48]. If they share no data and do not have similar or identical functions, we will consider them as independent and therefore not integrated. Following our argumentation about the similarity above [40], we argue that systems can also take various stages in between dependent and independent and therefore in between integrated and not integrated. We label these situations grey zones.

To upshot, shadow systems and ERP systems often occur in the same business processes, but also have certain differences. Additionally, we provided an insight into how we can compare IT systems in terms of their integration. Both lines of argumentation lead to the problem statement of our study and the related research question.

2.3. Problem Statement

As described above, the goal of the implementation of ERP systems is a high integration throughout a majority of the business processes in an organization [32]. Shadow systems instead are rather stand-alone and not or only loosely integrated with other enterprise systems. However, business departments often use them in the same processes [4,6,30]. In these cases, shadow systems either supplement or substitute the existing ERP system. They reveal that ERP implementations struggle with the goal of a high integration [6,20,29]. Additionally, these shadow systems hinder the goal of EAM to support integration requirements of an organization [18,19]. Therefore, organizations face the challenge of managing them accordingly and deciding about whether or not to integrate them into the ERP system. To be able to decide about the integration, organizations have to compare the commonalities and differences of the structure of shadow systems and their related ERP systems. A possible way to conduct this comparison is the determination of their dependency [23].

Scientific literature offers few studies on the topic of dependency of shadow systems and ERP systems. In general, there is little research on shadow systems and ERP systems combined. In some studies, the focus is only on one example in one case study [29,30]. Others mention several, but focus on the implications for ERP implementation [9,28,33,49–51] or organizational control [31]. No studies exist that explicitly explore a variety of different shadow system instances in various organizations and processes in the context of later stages of the lifecycle of an ERP system. Another focus of existing research is set on the reasons for the emergence of shadow systems. While some studies provide assumptions on a high level [31,52–55], others go more into detail [7,16,22,30,33,49,50,56,57]. This indicates that literature indeed is clear about a dependency between shadow systems and ERP systems. However, these studies concentrate on the phenomenon of shadow systems. There is no elaborated analysis of shadow systems and ERP systems, or how and to which extent they are dependent in existing literature.

Only, if we examine the dependency between single shadow systems and matured ERP systems in practice, we will be able to derive recommendations for their integration. To provide generalizable

results, it is necessary to analyze various shadow systems from different organizations in different industries. Regarding the described problem and the lack of research in this specific field, we formulate the following research question:

What dependencies of shadow systems and ERP systems can we find in practice?

3. Research Method

To be able to answer the research question, we passively observed the phenomenon of shadow systems in organizations [58]. This accounts for the scientific method of the case study [59]. This method will be useful, if knowledge about the influencing factors is rather low [60]. Research has not discovered the variables of integration as well as the general phenomenon of shadow systems extensively yet, as presented in the prior section. Therefore, this method seems appropriate. Multiple case studies are useful, as they can provide an in-depth description of a phenomenon as well as the ability to generalize [61,62]. The use of various cases allows researchers to replicate the findings in several different situations to ensure their validity [62]. Additionally, multiple-cases will be useful, if the intent of a study is the exploration of a topic [60]. As the goal of this study is to explore the relation between ERP systems and shadow systems, we considered not only one, but also several cases in the research process. This accounts for the method of the multiple-case study [59] that additionally allows broadening the empirical basis.

To achieve a theoretical replication [59,63], we chose three different organizations based in Germany and Switzerland. The organizations in our cases act in various industries, which reach from the service sector to the industry sector. Additionally, processes and business units varied. Table 1 shows an overview about these organizations, their processes and departments, as well as the interview partners from business and IT.

Table 1. Details on the multiple-case study.

	Case A	Case B	Case C
Industry	Insurance	Engineering	Electronics
No. of Employees	1300	47,500	5500
Analyzed Processes/Departments	Benefits Statements: Private and Corporate Clients	Order Management of a Manufacturing Unit (Sales to Shipment)	Corporate Marketing
Interview Partners in Business	2	5	3
IT	1	5	1

We conducted the case studies between June 2012 and June 2013, lasting three to four months each. To be able to get an overview over the complete IT infrastructure in the units, we identified all process-related IT solutions including shadow systems. This was done using different methods to collect information [60,63]. Different methods for data collection ensure that researchers can collect a variety of different data and can capture the complexity of the studied phenomenon [60]. Data sources were internal documents (process models, manuals), technical artefacts (software inventory tools, help desk reports), and interviews with staff from business and IT [59].

For the data collection and after reviewing the existing documents, we used interviews following a pre-tested, semi-structured guideline [64]. This interview guideline helped us to structure the interview, but also to ensure that we received all the relevant information. During the interview, we at first clarified the goal as well as our procedure. Afterwards, we formulated semi-structured questions about the purpose and detailed functionality of the systems, the underlying technology, the data, and the data sources. The interviews lasted one to two hours each. Interview partners were both from the business and the IT side. Participants from the IT department gave an insight into the existing enterprise system in their organization and its functionality. Participants from the business departments gave a detailed description of the existing shadow systems, their functionality, and the relation to the formal enterprise system. We took notes to collect the information that the interviewees presented [63].

We analyzed each of the found shadow systems individually on the dependency on the ERP system in the companies. Multiple investigators are able to enhance the insights and results in the analysis of a case study [63]. Therefore, three researchers conducted the analysis. We used the description of the existing ERP systems as well as our previous professional experiences with ERP systems. We recorded reasons from our collected data for the dependency stages. These notes were coded [65] which results in a verbal proposition that is appropriate to be able to make controlled deductions [62]. The coding enabled us to order the data and derive cross-case dependency categories. Based on prior research (discussed in Section 2), our focus was on the data level and the functionality level. To measure the dependency, we used a qualitative approach [66]. Therefore and for each shadow system, all researchers at first identified the used data and the provided functionalities separately. Afterwards, each of the researchers had a closer look at the data and each functionality and analyzed the corresponding ERP system regarding these two dimensions. Then, we commonly discussed, which data and functionalities were part of the shadow system. In a next step, we mutually decided whether there was corresponding data or a corresponding functionality in the ERP system. For the data level, this means that all researchers agreed that the shadow system shares its data with the corresponding ERP system. In this case, we consider the data as dependent. For the functionality level, it means that all researchers agreed that the shadow system has a similar or identical functionality as the ERP system. We then consider the functionality as dependent. After we commonly made a decision about the dependency of all the data and all the functionalities of the shadow system, we derived the overall dependency. If we could not classify the shadow system clearly as dependent or independent, we classified it in the category grey zone. The next chapter discusses our findings.

4. Findings

Based on our analyses, this section describes general results of the chosen cases. Then, we give detailed examples of shadow systems to illustrate the categories of dependency on the corresponding ERP systems. Afterwards, we discuss these findings to illustrate how they reply to the posed research question.

4.1. Dependency Stages of Shadow Systems and ERP Systems in the Case Studies

After conducting the interviews, we found 99 shadow systems. In organization A, we identified six shadow systems, 52 in organization B, and 41 in organization C. After the analysis of the instances, we derived three dependency categories: dependency, no dependency, and grey zone. In the category grey zone, we sum up three types of dependencies that we cannot clearly identify as dependent or independent. We describe and give examples for each of the categories in the following.

4.1.1. Dependency

We found 27 shadow systems that are dependent on the ERP system, following our argumentation in the prior section. The systems are all in this category, because their functionality is one of the ERP systems of the analyzed organization. Additionally, the ERP systems share the data that the shadow system processes.

- **System A:** In Case B, in the team of order processing of special products, we found a Microsoft Access application on an SQL (Structured Query Language) database to create offers for special products. It is associated with another shadow system for technical calculations and a database for orders. The creation of offers is a core functionality of the existing ERP system. Furthermore, the saved data is core data of the ERP system. Because the shadow system shares all its data with the ERP system and the ERP system can provide all the functionalities of the shadow system, we classify these systems as dependent.
- **System B:** In Case B, in the teams of quality management and spare part processing, the employees used an online portal for retrieving product information. The data are stored as pdf documents. Additionally, the system provides an authorization scheme to distinguish between different roles that have access to different documents. Product information, as well as authorization schemes, are core parts of the formal ERP system. Therefore, this shadow system belongs to this category.

- **System C:** In Case C, in the team of trade fairs and events, we found a small web based tool for inventory control that only one warehouseman uses. Inventory control is a core functionality of the ERP system as well as the storage of inventory data. Therefore, we classify the two systems as dependent.

4.1.2. No Dependency

For 36 shadow systems, we did not find a dependency on the corresponding ERP system. This means that they do not share data or have common functionalities. Therefore, the shadow systems are part of this category.

- **System D:** Employees in Case C, in the team of corporate marketing, used a free content management framework for setting up websites about the company in the business unit. As this is not a functionality in an ERP system and the ERP system does not provide the data, we classify them as independent.
- **System E:** The team of corporate marketing in Case C used a collection of tools for editing graphics and videos. ERP systems provide neither the functionalities nor the data, which puts the shadow system into this category.
- **System F:** In Case C, the team of trade fairs and events applied software for the development of applications for a specific web application platform. The shadow system also accounts for independent, as the ERP system provides neither the functionality nor the data.

4.1.3. Grey Zone

The 36 shadow systems in our cases exemplify three types within this category. The first type will occur, if only the data of the shadow system is dependent. The second type will occur, if the shadow system only has parts of its data and/or functionality in common with the ERP system. Shadow systems within the third type provide a functionality and data that we cannot clearly categorize as dependent.

In six cases, there is only a dependency of data, but not of the functionality. The shadow system shares its data with the ERP system. However, the ERP system cannot provide the functionality of the shadow system.

- **System G:** In Case B, the team of construction uses an external software to stamp drawings from the ERP system. Therefore, the project from the ERP system is loaded into the software. Afterwards, the team puts the stamp on the drawing. The software shares the data with the ERP system, but the ERP system cannot provide the functionality. Therefore, data is dependent and already loosely integrated. Functionality, however, is not.
- **System H:** In Case C, the team of corporate design loads data from the ERP system into the database of an online shop. The team takes the data from the ERP system. The data is therefore dependent and integrated via a manual interface, whereas the ERP system cannot provide the functionality of the upload. The functionality therefore is independent, whereas the data is.

In 14 cases, the shadow system and the ERP system are only partially dependent. The shadow system shares only parts of its data with the ERP system and additionally uses data from other sources. Another possibility is that the ERP system can provide only some of the functionalities of the shadow system but not all. It can also occur that the shadow system shares only some of its data and some of its functionalities with the ERP system.

- **System I:** In Case A, the team of benefits for organizations uses benefit and case reports for insurances. The provided summary of benefits in the ERP system is not sufficient. Therefore, each employee uses a tool of end user computing. There, she accumulates data from the ERP system as well as from other information sources, depending on the circumstances.

- **System J:** In Case B, the team of order processing uses a web application. It uses data from the ERP system but also external order data. It has several functionalities, such as the granting of discounts, planning of dates and the display of milestones for the order. The ERP system can provide all of these functionalities. Hence, all of the functionalities are dependent but only parts of the data.

The third reason for this category occurs, when one or all of the functionalities and the associated data of the shadow system are part of the ERP system. However, they are no core capability of the ERP system. They are rather far from the original purpose of the system. We therefore call this state of dependency unclear. However, the system can share further data or functionalities with the ERP system. We found this subcategory in 16 cases.

- **System K:** The team of corporate marketing in Case C uses a self-developed system, based on ASP (Active Server Pages) to archive documents in different languages. Although the ERP system can provide archiving, it is not its core competency. It shares part of its data with the ERP system, but also uses external sources. Due to the unclear functionality, we cannot make a clear proposition about its dependency on the ERP system.
- **System L:** The team of order processing of special products in Case B uses a shadow system to describe big projects in different languages. They use building text blocks to combine technical sales and distribution aspects. They use another system for pricing, but with some additional functionalities for customs and big orders. The pricing functionality and the data of distribution are parts of the ERP system. However, the functionality for combining building blocks is not a core competency of the ERP system and we can therefore not clearly define its dependency.

Figure 1 illustrates the resulting categories. We show the different dependencies of data and functionality of shadow systems to the ERP system. Additionally, we provide the percentage of each category that we found in our cases. The next section discusses the found dependency stages and percentages with regard to integration recommendations.

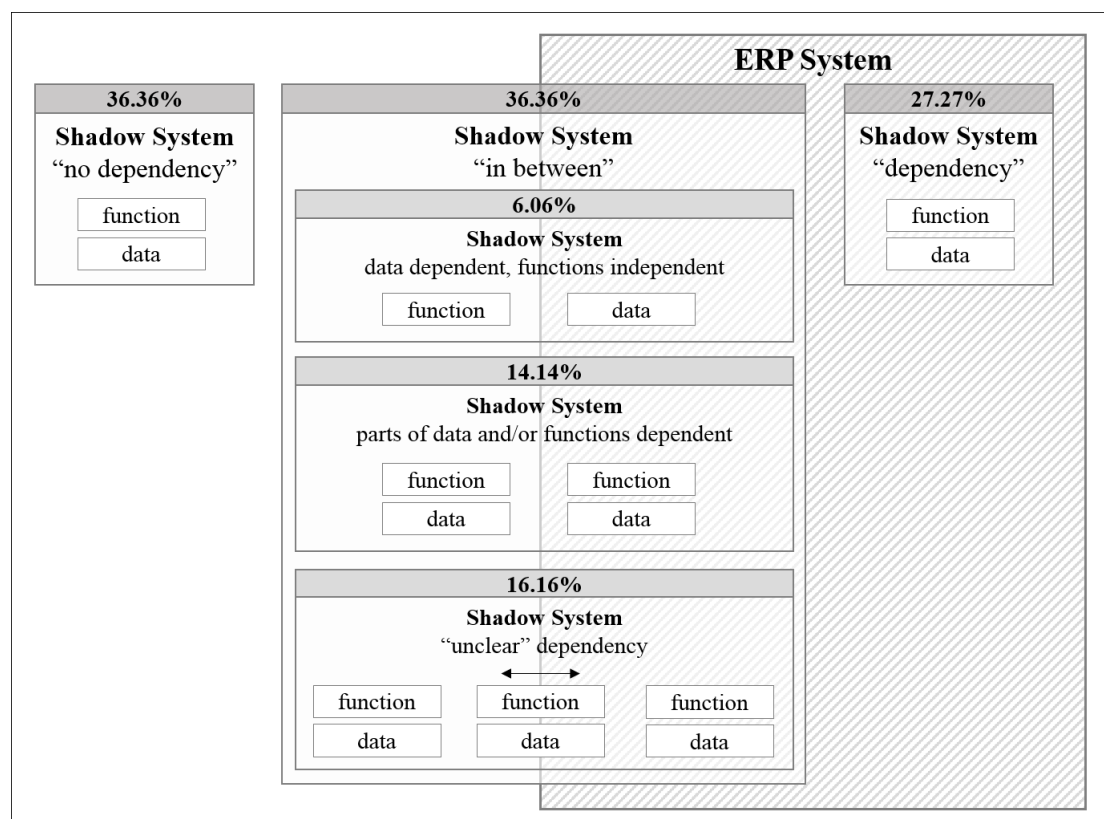


Figure 1. Shadow systems categories and their distribution.

4.2. Discussion

Considering the findings, we are able to reply to the posed research question: What dependencies of shadow systems and ERP systems can we find in practice? We have seen that identified dependencies are complex. Therefore, we cannot easily classify shadow systems and ERP systems as dependent or independent in every case. Besides the categories of dependency and no dependency, we found three states of dependency in between these categories, summed up in the category grey zone.

Based on this categorization, we can provide an insight into general integration possibilities that can be total, partial, or no integration [67]. Shadow systems in the category dependency share the same data and have the same functionality as the corresponding ERP system. Hence, a transfer of these systems into the ERP system would be possible. Independent shadow systems share neither their data nor their functionality with the corresponding ERP system. Therefore, integration might not be preferable due to a heavy change effort [32]. However, organizations should analyze the dependency on other enterprise systems to uncover further integration possibilities. If organizations decide against an integration and to leave the shadow systems in the business unit, a coordination of related IT tasks between business and IT units will be appropriate to ensure a high quality and reduce risk [21].

For shadow systems that are part of the category grey zone the corresponding ERP system is partly dependent or its dependency is unclear. The integration possibilities that we are able to state for this category in general are more complex. In these cases, organizations have to decide about the scope of integration. In our first type of the category grey zone, we found shadow systems that share all their data with the ERP system, but the ERP system cannot provide the functionality. As we did not find the opposite, this phenomenon points at deficiencies of the functionalities of the existing ERP systems as already discussed in research [20]. In some cases, systems share their data via an interface and therefore a loose integration already exists. However, users often handle these interfaces manually, which puts data integrity and security at risk [55]. As prior research also sees data integrity as a major problem in matured ERP systems [68], possible integration could be the provision of a secured data interface to the shadow systems to ensure data integrity. Furthermore, organizations could enhance the ERP system to integrate the functionality of the shadow systems [32]. In the second type of the category, there are partial dependencies of data or the functionality of the shadow systems. Organizations could decide to integrate the shadow systems partially. This integration could follow the different integration levels and the related data and functionalities. The third type contains shadow systems with an unclear dependency on the ERP system. This follows on an unclear affiliation of the functionality or the data to the ERP system. Therefore, organizations have to define what an ERP system should cover to be able to make a clear statement about integration possibilities. Alike to the previously discussed category, organizations should analyze the dependency on other enterprise systems and act accordingly.

In 28% of the shadow system instances, we found a dependency on the ERP system. This means that 28% of the identified shadow systems substitute the existing ERP system, which is also discovered by prior research [53,56,69]. The ERP system does not meet the needs of the business users in these cases. An additional 20% are partly dependent and 16% have an unclear dependency. These 36% of the shadow systems also point to deficient functionalities of the ERP system. The results highlight the relevance of integrating shadow systems into ERP systems in general to fulfill the needs of business users. By providing an integration approach, our study extends the research in the field of integration in general [25], especially considering integration as an ongoing task and not only during the implementation process of an ERP system [26,27]. Additionally, the research considers various instances of shadow systems, which extends the field of ERP systems and shadow systems.

5. Conclusions

The relation between shadow systems and ERP systems has different occurrences. We can describe these by determining various dependency stages. These stages can serve as a starting point for deriving integration recommendations. Integration recommendations for shadow systems follow the dependency of data and functionality on the ERP systems. For some dependency stages, we recommend the transfer to the ERP system or their partial or full integration. For others, we recommend

not to integrate but to analyze the dependency on other enterprise systems. The integration of former stand-alone shadow systems into ERP systems enhances enterprise integration.

We found a partial or full dependency and therefore a possibility for an enhancement of enterprise integration in 64% of the analyzed shadow systems in the case studies. These results point at weaknesses of the ERP systems in practice. Practitioners can use the results to compare shadow systems and ERP systems. Based on this, they can decide about the integration of identified shadow systems to ensure that the ERP system meets its goal of a high enterprise integration. With regard to theoretical implications, the study contributes to the topic of single system integration. Researchers can use the provided approach to conclude from the dependency of two systems to their integration. Furthermore, our findings extend the research field of shadow IT regarding its management with the focus on integration. It especially extends studies on shadow systems and established ERP systems that focus mostly on the implementation phase.

However, we have to acknowledge several limitations in our study that future research may address. The study includes only three companies. A broader empirical basis could enlarge the empirical insights. Additionally, our integration recommendations are mainly based on literature. An empirical testing of these recommendations could increase their validity. Furthermore, our research only focuses on the two integration levels of data and functionality. To ensure a valid integration decision, further research may incorporate more criteria for deciding about the integration of shadow systems into ERP systems and enterprise systems. Possible additional factors are cost, benefits, and architectural strategies of the integration. This includes risk considerations putting a research focus on shadow IT. These criteria may be empirically tested and evaluated accordingly. Additionally, system boundaries provide another possible focus of future research. These are crucial for deciding about an integration. Research has to develop methods to set these boundaries. Besides, research should be expanded to enterprise systems. The results could enhance insights into the efficiency of enterprise systems in general. Further studies may include organizations that do not possess a typical ERP system, such as the banking sector or service providers in general. Thereby, research could target an overall view on the integration of shadow systems into enterprise systems. Such a perspective could also use our findings to include the scalability of their integration into future research. This would additionally add to a broader empirical basis and a generalization of the results concerning integration of single IT systems in general.

Author Contributions: M.H., C.R., S.Z. and C.F. are all part of the research project. C.R. and S.Z. performed the case studies. M.H., C.R. and S.Z. analyzed the data. M.H., S.Z., C.R. and C.F. mutually discussed the results as well as the scientific contributions and determined the structure of the research paper. M.H. wrote the paper and consulted S.Z., C.R. and C.F. regularly to receive their feedback concerning the current status.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Smyth, K.; Freeman, J. *Blue Prism Rogue IT Survey 2007*; Blue Prism Ltd.: London, UK, 2007.
2. Chejfec, T. Shadow IT Survey v3. 2012. Available online: <http://chejfec.com/2012/11/03/Shadow-it-infographic/Shadow-it-survey-v3/> (accessed on 12 October 2015).
3. Zimmermann, S.; Rentrop, C. Schatten-IT. *HMD Prax. Wirtsch.* **2012**, *49*, 60–68. [[CrossRef](#)]
4. Chua, C.; Storey, V.; Chen, L. Central IT or Shadow IT? Factors Shaping Users' Decision to Go Rogue with IT. In Proceedings of the 35th International Conference on Information Systems, Auckland, New Zealand, 14–17 December 2014.
5. Tambo, T.; Baekgaard, L. Dilemmas in enterprise architecture research and practice from a perspective of feral information systems. In Proceedings of the 17th IEEE Enterprise Distributed Object Computing Conference Workshops, Vancouver, BC, Canada, 9–13 September 2013; pp. 289–295.
6. Houghton, L.; Kerr, D.V. A study into the creation of feral information systems as a response to an ERP implementation within the supply chain of a large government-owned corporation. *Int. J. Internet Enterp. Manag.* **2006**, *4*, 135–147. [[CrossRef](#)]
7. Behrens, S.; Sedera, W. Why do shadow systems exist after an ERP implementation? Lessons from a case study: Paper 136. In Proceedings of the 8th Pacific Asia Conference on Information Systems, Shanghai, China, 8–11 July 2004; pp. 1712–1726.

8. Silva, L.; Fulk, H.K. From disruptions to struggles: Theorizing power in ERP implementation projects. *Inf. Organ.* **2012**, *22*, 227–251. [[CrossRef](#)]
9. Boudreau, M.C.; Robey, D. Enacting Integrated Information Technology: A Human Agency Perspective. *Organ. Sci.* **2005**, *16*, 3–18. [[CrossRef](#)]
10. Lyytinen, K.; Newman, M. A tale of two coalitions—Marginalising the users while successfully implementing an enterprise resource planning system. *Inf. Syst. J.* **2015**, *25*, 71–101. [[CrossRef](#)]
11. Davenport, T.H. Putting the enterprise into the enterprise system. *Harv. Bus. Rev.* **1998**, *76*, 121–131. [[PubMed](#)]
12. Hochstein, A.; Zarnekow, R.; Brenner, W. ITIL as common practice reference model for IT service management: Formal assessment and implications for practice. In Proceedings of the International Conference on e-Technology, E-Commerce and E-Service, Hong Kong, China, 29 March–1 April 2005; pp. 704–710.
13. Sammon, D.; Adam, F. Towards a model of organisational prerequisites for enterprise-wide systems integration: Examining ERP and data warehousing. *J. Enterp. Inf. Manag.* **2005**, *18*, 458–470.
14. Thatte, S.; Grainger, N. Feral Systems: Why Users Write Them and How They Add Value. In Proceedings of the 5th Pre-ICIS Workshop on ES Research, St. Louis, MO, USA, 11 December 2010; pp. 1–16.
15. Sabău, G.; Muntean, M.; Bologa, A.R.; Bologa, R. Analysis of Integrated Software Solutions Market for Romanian Higher Education. *Econ. Comput. Econ. Cybern. Stud. Res.* **2009**, *43*, 197–202.
16. Zimmermann, S.; Rentrop, C. On the Emergence of Shadow IT—A Transaction Cost-Based Approach. In Proceedings of the 22nd European Conference on Information Systems, Tel Aviv, Israel, 9–11 June 2014; pp. 1–17.
17. Fürstenau, D.; Rothe, H. Shadow IT systems: Discerning the good and the evil. In Proceedings of the 22nd European Conference on Information Systems, Tel Aviv, Israel, 9–11 June 2014.
18. Ross, J.W.; Weill, P.; Robertson, D. *Enterprise Architecture as Strategy: Creating a Foundation for Business Execution*; Harvard Business Press: Cambridge, MA, USA, 2006.
19. Hoogervorst, J.A.N. Enterprise Architecture: Enabling Integration, Ability and Change. *Int. J. Coop. Inf. Syst.* **2004**, *13*, 213–233. [[CrossRef](#)]
20. Themistocleous, M.; Irani, Z.; O’Keefe, R.M.; Paul, R. ERP problems and application integration issues: An empirical survey. In Proceedings of the 34th Hawaii International Conference on System Sciences, Maui, HI, USA, 3–6 January 2001; pp. 1–10.
21. Zimmermann, S.; Rentrop, C.; Felden, C. Managing Shadow IT Instances—A Method to Control Autonomous IT Solutions in the Business Departments. In Proceedings of the 20th American Conference on Information Systems, Savannah, Georgia, 7–9 August 2014; pp. 1–12.
22. Kerr, D.; Houghton, L.; Burgess, K. Power Relationships that Lead to the Development of Feral Systems. *Australas. J. Inf. Syst.* **2007**, *14*, 141–152. [[CrossRef](#)]
23. Wendt, T.; Brigl, B.; Winter, A. Assessing the integration of information system components. In Proceedings of the 1st International Workshop on Interoperability of Heterogeneous Information Systems, Bremen, Germany, 31 October–5 November 2005; ACM: New York, NY, USA, 2005; pp. 55–62.
24. Ragowsky, A.; Licker, P.; Miller, J.; Gefen, D.; Stern, M. Do Not Call Me Chief Information Officer, But Chief Integration Officer. A Summary of the 2011 Detroit CIO Roundtable. *Commun. Assoc. Inf. Syst.* **2014**, *34*, 1333–1346.
25. Chowanetz, M.; Legner, C.; Thiesse, F. Integration: An Omitted Variable in Information Systems Research: Paper 227. In Proceedings of the European Conference of Information Systems, Barcelona, Spain, 10–13 June 2012.
26. Kähkönen, T.; Maglyas, A.; Smolander, K. The Life Cycle Challenge of ERP System Integration. In Proceedings of the Information Systems Development: Transforming Organisations and Society through Information Systems, Varaždin, Croatia, 2–4 September 2014.
27. Modol, J. A methodological and conceptual review of inter organizational information systems integration. In Proceedings of the European Conference on Information Systems, Gothenburg, Sweden, 12–14 June 2006; pp. 402–413.
28. Scott, S.V.; Wagner, E.L. Networks, negotiations, and new times: The implementation of enterprise resource planning into an academic administration. *Inf. Organ.* **2003**, *13*, 285–313. [[CrossRef](#)]

29. Jones, D.; Behrens, S.; Jamieson, K.; Tansley, E. The rise and fall of a shadow system: Lessons for enterprise system implementation: Paper 96. In Proceedings of the 15th Australasian Conference on Information Systems, Hobart, Australia, 1–3 December 2004.
30. Behrens, S. Shadow systems: The good, the bad and the ugly. *Commun. ACM* **2009**, *52*, 124–129. [[CrossRef](#)]
31. Ignatiadis, I.; Nandhakumar, J. The effect of ERP system workarounds on organizational control: An interpretivist case study. *Scand. J. Inf. Syst.* **2009**, *21*, 1–34.
32. Brehm, L.; Heinzl, A.; Markus, M.L. Tailoring ERP systems: A spectrum of choices and their implications. In Proceedings of the Hawaii International Conference on System Sciences, Maui, HI, USA, 3–6 January 2001; pp. 1–9.
33. Kerr, D.; Houghton, L. The dark side of ERP implementations: Narratives of domination, confusion and disruptive ambiguity. *Prometheus* **2015**, 1–15. [[CrossRef](#)]
34. Webster, J.; Watson, R.T. Analyzing the past to prepare for the future: Writing a literature review. *Manag. Inf. Syst. Q.* **2002**, *26*, 8–13.
35. Soh, C.; Kien, S.S.; Tay-Yap, J. Enterprise resource planning: Cultural fits and misfits: Is ERP a universal solution? *Commun. ACM* **2000**, *43*, 47–51. [[CrossRef](#)]
36. Land, F. *A historical analysis of implementing IS at J. Lyons*; Oxford University Press: Oxford, UK, 1999.
37. Olson, D.L.; Kesharwani, S. *Enterprise Information Systems: Contemporary Trends and Issues*; World Scientific: Singapore, 2010.
38. Kien, S.; Lian, Y. Building Enterprise Integration Through Enterprise Resource Planning Systems: Paper 169. In Proceedings of the International Conference on Information Systems, Phoenix, AZ, USA, 14 December 2009.
39. Hanseth, O.; Lyytinen, K. Theorizing about the design of information infrastructures: Design kernel theories and principles. *Sprouts* **2004**, *4*, 207–241.
40. Lin, D. An information-theoretic definition of similarity. In Proceedings of the 15th International Conference on Machine Learning, Madison, WI, USA, 24–27 July 1998.
41. Mertens, P. *Integrierte Informationsverarbeitung 1-Operative Systeme in der Industrie 16. Überarbeitete Auflage*; Gabler: Wiesbaden, Germany, 2007.
42. Linß, H. *Integrationsabhängige Nutzeffekte der Informationsverarbeitung: Vorgehensmodell und Empirische Ergebnisse*; Dt. Univ.-Verlag: Wiesbaden, Germany, 1995.
43. Rosemann, M. Gegenstand und Aufgaben des Integrationsmanagements. *Integrationsmanagement* **1999**, *5*, 5–18.
44. Heilmann, H. Integration: Ein zentraler Begriff der Wirtschaftsinformatik im Wandel der Zeit. *HMD* **1989**, *26*, 46–58.
45. Ruh, W.A.; Maginnis, F.X.; Brown, W.J. *Enterprise Application Integration: A Wiley Tech Brief*; Wiley: Hoboken, NJ, USA, 2002.
46. Vernadat, F.B. Enterprise Modelling and Integration: From Fact Modelling to Enterprise Interoperability. In *IFIP—The International Federation for Information Processing*; Springer US: New York, NY, USA, 2003; pp. 25–33.
47. Giachetti, R.E. A framework to review the information integration of the enterprise. *Int. J. Prod. Res.* **2004**, *42*, 1147–1166. [[CrossRef](#)]
48. Frank, U. Integration—Reflections on a Pivotal Concept for Designing and Evaluating Information Systems. In *Lecture Notes in Business Information Processing*; Springer: Berlin, Germany; Heidelberg, Germany, 2008; pp. 111–122.
49. Kerr, D.V.; Houghton, L. Just in time or Just in case: A Case study on the impact of context in ERP implementations. *Austral. J. Inf. Syst.* **2010**, *16*. [[CrossRef](#)]
50. Le Roux, D.B. Misfit and Reinvention in Information Systems: The Case of a South African Metropolitan Municipality. In Proceedings of the Southern African Institute for Computer Scientist and Information Technologists Annual Conference, Centurion, South Africa, 30 September–1 October 2014; pp. 217–228.
51. Robey, D.; Ross, J.W.; Boudreau, M.C. Learning to implement enterprise systems: An exploratory study of the dialectics of change. *J. Manag. Inf. Syst.* **2002**, *19*, 17–46.
52. Strong, D.M.; Volkoff, O. A roadmap for enterprise system implementation. *Computer* **2004**, *37*, 22–29. [[CrossRef](#)]
53. Oliver, D.; Romm, C. Justifying enterprise resource planning adoption. *J. Inf. Technol.* **2002**, *17*, 199–213. [[CrossRef](#)]

54. Kallinikos, J. Deconstructing information packages: Organizational and behavioural implications of ERP systems. *Inf. Technol. People* **2004**, *17*, 8–30. [[CrossRef](#)]
55. Kulkarni, A.; Williams, E.; Grimaila, M.R. Mitigating Security Risks for End User Computing Application (EUCA) Data. In Proceedings of the IEEE 2nd International Conference on Social Computing (SocialCom), Minneapolis, MN, USA, 20–22 August 2010; pp. 1171–1176.
56. Kerr, D.; Houghton, L. Feral Systems: The Likely Effects on Business Analytics Functions in an Enterprise Resource Planning System Environment. In Proceedings of the 19th Australasian Conference on Information Systems, Christchurch, New Zealand, 3–5 December 2008; pp. 484–491.
57. Thatte, S.; Grainger, N.; McKay, J. Feral practices. In Proceedings of the 23rd Australasian Conference on Information Systems, Geelong, Victoria, Australia, 3–5 December 2012; pp. 1–10.
58. Lee, A.S. Case studies as natural experiments. *Hum. Relat.* **1989**, *42*, 117–137. [[CrossRef](#)]
59. Yin, R.K. *Case Study Research: Design and Methods*; Sage Publications: Thousand Oaks, CA, USA, 2013.
60. Benbasat, I.; Goldstein, D.K.; Mead, M. The case research strategy in studies of information systems. *MIS Q.* **1987**, 369–386. [[CrossRef](#)]
61. Herriott, R.E.; Firestone, W.A. Multisite qualitative policy research: Optimizing description and generalizability. *Educ. Res.* **1983**, 14–19. [[CrossRef](#)]
62. Lee, A.S. A Scientific Methodology for MIS Case Studies. *MIS Q.* **1989**, *13*, 33–50. [[CrossRef](#)]
63. Eisenhardt, K.M. Building Theories from Case Study Research. *Acad. Manag. Rev.* **1989**, *14*, 532–550.
64. Myers, M.D.; Newman, M. The qualitative interview in IS research: Examining the craft. *Inf. Organ.* **2007**, *17*, 2–26. [[CrossRef](#)]
65. Corbin, J.; Strauss, A. *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*; Sage Publications: Thousand Oaks, CA, USA, 2014.
66. Flick, U. *An Introduction to Qualitative Research*; Sage: Thousand Oaks, CA, USA, 2009.
67. Giacomazzi, F.; Panella, C.; Pernici, B.; Sansoni, M. Information systems integration in mergers and acquisitions: A normative model. *Inf. Manag.* **1997**, *32*, 289–302. [[CrossRef](#)]
68. Carton, F.; Adam, F. Towards a model for determining the scope of ICT integration in the enterprise: The case of Enterprise Resource Planning (ERP) systems. In Proceedings of the 3rd European Conference on Information Management and Evaluation, Gothenburg, Sweden, 17–18 September 2009.
69. Wagner, E.L.; Newell, S.; Piccoli, G. Understanding Project Survival in an ES Environment: A Sociomaterial Practice Perspective. *J. Assoc. Inf. Syst.* **2010**, *11*, 276–297.



© 2016 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons by Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).