5-11-2023

IS Architecture Complexity Dynamics in M&A: Does Consolidation Reduce Complexity?

Eric Onderdelinden
Vrije Universiteit Amsterdam, eonderdelinden@deloitte.nl

Bart van den Hooff
Vrije Universiteit Amsterdam, b.j.vanden.hooff@vu.nl

Mario van Vliet
Vrije Universiteit Amsterdam, mvanvliet@deloitte.nl

IS ARCHITECTURE COMPLEXITY DYNAMICS IN M&A: DOES CONSOLIDATION REDUCE COMPLEXITY?

Research Paper

Eric Onderdelinden, Vrije Universiteit Amsterdam, Netherlands, eonderdelinden@deloitte.nl

Bart van den Hooff, Vrije Universiteit Amsterdam, Netherlands, b.j.vanden.hooff@vu.nl

Mario van Vliet, Vrije Universiteit Amsterdam, Netherlands, m2.vanvliet@vu.nl

Abstract

In this paper we aim to improve our understanding of the dynamics of IS architecture complexity (i.e, the change in this complexity over time) during the execution of a consolidation IS integration strategy (IIS). Based on two case studies, we find that unexpected levels of complexity emerge during IIS execution because of an underestimation of requisite complexity and an overestimation of the potential to reduce complexity. Our analysis shows that increased complexity is due to the fact that the intended consolidation IIS is only partially executed, and to increasingly emergent IIS execution. Additionally, we find that while complexity was reduced at the portfolio level, at more detailed levels of observation complexity was actually increased. Our paper contributes to knowledge in the field by providing a deeper insight into IS architecture complexity dynamics during the execution of a consolidation IIS, and the concept of IS architecture complexity in general.

Keywords: : IS integration, Consolidation, Complexity, M&A.

1 Introduction

Mergers and Acquisitions (M&A) are common events in the lifecycle of organizations. The predominant motivation behind M&A is to create value through exploiting potential synergies (Chatterjee, 1986). Synergy within the Information Systems (IS) domain represents a commonly accepted source of value, making IS integration an important factor in the success of M&A (Buck-Lew et al., 1992; McKiernan and Merali, 1995; Mehta and Hirschheim, 2007). At the same time, however, IS integration problems are commonly identified as an important factor explaining why the majority of M&A transactions do not deliver the envisioned value (Henningsson and Kettinger, 2016; Hitt et al., 2009). The complexity of the post-merger IS architecture is an important factor in explaining this disappointing value creation: this complexity often exceeds the level of complexity planned or expected pre-merger, leading to cost increases, inefficiencies, the need for additional resources and other issues that negatively affect value creation (Batelaan and Veltman, 2002; Busquets, 2015; Henningsson and Kettinger, 2016; Johnston and Yetton, 1996; Wijnhoven et al., 2006).

Although previous research on IS integration in M&A has focused on complexity as an independent variable, we are lacking insight into how the integration of previously independent IS architectures influences the complexity of the resulting landscape – i.e., complexity as a dependent variable (Henningsson et al., 2018; Yetton et al., 2022). When we consider the influence of IS integration on the complexity of the resulting IS landscape, it is clear that the choices made in terms of the IS Integration Strategy (IIS) have an influence here. For instance, a *co-existence* strategy, where the landscapes of both organizations are essentially retained (Baker and Niederman, 2014), is likely to lead to more complexity post-merger than pre-merger. A *consolidation* IIS however, where one organization is fully absorbed by the other and its IS landscape is discarded (Baker and Niederman, 2014), is often aimed at a reduction of complexity.

In this paper, we specifically focus on the consequences of a consolidation IIS for IS architecture complexity. This is a common IIS in practice, and it is often found to have a paradoxical nature: although the theoretical outcome of this strategy would be reduced complexity compared to the situation before the merger (one IS landscape instead of two), the objectives and expected outcomes of a consolidation IIS conflict with the principle of requisite complexity. This principle holds that every system should contain a minimum level of complexity in order to deal with requirements stemming from the system's environment (Boisot and McKelvey, 2011). As a consolidation strategy means that the size and complexity of the post-merger organization increases, following this principle would mean that IS complexity would also increase. As previous research provides diverging and sometimes conflicting insights into IS architecture complexity in M&A, our aim is to provide a more systematic insight into why and how such complexity develops in M&A. In this paper, we aim to address this gap by answering the question: *"How does IS architecture complexity develop over time while executing a consolidation IIS and what factors impact the dynamics of this complexity?"*

In our two qualitative case studies, we find that IS complexity increased during the execution of the consolidation IIS. In both cases, the requisite IS architecture complexity was underestimated pre-merger. Our detailed analysis shows that several M&A-specific constraints limit the ability to accurately assess post-merger IS complexity during M&A planning and that assumptions are formulated to compensate for this. These assumptions were not validated and their inaccuracy triggered emergent adaptation of the planned integration strategy, resulting in increased complexity. Furthermore, failing to discontinue applications, combined with temporary solutions, contributed to a further increase in IS architecture complexity.

We first review the literature on IS integration strategies and IS architecture complexity, followed by an overview of the methods used in our empirical study. Next we present our findings, focusing on complexity dynamics for different architectural domains (infrastructure, data, and applications) which differ from each other and contrast the portfolio level dynamics. We conclude with the theoretical implications and practical suggestions.

2 Literature Review

2.1 M&A, value objective realization and the role of IS integration

An M&A is a strategic decision to combine resources from two formerly separate firms into a new organization, changing firms and providing opportunities for renewal and growth (Giacomazzi et al., 1997). The M&A value objective represents the management vision on the value that can be created by exploiting potential synergies through combining previously independent resources (Haspeslagh and Jemison, 1991; Lubatkin, 1983). The Business Integration Strategy (BIS), derived from the M&A value objective, defines which of the potential sources of synergy will be exploited and provides guidance on the integration of organizational resources (Haspeslagh and Jemison, 1991).

Within M&A, integration of Information Systems (IS) represents a commonly accepted source of value (Buck-Lew et al., 1992; Hanelt et al., 2021), with the pre-merger IS landscape significantly influencing how M&A are shaped (Cao et al., 2022a) and their outcomes (Cao et al., 2022b). Hitt et al. (2009), however, report that the majority of M&A transactions do not deliver the envisioned value and that IS integration problems contribute to the failure in realizing expected synergies. The complexity of the post-merger IS architecture has been identified as one of the potential causes for this disappointing value creation through IS integration (Busquets, 2015; Johnston and Yetton, 1996; Seddon et al., 2010). Higher levels of IS architecture complexity are associated with increased running costs and slowdown of innovation (Henningsson and Yetton, 2013). Furthermore, complexity negatively influences IS flexibility (Beese et al., 2022), which is an important factor in determining M&A outcomes (Benitez et al., 2018). Security risks represent another negative consequence of increased IS complexity (Tanriverdi et al., 2019) .

2.2 IS architecture complexity

IS architecture is defined as "the overarching structure and properties of the relationships among the systems and applications in an organization's IT portfolio" (Tiwana and Konsynski, 2010, p. 288). The complexity of an organization's IS architecture is often conceptualized in terms of "the quantity and variety of system elements and the relationships between these" (Schmidt, 2015, p. 244). This can be characterized as a *structural* perspective on complexity: focusing on the number and variety of components, and the relationship between them (Beese et al., 2022; Xia and Lee, 2005; Widjaja and Gregory, 2020). In addition to this, there is also a *dynamic* dimension of complexity, wich refers to the continuous evolution of an architecture's form and function: the rate of change in the architecture (Haki et al., 2020; Widjaja and Gregory, 2020). As our focus is on the development of an IS architecture over time, how it changes due to IS integration, both the structural and dynamic dimension of complexity are relevant.

Where the definition of IS architecture complexity above refers to the complete *portfolio* of applications, complexity can also be defined at the level of applications themselves, referring to the size of the IS systems (applications) as the key complexity parameter (Henningsson et al., 2018; Robbins and Stylianou, 1999). The definition of complexity in terms of the number and variety of components and interaction beween them – and the rate of change concerning these – applies both at the portfolio and application levels.

IS architecture complexity in itself is not necessarily a negative thing, as the principle of requisite complexity illustrates (Boisot and McKelvey, 2011): every system should contain a minimum level of complexity in order to deal with requirements stemming from the system's environment. In line with this, Beese et al. (2022) find that business complexity drives technological complexity. Business complexity is characterized by the number, variety and interrelatedness of elements such as customers, business units, business processes, suppliers, partners, technologies, and regulations, as well as the rate with which these develop and change (Beese et al., 2022; Xia and Lee, 2005). In line with this principle, research on IS integration in M&A has found multiple examples of not meeting the requisite post-merger IS complexity - and resulting dysfunctional business processes (e.g., Busquets, 2015; Henningsson and Kettinger, 2016).

2.3 IS Integration Strategy and the dynamics of IS architecture complexity

How and to what extent the previously independent IS resources from both firms are integrated is essential in achieving the M&A value objective (Baker and Niederman, 2014). Integration implies selecting IS resources from both organizations and arranging these into a new IS architecture. The extent of integration ranges from complete integration to marginal integration (Wijnhoven et al., 2006). Complete integration aims for a significant reduction of the IS architecture complexity through removal of all the IS from one of the firms. This IS Integration Strategy (IIS) is commonly referred to as *consolidation* (Baker and Niederman, 2014). Marginal integration represent the other end of the spectrum and implies keeping all the IS from both firms while adding some integration and thus a relatively minor increase in complexity – a strategy of *co-existence* (Baker and Niederman, 2014). The other two integration strategies distinguished by Baker and Niederman (2014) are *combination* (combining superior IS components from both firms into the new IS landscape) and *transformation* (a completely new IS landscape to support completely new business functions and processes).

The execution of the IIS impacts the number of applications and the interfaces, and thus the complexity of the resulting IS architecture. This focus implies that in this paper, IS architecture complexity is primarily seen as a dependent variable, where much previous research has mainly considered this as an independent variable influencing IS integration (Henningsson et al., 2018). With the term *IS architecture complexity dynamics* we refer to the change in IS architecture complexity over time. Where a consolidation IIS should in theory lead to a reduction in complexity (as two pre-merger IS landscapes are reduced to one post-merger), this would be in conflict with the principle of requisite complexity and the idea that at IS complexity is driven by business complexity (Beese et al., 2022). This, together with the fact that consolidation is a strategy often found in practice, determines our focus on how a consolidation IIS influences the actual complexity of the resulting IS architecture.

2.4 Planned and emergent IS integration

While executing an IS integration strategy, IS complexity is considered and understood at an increasing level of detail. Moving from strategic planning to operational execution, through zooming in, inherently more components and interfaces (and complexities) in the IS architecture are discovered. Planned and emergent IIS execution are two distinct ways of dealing with these additional and more detailed insights.

Planned IIS execution implies strictly following the plan, building on the premise that IS architectures are fully understood, that all the relevant details of the acquired firm's IS are known, and that alignment between BIS and IIS has been adequately established while preparing for execution (Wijnhoven et al., 2006). Strictly sticking to the integration strategy implies ignoring the implications of the more detailed information becoming available during execution, which may trigger specific integration failure mechanisms (Baker and Niederman, 2014; Henningsson and Kettinger, 2016).

Emergent IIS execution is based on "letting things flow" (Busquets, 2015, p. 198) and represents an alternative approach. The emergent IIS evolves over time in response to a deeper understanding of IS, threats and opportunities. Emergent execution is generally more effective in dealing with uncertainty and complexity compared to planned execution (Sommer and Loch, 2004). Baker and Niederman (2014) found that in successful integrations, organizations are able to effectively handle surprises resulting from a lack of completeness of information in the due diligence process. In these successful examples, IIS is driven by sequential problem – solution pairings, representing an emergent strategy. Busquets (2015) illustrates that during an M&A an organization can evolve due to iterative problem solving, indicating that IIS concretely takes shape during execution and may differ from what was planned ex-ante.

In summary, IS architecture complexity dynamically changes during integration in M&A. The complexity dynamics may be a cause for failing to deliver the envisioned value. We currently only have limited insight in how and why IS architecture complexity changes dynamically during the (emergent) execution of a consolidation IIS. Although a consolidation IIS aims to reduce the IS complexity overall, this conflicts with the principle of requisite complexity. Thus, how IS architecture complexity develops during the execution of a consolidation IIS, is an open question which requires further empirical exploration. We will describe the methodological details of our empirical study in the next section.

3 Methods

The aim of our research is to develop a theory for analysis and description while aiming for explanation (Gregor, 2006). Qualitative research is an appropriate research approach for this type of research question (Klein and Myers, 1999). A case study research approach has been chosen because the context of actions and decisions is relevant and the practical experiences of the actors are important (Benbasat et al., 1987). We identified two relevant case studies together with a large Dutch consultancy firm engaged in many M&A-related IS integration projects focusing on IS-intensive organizations. The consultancy organization provided us access to an extensive archive of relevant information and artefacts covering the whole process of integration, from the preparation stages of the M&A up until the realized integration. Working with the consultancy organization additionally allowed us to conduct interviews with a large variety of people involved in the cases in different roles. This allowed us to collect multiple perspectives, contributing to the validity of our research through triangulation of our findings.

We conducted two subsequent case studies focusing on the development of IS complexity over time in M&A. The aim of the first case study was to develop an initial conceptual understanding of IS complexity dynamics in M&A, which we subsequently validated and extended in the second case study.

We chose to conduct more interviews in the first case study to come to a rich conceptual understanding, and to conduct fewer in the second case as it was focused on validation and refinement. The first case study concerns an M&A in the Telco industry and the second an M&A in the Financial Service Industry.

3.1 Case 1: Telco

The setting of the first case study is a merger between two large telecommunication companies in the Netherlands, described as "Alpha" and "Beta". "Alpha", acquired "Beta" to add the Beta customer base onto their own, as well as to connect the two physical networks to reach more potential customers. Alpha's parent company ("PCA)" counted on realizing business synergies through economies of scope and efficiency. The integrated firm rebranded itself to the outside world, taking on the name (and reputation) of Beta. We label the integrated firm "NewBeta". The rationale for the merger was based on strengthening NewBeta's market position by leveraging synergies through exploiting economies of scale and scope. The planned Business Integration Strategy (BIS) was absorption (Alpha fully integrating Beta), and the planned IIS was consolidation. The absorption BIS implied that Beta had to adopt the way of doing business from Alpha, although because of the brand value the Beta brand was retained. The consolidation IIS meant that PCA decided that Beta needed to conform to Alpha standards to improve efficiency by eliminating redundant systems. Our research covers the period from initial investigation prior to the official clearance until the formal end of the M&A phase. At that point in time, a next M&A was initiated.

3.2 Case 2: Bank

The setting of the second case study is a merger between banks in the Netherlands, described as "Real Estate Bank" (further REB) and "Large Bank" (LB) in this paper. REB was a small bank focused on commercial real estate financing. It operated mainly in the Netherlands and at the time of the merger was already a subsidiary of LB. LB is one of the largest Dutch banks and can be broadly divided in three business units: Local Banks, Business Financing and Wholesale. For this integration, an absorption BIS was planned combined with a consolidation IIS. Again, our data collection covers the entire integration trajectory until the formal end of the M&A. At that point in time all clients were migrated to the LB IS.

3.3 Data collection and analysis

We studied the Telco merger and the Bank merger sequentially. As explained, the first case study was aimed at developing an initial conceptual understanding of IS complexity dynamics in M&A, where the second one was more aimed at validating and extending this understanding. In that sense, the findings and outcomes from the first case study provided a rich starting point for the second case study. Collecting and validating multiple types of data allowed us to verify the results using triangulation as advised by Yin (2012). Throughout both case studies, our main sources of data were (1) archival data and design artifacts, and (2) interviews.

3.3.1 Archival data and design artifacts.

The integration of two previously independent and autonomous IS is a 'program' for which multiple documents and design artefacts are created. We gathered archival data and design artefacts related to the IS integration supplied by the consulting firm and the merging organizations. These documents (slide decks, reports, Enterprise Architecture (EA) artifacts, etc.) were mainly used to inform and shape the integration process. They provided a detailed description of the integration process and the specific steps that were taken. Using archival data helped determining what integration strategy the organizations aimed to pursue, and how they planned to execute the strategy.

The archival data and design artefacts were coded with a combination of provisional and emergent codes. The provisional codes (e.g. types of synergy, proposed integration strategies) were based on the literature reviewed (Miles et al., 2014). The emergent codes were developed during the analysis of the archival data. The list of potential reasons for engaging in M&A found in the literature on M&A is extensive and captured in the provisional codes while the emergent codes were specific to each of the cases, including case specific reasons for certain choices - for example why a particular integration strategy was chosen.

3.3.2 Interviews

For the Telco merger we conducted a total of 31 semi-structured interviews: 23 EA consultants from the consultancy organization who were directly involved in the integration, three former Alpha employees, three former Beta employees, and two consultants involved in a different integration project. As EA consultants undertake cross-organizational dialogue to construct EA planning activities, they can identify interdependencies between parts of the organization and how different issues affect each other. Furthermore, these EA consultants were typically involved in many projects throughout their career. This helps to position findings within the specific case, but also within a broader EA perspective. Interviews lasted between 45 and 60 minutes. The interviews allowed us to capture how the integration happened from different perspectives, yielding insights into the reasons why things happened and the consequences of the different decisions and actions. Every interview was recorded with explicit consent of the interviewee. All transcripts were thoroughly checked to ensure that all references to persons or organizations are removed or censored.

For the Bank merger, a total of nine semi-structured interviews were conducted: two employees of LB, one employee of REB, two consultants involved in the customer migration and four consultants that helped guide the M&A. The interviewees represent a diverse group in terms of hierarchy, function and business unit. Having diverse perspectives helped this study significantly, providing rich and detailed explanations that enhanced our understanding of how architectural complexity emerged.

After transcribing the recorded interviews, transcripts were analyzed and coded with Atlas.ti. For both case studies, three phases can be distinguished in the coding process: open coding, followed by axial coding where the initial codes were ordered in overarching categories. Finally, the codes and categories were prioritized and systematically related to each other to obtain a comprehensive view on how IS complexity in M&A develops over time, and what factors impact its development. Table 1 provides an overview of the final code structure.

Category	Case	Description
IS architecture complexity (change)	1,2	Codes related to [changes in] the quantity, variety and relationships between elements in an IS architecture.
Endogenous. exogenous (Mechanisms)	1,2	Codes related to mechanisms which caused an increase post-merger IS architecture complexity.
IS integration strategy (Change)	1,2	Codes related to the interventions taken following a merger or acquisition to execute or refine the IIS.
Strategic drivers	1,2	Codes related to strategic drivers of the IIS.
Operational Drivers	1,2	Codes related to operational drivers of the IS integration strategy and deviations from the planned IIS
Noteworthy and Interesting	1,2	Codes related to findings in the empirical data which cannot be plotted in the prior six categories.
Requisite Complexity/Architectural waste	1	Codes related to necessity of IS architecture complexity increase

Table 1. Overview of codes used in the data analysis.

4 Findings

In both cases, a significant IS complexity reduction was planned through a consolidation IS integration strategy. In both cases the IIS was aimed at keeping the applications, databases and infrastructure of the buyer while moving data from the acquired company to the buyer's IS. In both cases, contrary to pre-merger planning and deviating from the IIS, additional components were introduced during IIS execution, increasing the IS architecture complexity. We found different complexity dynamics in three different architecture domains: infrastructure, data, and applications. These will all be discussed in the following paragraphs.

4.1 Telco Case

As the foundation for formulating the BIS and IIS, the eTOM model was used (TM Forum, 2014). eTOM is widely adopted within the industry and describes the key business processes and interactions of telecom service providers, distinguishing two major process/application domains: the Business Support Systems (BSS) domain, and the Operations Support Systems (OSS) domain. BSS relates to customer supporting processes such as taking orders and processing bills, supported by, for instance, CRM and ERP systems. OSS refers to the network supporting processes such as maintaining network inventory and network planning, with associated applications such as service monitoring and configuration management. The eTOM is complemented with an Information Framework (SID) and an Application Framework (TAM) (TM Forum, 2014). All were used during the M&A preparation phase to assess the similarity of Alpha and Beta. The processes of both organizations appeared to be very similar, which was one of the main reasons for senior management engaged in strategic planning to opt for an absorption BIS and a consolidation IIS. These similarities between Alpha and Beta were not validated in more detail, as explained by the PCA lead architect:

"This [analysis] is done very high level. We take the Alpha BSS to replace the Beta BSS. We think roughly, it's all back of the napkin analysis, there hasn't been a detailed analysis [during preparation]."

The practical outcome of the high-level analysis was the aim to significantly reduce the IS architecture complexity, as illustrated by the lead integration architect: *"Because you have two of everything and the goal is to have one of everything."*

Higher than planned IS complexity was observed comparing the outcome of the high-level analysis and the results of the execution of the ISS. Infrastructure (the OSS) represents the most fundamental influence on the post-merger IS architecture complexity in this case and will be discussed first, followed by the IS complexity for applications and data.

4.1.1 Infrastructure complexity dynamics

During IIS execution, there was a clear increase in complexity for the infrastructure, which persisted until the formal end of the integration. This increase was caused by incompatibility in the OSS stacks and the absence of achievable value potential. Although the Beta OSS was not analysed pre-merger, the strategic assumption was that a consolidation IIS could be executed here, while not touching the physical infrastructure. However, according to both the Alpha and the Beta lead architects, applications and physical network are tightly coupled and cannot be changed independently.

"[…] the network, that's physical cables that lie in the ground that you must control, that you are not going to change, they remain there. So, you can't do much with the OSS part, technically speaking."

The tight coupling of OSS applications to the physical infrastructure made consolidation of the OSS impossible. Furthermore, experts involved in the integration signalled an absence of achievable value potential in OSS integration. Thus, for the OSS domain, the consolidation IIS had evolved into a co-existence IIS: both OSSes remained in existence. Further changes to the consolidation IIS resulted from the fact that the Alpha databases infrastructure was not able to handle the load of the merged company. This was addressed by replacing the Alpha servers with more modern, higher capacity servers – introducing a transformation element, as illustrated by the following quote from the delivery manager:

"The old system was not good enough; it could not handle all requests because they went from 2 to 5 million customers. The Alpha system will therefore be phased out and must be moved to a new system"

4.1.2 Application and application portfolio complexity dynamics

Increased IS complexity was observed for both the application portfolio and the individual applications. Keeping Alpha applications and discontinuing Beta applications was expected to reduce the application portfolio complexity. The decision to execute a co-existence ISS for the OSS-es, however, had a significant impact on the envisioned consolidation IIS. Although OSS and BSS are conceptually independent, numerous interdependencies exist. For instance, end-user equipment such as a set top box primarily impacts the OSS, as it introduces a new infrastructure element, which needs to be activated and managed. This, consequently, impacts the BSS as this new device needs to be delivered and invoiced through the BSS applications.

As a result of the decision to keep the Beta OSS, the envisioned application portfolio architecture consisted of Alpha OSS and Beta OSS and Alpha BSS. Therefore, additional interfaces needed to be developed to connect the Beta OSS to the Alpha BSS. For instance, the Beta CRM system was to be fully replaced by the Alpha system. In order to achieve this, the Beta customer data needed to be migrated to the Alpha CRM system. A temporary integration layer was created to support continued delivery of products and services to clients during the execution of the IIS, which led to an (albeit temporary) increase in complexity. Furthermore, specific Beta applications had to remain operational before being replaced by the comparable Alpha application, even after removal of the Beta CRM. This required changing and adding applications and interfaces. This increased the IS complexity even further, as illustrated by one data migration consultant: "*Well, every component of a system that communicates then also needs to be changed and then it gets really complicated.*"

During the execution of the consolidation IIS, many of the Beta specific services were identified as being of value. "Cherry picking" of Beta application features and introducing these into Alpha applications was observed on multiple occasions, shaping a combination IIS. This increased the IS complexity for applications. Although complexity for the application portfolio was reduced by decommissioning a Beta application, this needed to be compensated by introducing additional functionality within the Alpha application – a phenomenon we observed multiple times.

Thus, executing a consolidation IIS required a temporary increase of the IS complexity for the application portfolio because of the additional applications needed for data migration inherent to a consolidation IIS. Introducing combination IIS aspects was found to further (and more permanently) increase the IS complexity for both the application portfolio and individual applications.

4.1.3 Data complexity dynamics

Our findings also indicate an increase in the IS complexity for data, which was partly due to differences between specific data structures, but also to the changes to the IIS for applications. Pre-merger it was implicitly assumed, based on the similarity of business processes, that the Alpha applications would be able to handle the data from the Beta applications. This assumption concerned customer data, product data and service data. In practice, however, this was not the case. During the preparation of the migration of customer data, it was discovered that the Alpha CRM was not able to store and process the data volume required for (the significantly higher number of) Beta customers. Additionally, it was discovered that the Alpha CRM did not provide the data structures required for supporting several Beta specific products and services delivered to the Beta customers. As illustrated by the integration architect

"I understand that it seems like it almost stays the same, and the applications pretty much stay the same, but, for example, here we didn't have this free [product], and now we do. […] So now we have a new variant of [product] in the [product] manager, which requires [data] changes [in all communicating applications and interfaces]."

Another example of IS complexity for data underestimated pre-merger was provided by the Project manager:

"A very interesting one that is not included here is reporting; I want to know how many open orders I have, how many sales I have made. Now I suddenly must build a report that has to extract information from two systems."

So, although only one report is built, compared to two pre-merger, as is predicted pre-merger, for the execution of a consolidation IIS, the report itself has become more complex and must be fed from two systems. Another example of complexity reduction at the application portfolio level of observation, but an increase at the application level of observation.

4.2 Bank Case

For the Bank merger, the plan was that none of the REB applications would be used by LB after the merger. All customer related data and supporting documents would be migrated from REB to LB. Like the Telco case, we saw an increase in complexity as a result of the Bank merger. Where in the Telco case, we saw complexity initially emerge due to choices made at the level of infrastructure, the application level was the crucial one in the Bank case. Hence, this will be discussed first.

4.2.1 Application and application portfolio complexity dynamics

At the high level of abstraction that is typical during the strategic preparation phase of an M&A, a consolidation IIS was planned: the existing unchanged LB applications would constitute the full post-merger IS application architecture. Although LB did deliver commercial real estate financing services, both the scale and the complexity of these services were not comparable to what REB offered, as will be further explained in the next section. Not identifying these crucial differences in scale and complexity between the services provided by the two banks, led to complications during execution.

"*So, for LB, they had to absorb [very complex] client portfolio structures, for which they did not have the required functionality nor business processes*."

Ultimately, the decision was made to enhance the LB applications to support these more complex contract structures, increasing the complexity of the applications. This decision implied introducing elements of transformation into the IIS, as the functionality of the LB application was enriched. Like the Telco case, we see efforts toward complexity reduction at the application portfolio level leading to complexity increase within specific applications.

Further complexity increases were observed in the applications used to migrate the client data from REB to LB. For example, the Customer Due Diligence (CDD) process of LB was considerably stricter than REB's process. Many of the REB customer records did not meet LB's CDD criteria, triggering a significant increase in complexity in the migration application during the execution:

Then the [representatives of] LB said, "however you arrange it, no customer data flows through the CDD unchecked. Because that [unchecked flow] was the initial idea. Just automatically tick all CDD boxes [in the software] and you're done. Unfortunately. Yes, this decision implied adapting our entire migration software.

A further unplanned increase in the complexity of the migration application resulted from the fact that the transfer of clients impacted the financial administration of both organizations involved. Contracts in the REB applications had to be terminated administratively, and to be continued in the LB application. This required the development of additional software. This migration application itself added to the application portfolio complexity, but as the following quote illustrates, this increase was temporary in nature, as the migration software would be decommissioned after the migration.

By the way, that whole extraction and transformation tooling will soon no longer be of any use to them because the source system that is currently used at REB will no longer exist and that transformation tooling that is very specifically aimed at the REB environment's transformation to LB.

4.2.2 Data complexity dynamics

As mentioned before, the size of REB and LB operations, and their business models, were very different. This difference was reflected in the databases and the data structures. At REB, private persons could finance commercial real estate, whereas at LB, a private person cannot be a primary contractor. As one consultant said: "*Yes, both are banks, but one finances objects while the other finances customers*."

During the preparation phase the impact of this difference was not considered. During the migration preparation it was found that a specific contracting structure ("compensation merit"), was used at REB*,* allowing significant flexibility in financing real estate by commercial parties as well as by private persons. However, at LB, a commercial loan had to be associated with a single commercial party. Because of these business rules, the LB application lacked the data structures required to support these REB-specific financing structures. The higher complexity of the contracting structure in REB was also reflected in the number of documents required for a contract. This crucial information and the consequences were missed during the preparation phase. As one interviewee illustrated:

"That's just there in the system, but a big difference is that we sometimes had files with a thousand documents or more [At REB] *- at LB that doesn't happen, 100 is already very large. So that's for LB, they must absorb a portfolio that they are not equipped for basically"*

Addressing this issue did not only increase the complexity of the application, it also increased data complexity. To overcome the problems associated with compensation merit, the project team decided to create fictitious companies that would be used to 'anchor' the private person and their loans. An additional data structure was created tying private persons to their commercial loans. This solution also required the re-design of tools. and the introduction of additional functionality to automatically create and add this 'extra' layer of fictitious companies for all customers.

Yet another unanticipated complexity increase, due to incompatibilities between REB and LB applications in dealing with email attachments, was discovered during the data migration from REB to LB. As one interviewee explains:

[…] at REB we had emails in the files including attachments. At LB a policy existed prescribing that no attachments should be in e-mails, and that attachments should be stored in the DMS. However, for us it was [technically] impossible to extract the attachments from the files and store them in the correct way in the DMS. LB insisted on compliance with their policy for a very long time, but in the end allowed a deviation from that policy - e-mails and attachment were stored combined in the LB application."

In this final example one could argue that a combination IIS was added to the mix, as the REB and LB solutions for email attachments were both kept. There is even an element of transformation, because new functionality and data structures were introduced into the LB application.

4.2.3 Infrastructure complexity dynamics

For the REB and LB merger a consolidation IIS was planned for the infrastructure, and that strategy was essentially also executed. A key difference with the Telco case is the absence of anything like an OSS. A further difference is that this M&A implied the closing of all the REB offices which ended the need for the generic REB IS infrastructure. Still, even in this straightforward case, the IIS for the infrastructure was altered during execution. In this case the IS infrastructure of REB was kept operational for a longer time, because of a decision to only migrate the large customers to LB and sell the remaining portfolio of customer to another party. The infrastructure was kept operational until the remaining portfolio of customers was sold. The implication is that the IS complexity for infrastructure remained higher than planned waiting for the finalization of the sale. So, applying a strict interpretation of the IIS definition, this was a consolidation IIS temporarily combined with a co-existence IIS and a higher than planned IS complexity during that period. This was only temporary, and ultimately, after the sales process had been completed, the complexity of the IS infrastructure was not significantly affected by the integration.

5 Discussion

Our aim in this paper was to improve our understanding of the dynamics of IS complexity and the factors that impact it, during the execution of a consolidation IIS. Table 2 summarizes our findings and illustrates that across both cases the realized complexity was higher compared to the planned IS architecture complexity – i.e., the level of complexity implied by the consolidation IIS. Furthermore, our analysis points towards 3 factors that contributed to the difference between planned and realized complexity. Each of these factors leads to the introduction of one or more architectural components that account for the higher than planned complexity.

Factor 1: Temporary measures to facilitate the execution of the IIS. In both cases we find that temporary applications were introduced to facilitate the data migration associated with the consolidation of systems. Although this addition was temporary in nature, it led to an increase of the complexity of the application portfolio during the execution of the IIS. In both cases we found these temporary applications were not decommissioned at the end of the IIS execution and remained in existence. So, in the end, these measures appear to lead to a structural increase in complexity.

	Telco Case			**Bank Case**		
	Realized Com-plexity	Planned IIS	Executed IIS	Realized Com-plexity	Planned IIS	Executed IIS
Appli-cations portfolio	Higher	Consolidation	Consolidation, Coexistence, Combination Transformation	Higher	Consolidation Transformation	Consolidation
Appli-cations	Higher	N. A	Combination Transformation	Higher	NA	Transformation
Data	Higher	Consolidation/ Combination	Consolidation, Coexistence, Combination	Higher	Consolidation / Combination (implicit)	Consolidation, Coexistence, Combination, Transformation
Infra-structure	Higher	Consolidation	Consolidation Coexistence, Transformation	Com-parable	Consolidation	Consolidation

Table 2. Realized vs planned complexity in the two case studies.

Factor 2: Increasingly emergent IIS execution. In both cases we observed that business processes and services were not fully replaced by those of the buyer, as implied by the absorption BIS. Consequently, we saw an emergent execution of the IIS for the application portfolio, where elements of coexistence, combination and even some aspects of transformation were added to the consolidation IIS. In the Telco case, we also saw an increase in complexity and the combination of multiple IIS for the infrastructure domain. This increasingly emergent IIS execution resulted from the necessity to address practical problems associated with an underestimation of differences between the firms, and a consequent underestimation of the IS architecture's requisite complexity. Initially, synergies were to be achieved by removing assumed duplicate applications and infrastructure. During execution, many decisions were made from a practical perspective, focusing on securing envisioned synergies and value or on addressing operational problems. During emergent execution, complexity increased through (1) additional applications and interfaces to facilitate continued business operations in a combined application portfolio, and (2) delayed decommissioning of replaced applications, because the required IS staff was allocated to other activities.

Factor 3: Supporting business requirements: Finally, increased IS complexity was found to partially stem from supporting business requirements – which existed pre-merger but were not recognized. The difference between the services of the REB and the LB are a typical example which resulted in both more complex applications and more complex data structures. The additional complexity is driven by technological necessities, business requirements and value realization. From the case study two architectural components emerge that contribute to increased IS complexity: (1) new application functionality because of innovation requirements from the business, and (2) more complex data structures to support more complex products or services.

5.1 Theoretical implications

Our study contributes to theory concerning IS architecture complexity dynamics in M&A by identifying three factors that impact the dynamics of IS architecture complexity. We further contribute by enhancing understanding of the concept of IS architecture complexity dynamics by showing that complexity may be reduced at one level of observation while being increased at a more detailed level of observation. Finally we find that complexity reduction through a consolidation IIS is limited, due to its inherent contradiction to the principle of requisite complexity

5.1.1 Factors that impact IS architecture complexity

We identified three factors that contribute to increased IS complexity while executing the IIS. Each of these factors requires the introduction of specific architectural components. Although these components are valuable as they help the organization deal with various challenges, surprises and unforeseen complications, they tend to increase the complexity of the post-merger IS architecture. As different integration strategies are being combined, the high-level complexity reduction plans implied in the planned consolidation strategy are found to be increasingly impossible to realize. New components and relationships are being introduced to accommodate for issues confronted during strategy execution, often in short term cycles favouring short term value over longer term complexity reduction.

5.1.2 Requisite complexity vs consolidation IIS

Underestimation of the requisite IS complexity during the preparation phase of the M&A was found to be an important factor. Incomplete data gathering pre-merger, due to legal, timing and other practical constraints, contributes to this underestimation, as well as wishful thinking by those responsible for the integration strategies. For instance, the information that can be shared on strategy and operations is necessarily limited the pre-merger phase, taking into account the risk that the merger may fail. To compensate for this lack of data, strategic level assumptions about business and IS similarity are formulated – in our case studies, based on an overly optimistic estimation of the integration possibilities. In both cases, these assumptions were insufficiently validated prior to starting IIS execution. In the Telco case we see that assumptions were formulated starting from an industry reference model, eTOM, and that both firms were compared against the reference model, but not against each other. In the Bank case we see a similar pattern: both organizations were compared since both finance business real estate but whether both firms do that in a similar way was not validated. Failing to recognize these differences, which are mostly essential for the support of business processes, customers, products and services, led to an underestimation of the requisite IS architecture complexity.

5.1.3 Conceptualizing IS complexity dynamics

Building on established definitions of IS complexity we conceptualized IS architecture complexity as the *quantity* and *variety* of system *elements* and the *relationships* between them, as well as the *rate of change* in these elements and relationships. Our study adds to this conceptualization by showing that complexity dynamics are different at distinct levels of observation. For instance, complexity may be reduced at a high level of observation, for example through decommissioning of applications, while at the same time it increases at a more detailed level of observation - for example through the introduction of new functionality within applications. This implies that, for a better understanding of IS architecture complexity, we need to incorporate the complexity of the different components and interfaces themselves. This would address the observation that moving application functionality from a consolidated application to its replacement reduces complexity at one level and increases it at the next level.

Furthermore, our findings indicate that IS applications that are not decommissioned but are no longer actively used do not contribute to IS architecture complexity in the same way that actively used applications do. Such applications are an example of “architectural waste”, rather than an increase in requisite complexity (Schmidt, 2015). Consequently, we propose to also include the operational usage of components in the conceptualization of IS architecture complexity. In other words, IS architecture complexity concerns (1) the quantity and variety of operationally used components and the relationships between them, (2) at different levels of observation (from the architecture as a whole to the most granular level of components and relationships), as well as (3) the uncertain, unpredictable, and often ambiguous nature and rate of change of these components and relationships.

5.2 Practical Implications

A first implication for practice concerns the importance of considering both planned and emergent integration strategies in analyzing the value contribution of IS integration in M&A. The increasing level of detail at which IS architecture complexity is considered when moving from strategy to execution almost inherently means that requisite complexity is found to exceed what was planned at the strategy level. Consequently, "sticking to the plan" (strictly adhering to the strategic choices made) becomes counterproductive as it implies that the more detailed information is ignored, which triggers specific integration failure mechanisms (Baker and Niederman, 2014; Henningsson and Kettinger, 2016). For practice this means that those responsible for IS integration should assume that not all relevant details are known during strategy formulation. As a consequence, there should be room for emergent strategizing – allowing for activities that may not be in line with what was planned, but that do enable the integration to progress. Such emergent strategies enable employees to deal with the uncertainty, surprises and unexpected complications they are confronted with during the operational integration.

Our study also emphasizes the importance of validating pre-merger assumptions in terms of the feasibility of a consolidation strategy. In both case studies these assumptions were not validated prior to execution, underestimating the post-merger IS complexity at all levels of detail, and leading to operational problems and challenges in executing the IIS. On the one hand, this means that such assumptions should be validated more thoroughly before the start of the integration. On the other hands, acknowledging that there are always factors that complicate this validation means that organizations should create slack in IIS execution, allowing for improvisations that address such problems and challenges.

5.3 Limitations and Further Research

The main limitation of our study concerns the specific focus of our case studies. These cases only represent a very specific (though common) combination of BIS (absorption) and IIS (consolidation). Other BIS-IIS combinations may exhibit different IS complexity dynamics. Furthermore, in both cases we identify factors that lead to a transition from planned IIS execution to emergent IIS execution, but these factors may well be case-specific. Further research is required for a more complete understanding of the factors that cause a change from planned to emergent IIS execution.

Furthermore, complexity dynamics are likely to be impacted by organizational characteristics. For the Telco case, the infrastructure is a key asset which has a major influence on the complexity dynamics. For the Bank it is the complexity of the products and services and resulting contracts that exert the major influence on the complexity dynamics. Further research would be required to validate our findings for other industries.

Our study appears to confirm the importance of requisite complexity, which predicts a limited potential to reduce IS architecture complexity in M&A. This conflicts with a consolidation strategy's focus on reducing complexity. Additional research would be required to provide a deeper insight into the paradoxical relationship between the principle of requisite complexity on the one hand, and a consolidation IIS that predicts significant synergy potential through complexity reduction on the other.

References

Baker, E.W. and Niederman, F. (2014), "Integrating the IS functions after mergers and acquisitions: Analyzing business-IT alignment", *Journal of Strategic Information Systems*, Elsevier B.V., Vol. 23 No. 2, pp. 112–127.

Batelaan, M. and Veltman, J. (2002), "Vijf mythes over postfusie ICT- integratie", *Management & Informatie*, Vol. 5, pp. 50–58.

Beese, J., Aier, S., Haki, K. and Winter, R. (2022), "The impact of enterprise architecture management on information systems architecture complexity", *European Journal of Information Systems*, pp. 1–21.

Beese, J., Aier, S., Haki, M.K. and Aleatrati Khosroshahi, P. (2016), "Drivers and effects of information systems architecture complexity: A mixed-methods study", *24th European Conference on Information Systems, ECIS 2016*, available at: https://aisel.aisnet.org/ecis2016_rp/74

Benbasat, I., Goldstein, D.K., Mead, M. and School, H.B. (1987), "The Case Research Strategy in Studies of Information Systems", *Source: MIS Quarterly*, Management Information Systems Research Center, Vol. 11 No. 3, pp. 369–386.

Benbya, H. and McKelvey, B. (2006), "Using coevolutionary and complexity theories to improve IS alignment: A multi-level approach", *Journal of Information Technology*, Vol. 21 No. 4, pp. 284–298.

Benitez, J., Ray, G., and Henseler, J. (2018), "Impact of information technology infrastructure flexibility on mergers and acquisitions", *MIS Quarterly*, Vol. 42 No. 1, pp. 25-43.

Boisot, M. and McKelvey, B. (2011), "Connectivity, extremes, and adaptation: A power-law perspective of organizational effectiveness", *Journal of Management Inquiry*, SAGE PublicationsSage CA: Los Angeles, CA, Vol. 20 No. 2, pp. 119–133.

Buck-Lew, M., Wardle, C.E. and Pliskin, N. (1992), "Accounting for information technology in corporate acquisitions", *Information & Management*, Elsevier, Vol. 22 No. 6, pp. 363–369.

Busquets, J. (2015), "Discovery paths: Exploring emergence and IT evolutionary design in cross-border M and As. Analysing grupo Santander's acquisition of abbey (2004-2009)", *European Journal of Information Systems*, Palgrave Macmillan Ltd.,

Chatterjee, S. (1986), "Types of synergy and economic value: The impact of acquisitions on merging and rival firms", *Strategic Management Journal*, Wiley Online Library, Vol. 7 No. 2, pp. 119–139.

Giacomazzi, F., Panella, C., Pernici, B. and Sansoni, M. (1997), "Information systems integration in mergers and acquisitions: A normative model", *Information & Management*, Elsevier, Vol. 32 No. 6, pp. 289–302.

Cao, C., Ray, G., Subramani, M. and Gupta, A. (2022a), " Enterprise Systems and the Likelihood of Horizontal, Vertical, and Conglomerate Mergers and Acquisitions", *MIS Quarterly*, Vol. 46 No. 2, pp.1227-1242.

Cao, C., Ray, G., Subramani, M. and Gupta, A. (2022b). "Enterprise Systems and M&A Outcomes for Acquirers and Targets", *MIS Quarterly,* Vol. 46 No.3., pp.1295-1322.

Gregor, S. (2006), "The nature of theory in information systems", *MIS Quarterly*, Vol. 30 No. 3, pp. 611–642.

Haki, K., Beese, J., Aier, S. and Winter, R. (2020), "The evolution of information systems architecture: An agent-based simulation model", *MIS Quarterly: Management Information Systems*, Vol. 44 No. 1, pp. 155–184.

Haspeslagh, P.C. and Jemison, D.B. (1991), *Managing Acquisitions: Creating Value Through Corporate Renewal*, The Free Press.

Henningsson, S. and Kettinger, W.J. (2016), "Understanding Information Systems Integration Deficiencies in Mergers and Acquisitions: A Configurational Perspective", *Journal of*

Management Information Systems, available at:https://doi.org/10.1080/07421222.2016.1267516.

Henningsson, S. and Yetton, P. (2013), *IT-Based Value Creation in Serial Acquisitions*, *The 13th European Academy of Management Conference (EURAM 2013)*.

Henningsson, S., Yetton, P.W. and Wynne, P.J. (2018), "A review of information system integration in mergers and acquisitions", *Journal of Information Technology*, Vol. 33 No. 4, pp. 255–303.

Hitt, M.A.A., King, D., Krishnan, H., Makri, M., Schijven, M., Shimizu, K. and Zhu, H. (2009), "Mergers and acquisitions: Overcoming pitfalls, building synergy, and creating value", *Business Horizons*, Vol. 52 No. 6, pp. 523–529.

Johnston, K.D. and Yetton, P.W. (1996), "Integrating information technology divisions in a bank merger Fit, compatibility and models of change", *Journal of Strategic Information Systems*, Vol. 5 No. 3, pp. 189–211.

Klein, H.K. and Myers, M.D. (1999), "A Set of Principles for Conducting and Evaluating Interpretive Field Studies in Information Systems", *MIS Quarterly*, Vol. 23 No. 1, p. 67.

Lubatkin, M. (1983), "Mergers and the Performance of the Acquiring Firm.", *Academy of Management Review*, Vol. 8 No. 2, pp. 218–225.

McKiernan, P. and Merali, Y. (1995), "Integrating information systems after a merger", *Long Range Planning*, Elsevier, Vol. 28 No. 4, pp. 4–5.

Mehta, M. and Hirschheim, R. (2007), "Strategic Alignment In Mergers And Acquisitions: Theorizing IS Integration Decision making.", *Journal of the Association for Information Systems*, Vol. 8 No. 3, pp. 143–174.

Miles, M.B., Huberman, A.M. and Saldaña, J. (2014), *Qualitative Data Analysis. A Methods Sourcebook*, *Qualitative Data Analysis. A Methods Sourcebook*, 3rd ed., Vol. 28, SAGE Publications, Los Angeles.

Robbins, S.S. and Stylianou, A.C. (1999), "Post-merger systems integration: the impact on IS capabilities", *Information & Management*, Vol. 36 No. 4, pp. 205–212.

Schmidt, C. (2015), "Business Architecture Quantified: How to Measure Business Complexity", Springer, Cham, pp. 243–268.

Schneider, A., Zec, M. and Matthes, F. (2014), "Adopting Notions of Complexity for Enterprise Architecture Management", *AMCIS 2014 Proceedings*.

Seddon, P.B., Reynolds, P. and Willcocks, L.P. (2010), "Post-Merger IT Integration: A Comparison of Two Case Studies", *PACIS 2010 Proceedings*, pp. 1076–1087.

Sommer, S.C. and Loch, C.H. (2004), "Selectionism and learning in projects with complexity and unforeseeable uncertainty", *Management Science*, Vol. 50 No. 10, pp. 1334–1347.

Tanriverdi, H., Roumani, Y. and Nwankpa, J. (2019), "Structural Complexity and Data Breach Risk", *ICIS 20219 Proceedings*, pp. 1–17.

Tiwana, A. and Konsynski, B. (2010). "Complementarities between organizational IT architecture and governance structure." *Information Systems Research,* 21(2), 288-304.

TM Forum. (2014), "Business Process Framework - eTOM - Overview - TM Forum", available at: https://www.tmforum.org/business-process-framework/.

Widjaja, T. and Gregory, R.W. (2020), "Monitoring the complexity of IT architectures: Design principles and an IT artifact", *Journal of the Association for Information Systems*, Vol. 21 No. 3, pp. 664–694.

Wijnhoven, F., Spil, T., Stegwee, R. and Tjang A Fa, R. (2006), "Post-merger IT integration strategies: An IT alignment perspective", *Journal of Strategic Information Systems*, Vol. 15 No. 1, pp. 5–28.

Xia, W. and Lee, G. (2005). "Complexity of Information Systems Development Projects: Conceptualization and Measurement Development." *Journal of Management Information Systems*, 22(1), 45-83.

Yetton, P.W., Henningsson, S., Böhm, M., Leimeister, J.M. and Krcmar, H. (2022), “How IT carve-out project complexity influences divestor performance in M&As”, *European Journal of Information Systems*, available at:https://doi.org/10.1080/0960085X.2022.2085201.