TAMPERE UNIVERSITY
OF APPLIED SCIENCES

SAAS APPLICATION INTEGRATION CHALLENGES

Tiina Kaikkonen

Master’s thesis
February 2019
Master’s Degree in Business Administration
International Business Management

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Master's Degree in Business Adminstration
International Business Management

KAIKKONEN, TIINA:
SaaS Application Integration Challenges

Master's thesis 69 pages, appendices 5 pages
February 2019

Cloud services are all the time growing method to provide new services to businesses to gain benefit by subscribing the service instead of investing to it. Many on-premise solutions are replaced with new software as a service model where application is running in a supplier datacenter having scalability and elasticity and without any need for customer's own resources. An application can be used with any device connected to the internet. Challenges are appearing when integration from a cloud application is needed to several on-premise or other systems. Each system is having own processes, practises and integration layers.

The purpose of the thesis was to identify pitfalls in the Company X and provide best practises and checklists for project management to support future SaaS application integration implementation projects. Two large projects were used as a reference. Issues from these projects were investigated, categorized and prioritized. Used research method was action research where team proposed new solutions to the issue items, and those were partially executed and tested in the following iteration rounds. Study was concentrating in the case company's findings that were related to processes and practicalities, technical issues were left out from the scope. In addition, processes or practises which were working correctly without any issues were left out. Study find out several items to be developed in the processes, IT design, data management, roles and responsibilities as well as project and vendor management.

During iteration rounds best practises and checklist based on analyzed findings were collected, and these were presented as a final outcome. All confidential material has been removed from final thesis.

Key words: SaaS application integration, processes, data management

CONTENTS

ABBREVIATIONS AND TERMS

AMS	Application Management Services
BDD	Business Driven Development
CAGR	Compound Annual Growth Rate
COBIT	Control Objectives for Information and related Technology
DAMA	Data Management
ERP	Enterprise Resource Planning
ESB	Enterprise Service Bus
GDPR	General Data Protection Regulation
IaaS	Infrastructure as a Service
ITIL	Information Technology Infrastructure Library
ITSM	IT Service Management
PaaS	Platform as a Service
RACI	Responsibility Assignment Matrix
SaaS	Software as a Service
SAM	Strategic Alignment Model
SLA	Service Level Agreement
TCO	Total Cost of Ownership

1 INTRODUCTION

The purpose of this chapter is to give background information to the thesis topic ‘SaaS application integration challenges’. Thesis case company is an international corporation where integrations are playing big role to get all different IT systems working together as expected. Introduction chapter will describe research topic where also background information for the study is explained, after background information the research questions and thesis structure are presented.

1.1 Research topic

Traditional method to fulfil businesses’ software requirements has been on-premises software installation. When utilizing this method, a software application is installed and operated from companies’ in-house servers and computing infrastructure. Security, availability and overall management of these applications have been on companies’ own responsibility. To avoid or minimize these responsibilities and get well prepared updates to applications many companies have changed their approach more and more towards cloud computing.

Cloud computing is used as a general term for the delivery of computed services – servers, storage, databases, networking, software over the internet (‘the cloud’). According to Wang, Ranjan, Chen and Benatallah (2017) cloud computing key features are Agility, Location independence, Multi-Tenancy, Reliability, Scalability and Maintenance. There are several different service models provided as cloud service, but broadly those are classified into three main services Software as a Service (SaaS) – offers ready business applications, Infrastructure as a service (IaaS) – offers storage and compute resources and Platform as a service (PaaS) – offers development environment. (Wang et al 2017, 1.1, 1.5)

Cloud computing is the next stage in evolution of the Internet where services can be delivered to the customer whenever and wherever needed. This will offer also possibility to treat the cloud system as a pool of resourcing instead of setting up independent environment built in house. Cloud deployments can be categorized for three different types: Public Cloud, Private Cloud or Hybrid Cloud (Hurwitz, Bloor, Kaufman & Halper 2009, 8).

Marinescu (2015, 2) presented overall picture about different aspects related to cloud computing as seen in figure 1. The model includes following aspects: delivery models, deployment models, infrastructure, resources and defining attributes. All these aspects are not part of this thesis, focus is in the SaaS application delivery model and especially application integration to the other systems. Further, technical aspects will be out of the scope of this thesis to be able to concentrate on collaboration and challenges between actual business and IT.

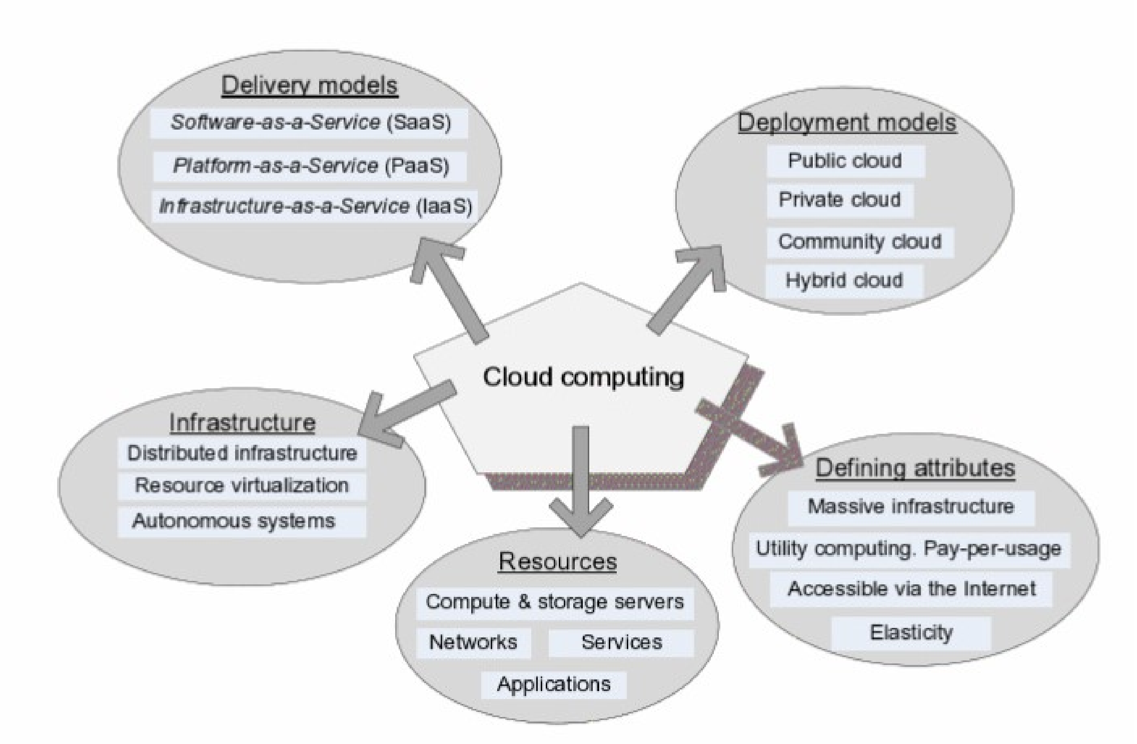


FIGURE 1. Cloud computing aspects (Marinescu 2015)

Even though during the application purchase negotiations it is said about easiness to implement cloud services, reality is not always so bright, there are lot of challenges especially in large companies where own practises and internal processes are already in place and end-users are not so willing to change the current way-of-working. When planning to implement cloud application, it would be important to recognize the nature of cloud application is to gain benefits from large customer basis, functionalities are not built to support approaches where different sites or countries inside one corporation have different rules to be implemented.

This thesis is provided to an international company having production in 13 countries and in 6 continents, totally the group has more than 100 companies and employees in 45 countries. IT services are centralized under one function, which is providing services for businesses and companies throughout the whole group. Service is including also responsibility for all integrations from and to SaaS applications. This thesis is concentrating to Global Functions related enhancements where key customers in IT application implementations in are mainly intercompany customers and service providers are external companies.

Instead of an on-premises solutions, cloud applications are currently all the time growing approach to fulfil case company's business needs. This thesis is focusing to investigate best possible way to integrate a new cloud application to other existing on-premise or cloud applications and give special consideration to the recognized integration challenges from the previous projects. Focus area will be in the Saas (Software as a Service) applications and approach is more business than technical point of view. Other two cloud products PaaS and IaaS will not be part of this study to be able to concentrate challenges with SaaS application integrations.

Integrating cloud based and on-premise or two different cloud-based applications can be very demanding exercise. In practise, integrations in the case company have been seen very challenging in multi-application environment where different kind of requirements from different systems occur. In addition, there are integration needs for inbound and outbound interfaces with different local requirements. Especially application or software implementations needed for the global usage are quite challenging due to different business cultures, local legal requirements and practises, ways of working and replacing own local applications in different parts of the world. Integration and customization issues are quite commonly overlooked until problems are raised and this is often too late.

A master data and its reliability is critical aspect for all integrations. The maintenance of master data type of elements should be done only in one application and data transfers to the other applications or databases are needed. Without integrations from the master system, there would be similar master data updated in different systems and with each environment using local requirements and own terminology, and where data might have different meaning than in the original master system. This would result to duplicate and different-aged data, which leads to inaccurate reports and damaged data discrepancies. Also, end users might do incorrect interpretations due to different terminology for the equivalent master data element. Final target should always be that end users need to be sure that data is correct and accurate in the target system and used terminology equals between all environments.

Taking full advantage from the SaaS applications would require standard interfaces and practises. There have been challenges in case company when implementing global level SaaS applications and trying to integrate these into existing enterprise level systems or package applications. Additional challenge in case company has been with business processes where business or function has wanted to keep nearly all processes as-is and has not been so willing to take pre-defined SaaS application processes or procedures into use. All possible scenarios or processes which were supported in earlier solution were no longer supported with the new SaaS application, but process harmonization was needed.

Many companies' strategies, as also in the case company, are suggesting to implement the cloud application to avoid own infrastructure and costs it is causing. Reality has anyway shown that business expectations and requirements are not always so easy to get realized with standard application. Quite often during the project phase it is said that nothing will be changed from integration point of view and everything can remain as they are but unfortunately this is not the whole truth due to changing business processes and technical capabilities. Same comment is valid to both inbound and outbound interfaces.

One challenge in the case company is caused because business users preconception is that new SaaS application can work exactly similar way as previous, maybe large and very customized on-premise system has been working. Fundamentally cloud applications are meant for wider customer group and a customization might impact negatively to whole basic processes. From business point of view adaption of the SaaS application could mean changes in previous processes and all required customizations are not possible to implement with standard configuration.

In addition to the application process changes, also master data is playing a big role. It is important that ownership for each master data element is recognized and clear owner is nominated, and responsibilities related to it are recognized. Individual master data may be transferred to several applications and then forwarded again to third application or report. It is important that data owner knows all the flows from the beginning to the end of processes, especially in case of data privacy aspects.

In the large companies, integrations have significant influence, information from one SaaS application could be transferred to the several different applications. If data from the source system is not reliable, it could cause remarkable problems in the target systems. Requirements differ from the application to the application, and standard interfaces are not always existing. From May 2018 onwards the data privacy regulation (GDPR) was established in EU, and offences can be remarkable if legal requirements are not considered enough carefully. Before transferring any data to other systems also data privacy issues should be investigated carefully.

In the case company there has not been clear rules or guidance how above matters should be taken into consideration when implementing new SaaS service with integrations. This thesis topic is to investigate best practises how the implementation project should observe other applications business needs already in phase of planning the integrations and how best practises should be considered in actual execution phase. Technical aspect how integrations should be adopted, will not be considered in this study.

1.2 Research questions

The aim for this study is to understand better how integration related topics should be considered and how to avoid possible pitfalls in the implementation projects. Theoretical framework is explaining main elements related integrations to fulfil the research aim.

Main research question is:

- How to develop an effective approach to manage, identify and understand integration requirements towards other corporate systems when adopting new SaaS application?

Research objective to this question is to identify best practises to support project management in cases where SaaS application has several integrations.

Secondary research questions are:

- What should be considered when adopting SaaS system methodology and standard process in Corporate X business environment?
- How to identify master data and its ownership and identify responsibilities for master data owners when same data is shared in many systems?

For first question research objective is to establish supportive checklist to the project IT Lead for the items to be discussed when planning and executing the implementation. For second question research objective is to consider and to describe basic rules to identify new master data element, provide rules for identifying master data and define checklist for integration related responsibilities for master data.

1.3 Thesis structure

Chapter 1 focuses on general overview and introduction of the thesis subject and information how the scope is restricted. Also, research questions and aim are described.

Chapter 2 will open the theoretical framework for this study. The focus will be on the case company's main pain points from previous projects. This chapter will provide academic approach to the study and is giving support to answer research questions.

Chapter 3 will concentrate on the research methodology used in this study. It contains strategy and method aspects. Data management and data related functions have been raised to high level due to fact that major part of challenges in integrations are related directly or indirectly to data.

Chapter 4 is including research results with the analysis. Chapter starts with explanation about data acquisition and continues with results. In this study two major projects have been used as a reference and this data acquisition opens background for those. Chapter has been dived into the sub chapters based on main finding areas.

Chapter 5 contains discussion and conclusions based on the research results and analysis. This chapter is overview of the whole study and will also include suggestions according to previous chapters. It will give also proposals how the case company can develop the process in the future.

2 THEORETICAL FRAMEWORK

The main objective of this chapter is to give theoretical background for topics that are mainly impacting to the study outcome. First sub chapter will provide theoretical background to SaaS application itself and are following with description of main elements where challenges have been occurred in the case company. In this thesis aim is not to give full picture about all possible aspects related to application integration, only the ones that are related to the case company challenges.

2.1 Software as a Service (SaaS)

SaaS software is rather rented than purchased software. Instead of buying the application and paying for periodic upgrades and maintenance, SaaS application is a subscription based and maintained in supplier’s datacentre. Application usage will happen via the internet from any location. Companies have seen this beneficial and SaaS application market is growing significantly. The market is expected to grow at a compound annual growth rate (CAGR) of 28,3 % between 2016 and 2025 (M2 Presswire 2017).

Gartner (2009) has defined five attributes to clarify cloud-based software delivery:

Service-Based: service is more important than used technology

Scalable and Elastic: service can be scaled up or down based on demand and there will be no initial setup costs

Shared: pool of resources to build economies of scale, no hardware costs, automatic updates

Metered by Use: payment plans are implied based on usage, company will pay only what is used and no hidden costs will appear

Uses Internet Technologies: service is using Internet identifiers, where application can be accessed via any internet enabled device (e.g. desktops, smart phones and tablets) and it is accessible from any location

These web-based applications are designed as centralized, shared-instance and multitenant applications. Supplier will take care of the whole service, including security, performance, availability, reliability and scalability. This approach enables a company to free the resources for more profitable requirements and ensures that correct and adequate capacity is always in place, whenever or wherever needed. This requirement is realized heavily in the case company that is an international corporation with hundreds of locations and several time zones. International company requires services running 24/7 and it will be very resource consuming if global services are provided from one on-premise location.

High availability and reliability is one of the key challenges for a cloud service provider. Customers are expecting full availability without any functional failures and this is normally written in service level agreements (SLA) with possible violation penalties. Service outages can seriously impact customer workloads and processes. Reports state that up to $ 285 million yearly have been lost due to failures with about 99,91 % availability (Snyder 2015).

There are three factors pushing adoption of a SaaS application and a cloud in general. In many cases the main reason is potential cost reduction, when own infrastructure will not be needed. Second factor is IT operational complexity, when SaaS is providing simpler way to adopt and administer an application. Third factor is related to innovation, ability to launch new services in a short time. When on-premise software development is elastic and focuses on customization where features are easily added without considering increase to total costs of ownership (TCO), cloud solution is restricting offered elasticity and due to this is saving maintenance costs as also shown in figure 2. (Bibi, Katsaros & Bozanis 2012, 87)

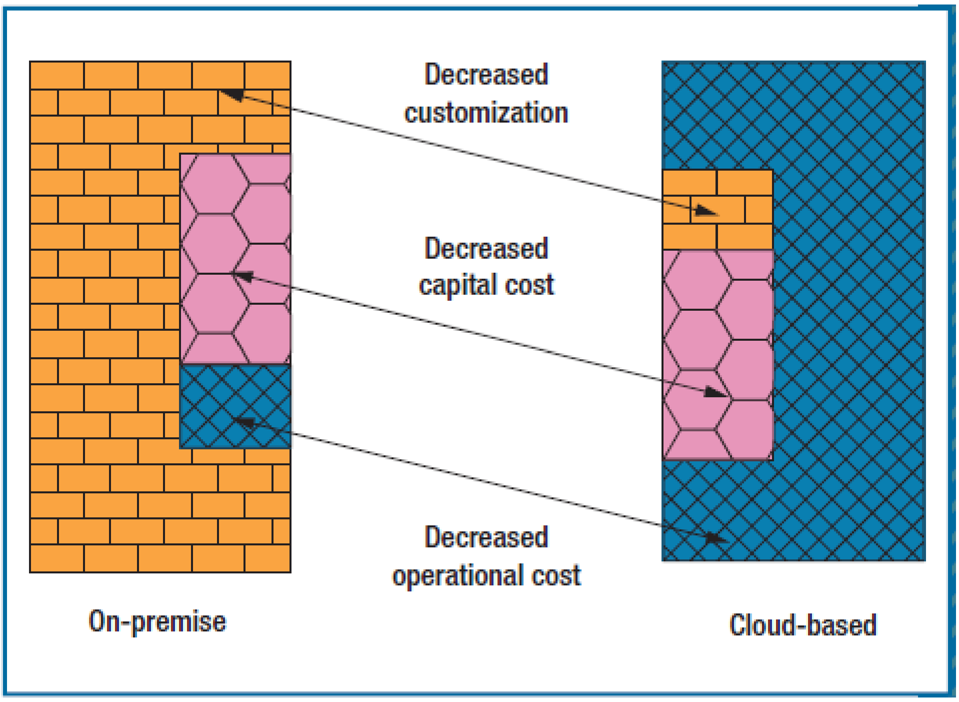


FIGURE 2: Qualitative comparison of software acquisition approaches (Bibi et al 2012).

Mainly SaaS applications are following pay-as-you-go subscription model, where charging is based on service before using it, without possibility to use more than have been paid. This approach allows low risk and quick time-to-market. In this approach there is no need to purchase separate infrastructure which also means no set-up or maintenance costs. When requirement for additional resources is raised, SaaS application is flexible to implement needed capacity changes quickly to a subscription basis.

As many employees in the case company are no longer working in the one fixed location but are travelling frequently and need access to the services regardless of time zones or actual location, mobile accesses are required. SaaS will give the capability to use application supplied by the service provider in a cloud infrastructure and which are accessible via a web browser. Customer cannot manage or control a cloud infrastructure including network, servers, operating systems, storage or application capabilities. Possible exception will be limited to the application configuration settings. (Marinescu, 2015, 2 – 13)

In multitenancy environment same codebase and instances of the application are offered for all the tenants, this will enable large economics of scale. Approach will cause from customer viewpoint limitation when individual customer cannot change each tenant's customizations. Application have some configurable components which allow doing changes to the presentation, logic and database layers without code changes. (Hai & Sakoda 2009, 257 – 264.)

2.2 Data Management

Data is an enterprise asset for organization but deriving value for data do not happen by accident, it requires intention, planning, coordination, and commitment (DMBOK2 2017, 1 / 1). Data is dynamic because it can be used for several different purposes and can be used by several people same time. In addition of inaccurate, incomplete or out of date data, it will present also risks with misuse or misunderstood information, unreliability and inappropriate use (Sebastian-Coleman 2018, 2).

To meet strategic organizational goals and have data supporting the targets, it will require committed, nominated person to ensure data management throughout the data lifecycle, from technical management to utilization. The data management professional need to have both technical and business skills to support the whole data chain. This kind of global role is missing from the case company, in some functions or individual data elements this is considered but not fully throughout the enterprise and this is causing challenges in the integrations. Also, COBIT control objective DS11 (2005, 143) control objective for data management is focusing on maintaining the completeness, accuracy, availability and protection of the data to ensure the accurate information to the business.

Data management is an IT practice, where goal is to organize and control the data resources so that it is accessible, reliable and timely available for the users. It includes suite of tools to for collecting, validating, storing, organizing, protecting, processing and maintaining the data (Kidd, 2018). Data management requires an enterprise perspective to be able to apply needed data effectively across the corporate and due to this data management and data governance are intertwined.

According to DAMA Guide (2010, 18) the mission for the data management is to meet and exceed the information needs of all the stakeholders in the terms of information availability, security and quality. Goals are to understand the information needs, ensure integrity of the data and continually improve data quality. To support achieving these goals The DAMA Wheel has been created to define ten knowledge areas for data management (figure 3). Data governance is located at the centre of the other functions since governance is required for consistency and balance between the functions. Each function is necessary for the data management, but they can be implemented in different time. Only areas where challenges are recognized in the case company, are part of this study. Main focus in this thesis is with data governance, architecture, modelling and integration.

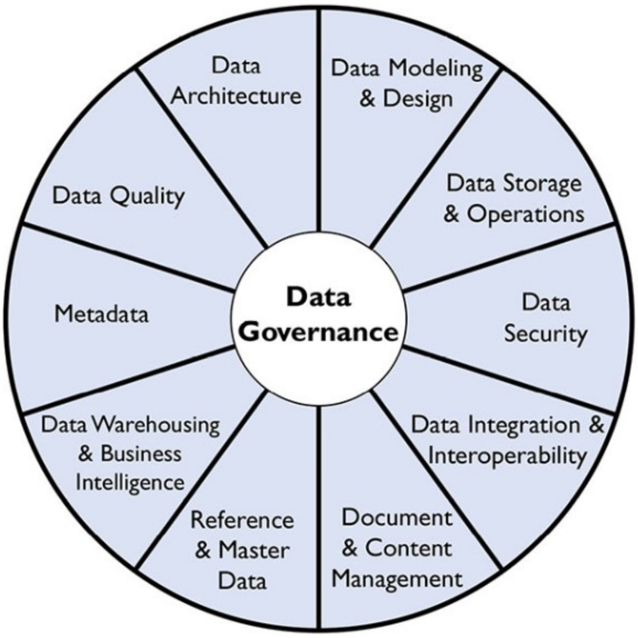

Data Architecture
Data Modeling & Design
Data Storage & Operations
Data Security
Data Integration & Interoperability
Document & Content Management
Reference & Master Data
Data Warehousing & Business Intelligence
Metadata
Data Quality
Data Governance

FIGURE 3. The DAMA Wheel (DMBOK2 2017, 1 / 3.3)

In addition of the DAMA Wheel there is another linked framework that defines components of the structure – The Environmental Factors hexacon (figure 4). Each of the DAMA Wheel functions have seven environmental elements to be considered when planning and executing the function. These elements will show the relationship between people, process and technology. In this model goals and principles are at the centre to provide guidance how to execute activities. (DMBOK2 2017, 1 / 3.3)

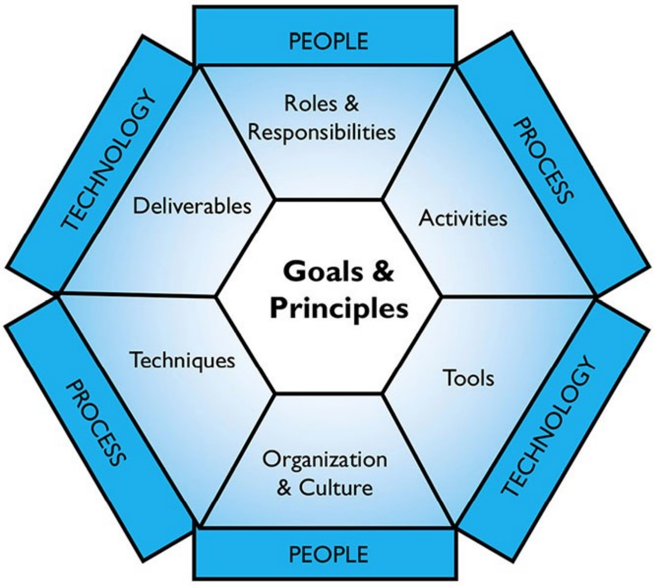


FIGURE 4. The Environmental Factors hexacon (DMBOK2 2017, 1 / 3.3)

Important aspect to the data management is data ownership where related policies and custodial responsibilities need to be clear and understood by all parties. Lack of ownership in the case company causes constant discussions about role and responsibilities when changes or data cleaning are required by the target system. Ownership means also responsibility of the data content, meaning that data is up to date and all needed cleaning practises are in place. This will ensure that all subsequent dependent processes have minimal impact to the data quality issues. Master data management process should ensure complete, consistent, up-to-date and authoritative data with high quality and which enables sharing data across functions and applications. Master data belongs to the organization, not to a particular application or department. (DMBOK2 1 / 1.2)

Data quality is crucial factor, without high quality data outcome will never be reliable. According to DMBOK (2017, 13 / 1.2) data quality principles include following aspects

- Criticality; prioritization and analysing the level of risk if data is not correct
- Lifecycle management; during the data lifecycle, data can be cleansed, transformed, merged, enhanced or aggregated (figure 5)
- Prevention; prevent data errors, not just correct the records
- Root cause analysis; often requires changes in processes and in the systems. The Shewhart cycle is problem solving ‘plan-do-check-act’ -model where improvement comes via defined set of steps. Measuring the data against standards, identify and remediate the data (figure 6).

- Governance; support the development of high quality data
- Standards-driven; measurable standards
- Objective measurement and transparency; methodology and results to be shared
- Embedded in business processes; data quality standards to be followed by business process owners
- Systematically enforced; system owner responsibility
- Connected to service levels; data quality reporting and issue management should be part of SLA

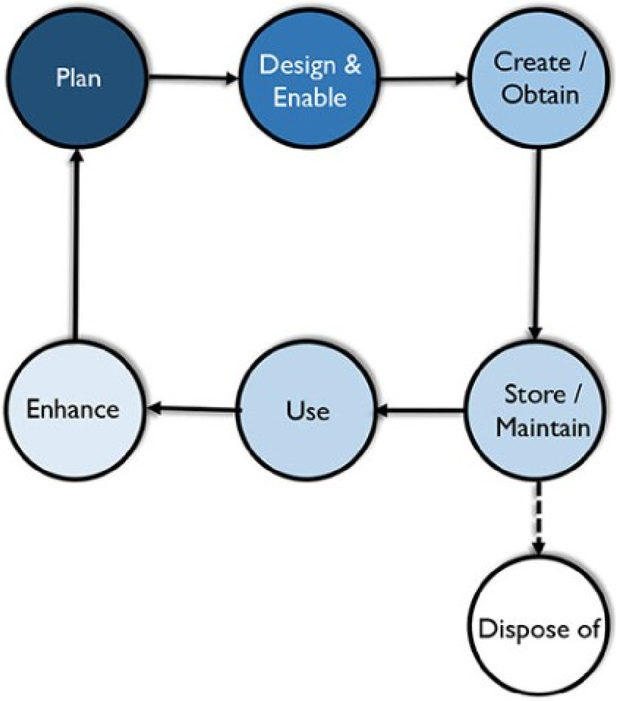


FIGURE 5. Data lifecycle key activities (DMBOK2 2017, 1 / 2.5)

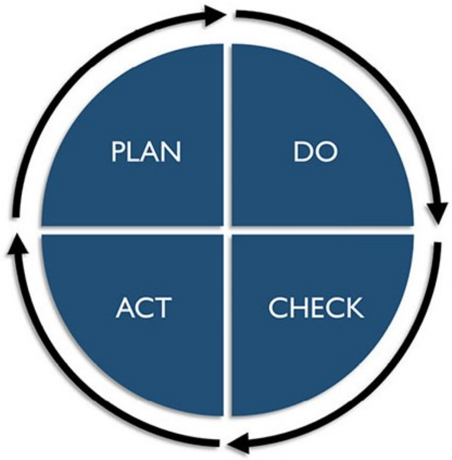


FIGURE 6. The Shewhart Chart (DMBOK2 2017, 13 / 1.3)

Master data ownership's role includes global understanding what applications are using the data and how it is used there, or will it even be transferred forward to third application or other reporting usage. Overall responsibility cannot be transferred to the target organization or application, but adequate level understanding need to be collected from the target system representatives to an actual data owner. Master data ownership includes requirement to fulfil also requirements from the target system even though particularly that is not needed in SaaS application in question.

Data handling ethics and data privacy issues belongs to the master data owner. It includes following core functionalities: manage quality and reliability, prevent misuse and control accesses. Handling of the information should not be considered only on technical perspective, because data contain information that represents people (customers, employees, vendors etc.) and this makes management also ethical. To be considered that the data environment is evolving rapidly and need constant follow-up and upkeep, and this does not mean only from legal viewpoint but also ethical reasons to guarantee data is protected and is not misused. (DMBOK2 2 / 1)

When above described ownership, roles are at least partially missing from the case company, there are several different overlapping requirements or wishes from different stakeholders. And whole corporation is lacking people or organization who will give final decision and have overall understanding of the data management and needed aspects relating to it.

2.3 Data Governance

The data governance is the strategy of the data, being in the middle of the whole data management (figure 3). The purpose of data governance is to ensure that data is managed properly, it is shared activity between business data activities and technical data management. Data governance should provide the principles, policy, processes, framework and metrics. It will guide how all other data management functions are performed. According to the Global Data Management Community's DMBOK2 (2017, 1 / 1) information should be managed as an asset, because data is a valuable asset for the company. This is not always recognized by the businesses.

As data is the heart of all transactions, it is the one of most critical issues to the business even though it is very underestimated in many companies, also in the case company. Not even all international companies have separate responsible to organize data management in the enterprise level. The Leaders Data Manifesto (2017) states: *Organizations' best opportunities for organic growth lie in data*. This means also that data management needs clear committed leadership and the involvement from the organization.

Data governance should give framework for the data management. Delez (2018) defines framework to include strategic and operational data governance levels (figure 7). Strategic data governance will ensure that the overall set-up is working, is responsible for escalations and manages the evolution required by the environment and the corporate strategy. Operational governance is translator between data requirements and data providers, responsible for frequent monitoring the compliance of the data flows and managing the constant improvements. There strategic level governance contains priorities, resources, escalation, methods, performance and improvement management. Operational data governance is responsible for enablement and control.

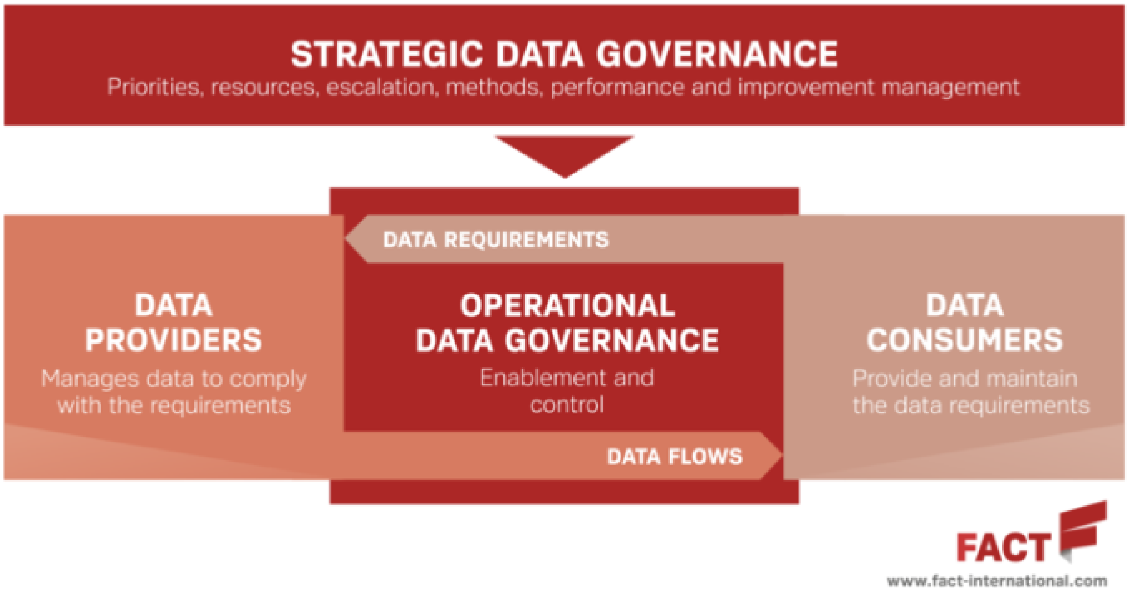

STRATEGIC DATA GOVERNANCE
Priorities, resources, escalation, methods, performance and improvement management
DATA REQUIREMENTS
DATA PROVIDERS
Manages data to comply with the requirements
OPERATIONAL DATA GOVERNANCE
Enablement and control
DATA CONSUMERS
Provide and maintain the data requirements
DATA FLOWS
FACT
www.fact-international.com

FIGURE 7. Data Governance model (Delez 2018)

Data governance will be the enabler for actual data management. Ownership for certain data subject needs to be agreed, responsibility should be in the business or function side, not IT. Additionally, data cannot be owned by the application, ownership need to be set clearly to the business or function. Producing high quality data requires cross-functional commitment and coordination. This fact needs to be considered when planning specific data element ownership, the new owner needs to understand this principle and not only secure own needs for the data. To be remembered, high quality data need organizational commitment and leadership, it will not come without any effort.

Delez (2018) has categorized data into following types: (1) Master data, which need to be as accurate and current as possible, (2) Transactional data to represent the actual way of performing business operations, (3) Reporting data which is valid only for the current state of the affairs and is very sensitive to a quality of master and transactional data and (4) Environmental intelligence data which is not collected with a clearly determined usage and can differ between different timeframes.

Business Application Research Center, BARC (2018) defines data governance aspects to include organizational (the ‘where’ and ‘who’), business (the ‘fact’) and technical (the ‘how’) views (figure 8). Data governance is ongoing, iterative process affecting in the strategic, tactical and operational levels, including coordination with different projects. Communication is needed between these levels but decision and escalation come from strategic level.

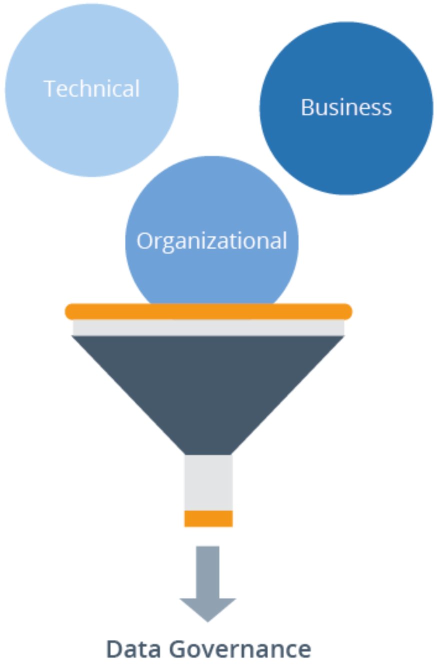


FIGURE 8. Aspects for Data Governance (Business Application Research Center 2018)

Henderson & Venkatraman (1993, 1999) derived four alignment perspectives for guiding management practises, it is called Strategic Alignment Model (SAM) and it is recognizing the changing role of IT towards more strategic partner where its role is to support and shape business strategy decisions. Model is pointing the importance of data in the center of the model and requirement for strategic collaboration between business and IT.

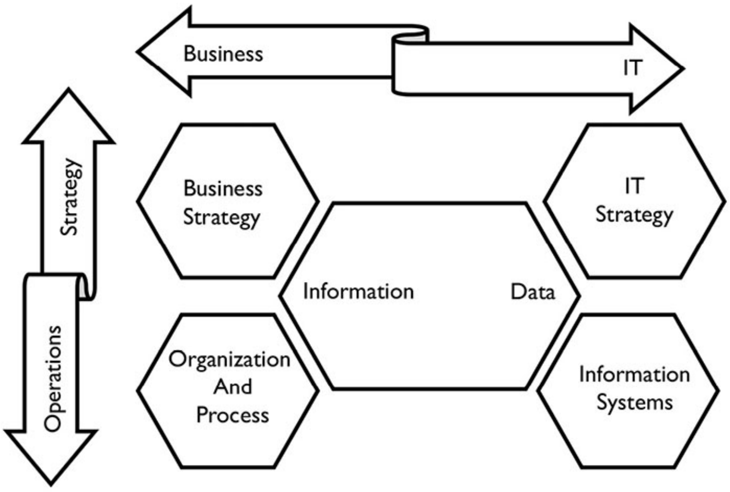


FIGURE 9. Strategic Alignment Model (Henderson & Venkatraman 1993 & 1999)

Another model, extended from SAM by Abcouwer, Maes & Truijens (1997), The Amsterdam Information Model (AIM), also called as 9-cells model (figure 10) has similar strategic perspective on business and IT alignment, but having middle vertical and horizontal layers. Additional layers are focusing on structure and tactics, including planning and architecture and where information is clearly separated from IT. (Abcouwer, Maes & Truijens 1997, according to DMBOK2 2017, 1 / 3.2).

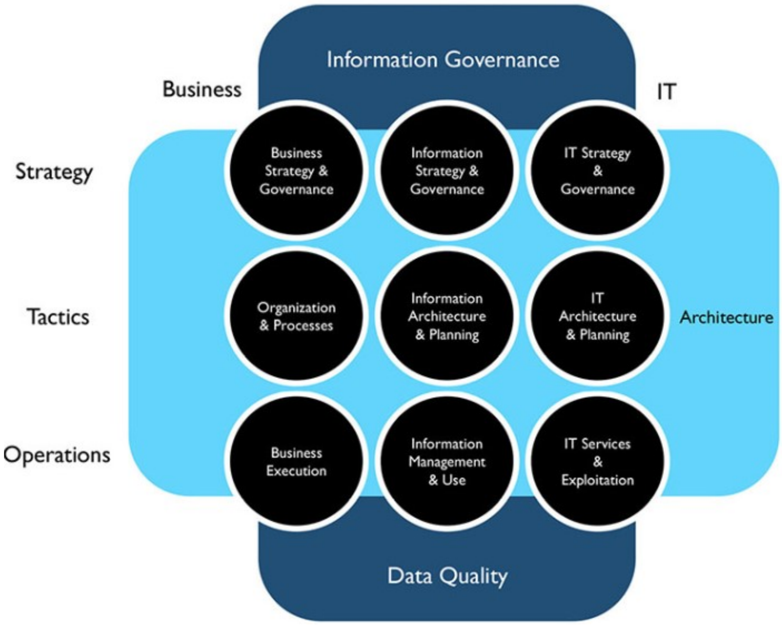

Information Governance
Business
IT
Strategy
Business Strategy & Governance
Information Strategy & Governance
IT Strategy & Governance
Tactics
Organization & Processes
Information Architecture & Planning
IT Architecture & Planning
Architecture
Operations
Business Execution
Information Management & Use
IT Services & Exploitation
Data Quality

FIGURE 10. The Amsterdam Information Model (Abcouwer, Maes & Truijens 1997)

Data modelling is needed to realize the importance of the data which needs to be clean and have high quality. According to DAMA – The Data Management Association (2017) data management activities include everything from ability to make decisions how to get the strategic value from data to the technical deployment and performance of databases.

New applications will include lot of new master data elements. Master data can be equal to the previous system, but it can also be something new occurred only due to different business approach or the master data element content might have wider meaning in the new system. Each master data has different classes and characteristics that need to be recognized and difference to previous system needs to be understood. This is especially important if there will be either data migrations to the new system or still partially different systems are using both code values.

Ensuring data quality is continuous activity and requires planning, commitment and mindset that builds quality into the processes and systems. Planning includes aspect to understand data connections and relationship between business processes and applications, and to be considered that other functions can impact data quality via integrations. Each data has a lifecycle where data may be cleansed, transformed, merged, enhanced or aggregated – process need to consider these transactions. (DMBOK2 2017, 2 / 2.5.9)

Ladley (2012) introduced the Governance V concept to clarify meaning for governance. Definition is stated as follows: *The purpose of data governance is to ensure that data is managed properly, according to policies and best practises.* (Ladley, 2012, 11). The Governance V includes in the left side **Oversight** (input to data, content life cycles, rules, policies) to ensure that data management is happening as it is supposed to. And in the right side includes **Execution**, actual data management – hands-on activities. Separation is needed between these two duties. In the bottom of the V there are activities using the data, e.g. creation, use, manipulation and disposal. (ibid.)

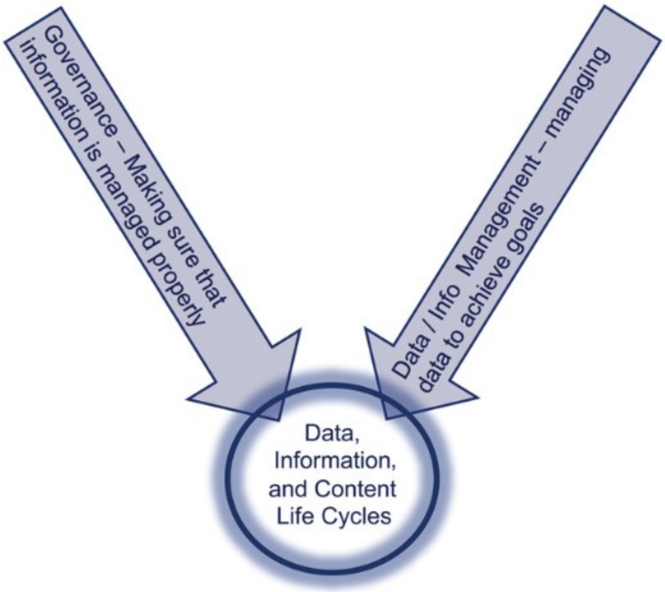

Governance – Making sure that information is managed properly
Data / Info Management – managing data to achieve goals
Data, Information, and Content Life Cycles

FIGURE 11. The Governance V (Ladley 2012)

It will be important to understand that the data governance should have clear organization that supports business strategy as well as its own cultural context. Organization should have legislative and judicial view (Do the right things) and executive view (Do things right). In addition, multiple layers should be considered to address concerns at different levels within the enterprise (local, division, programs, enterprise). (DMBOK2 2017, 1 / 1.3)

2.4 Data Architecture

As a large-scale architecture can be divided to four domains; business architecture, data architecture, application architecture and technology architecture. Business architecture will identify how to create the value, data architecture describes how the data is organized and managed, application architecture describes the structure and functionality and technology architecture describes the technology. (DMBOK2 2017, 4 / 1.3.1)

This study will concentrate only to the data architecture and only that functionality will be part of the thesis' theory. Viewpoint in the data architecture is especially data linkage to the application integrations. In the case company there are several architects who are managing their own application or infrastructure area, and, in some cases, it has been difficult to have common understanding if related integration and/or related data is touching several architects' areas and all architects want to see only own benefit. Data architecture perspectives to be considered are **outcomes** (models, definitions, data flows), **activities** (to form, deploy and fulfil intentions) and **behaviour** (collaboration, mindset, skills) (Sebastian-Coleman 2018, 6).

DMBOK2 (2017) states that data architecture's essential components are outcomes, activities and behaviour. Outcomes are artefacts; models, definitions and data flows on various levels. Activities refer to forms, deployment and fulfilling intentions and behaviour refers to collaborations, mindsets and skills among the various roles affecting data architecture. (DMBOK2 2017, 4 / 1)

Data architecture is needed to understand the framework or outline that provides guidelines for the data handling from the beginning to the end. It is critical to understand the importance of a data architecture, it will be basis for all integration activities. If structures and processes are not solid, extendable information architecture require more decisions, approvals and political battles. If solid structures are established, it will be much easier to do things correctly to support any solutions in the future. Tupper (2011, 6) defines main forces which determines the solution (figure 12): economy (cost of the components), performance, simplicity (the more components involved, the more complex process) and interfit (assembling components). Whole design is balancing of these forces.

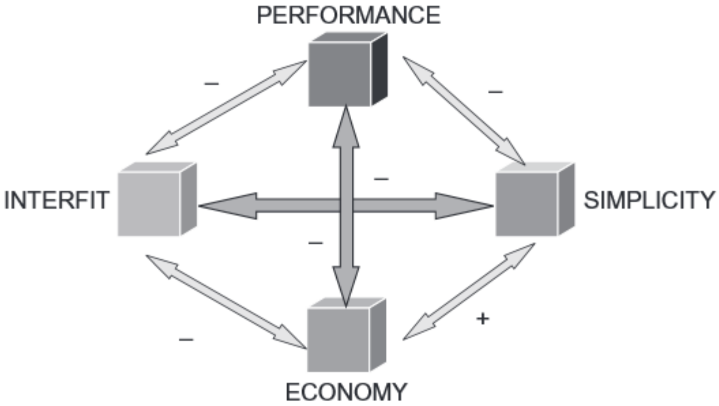


FIGURE 12. Forces on a design problem (Tupper 2011,7)

The goal of data architecture is to be a bridge between the business strategy and technology execution. Data architect should be facilitator between business and IT, acting as agent for the change, transformation and agility. The role includes ensuring business needs and system requirements are fulfilled, managing complex data and information delivery throughout the enterprise and strategically preparing organizations quickly evolve business opportunities with emerging technologies. (DMBOK2 2017, 4 / 1.1)

DMBOK2 (2017, 4 / 1.2) defines that deliverables for the data architecture are design, data flows, data value chains, data model and implementation roadmap. Design should always bring value to the organization with optimal technical footprint, operational and project efficiencies, and increased ability to use the data. When designing integrations there should be overall understanding of the whole chain, from the source system via enterprise service bus (ESB) to the target system.

2.5 Data Modeling

Data modeling is the process of discovering, analyzing and scoping data requirements and then representing and communicating these data requirements in a precise form (DMBOK 2 5 / 1). Data modeling is the core of most integration challenges. With adequate data modeling it will be possible to make it easier to handle the data quality and data-related errors. It is about increasing the communication between business owners and IT to achieve better results and avoid mis-communication.

Data modeling role has been changing due to the package software's. Because flexibility starts from the data model, core data modeling need to be done by a software provider with a very high standard to support different customer's different needs. When choosing software (on-premise or cloud application) it is important to understand business requirements and application data model to ensure data can be used for needed purposes, also taking into consideration future coming integrations and requirements from target systems.

Data models define logical inter-relationships and data flow between different data elements. It will also identify how the data is stored and retrieved. It will help represent what data is required and what format is to be used for different business processes or in different integrations. Data modeling can be done either by IT personnel or it can be done in business-driven attitude.

Simsion and Witt (2005, 8 – 10) commented that data modeling is one of the most important component on IT design. Reasons to give additional effort to data modeling are *leverage* in the sense that a small change to data model may have a major impact on the system or integration, *conciseness* when expressing systems requirements and capabilities and *data quality*, to be considered that data is valuable business asset.

Traditionally data-driven model is commonly planned from the source system point of view, without considering the whole business processes impacted as will be done in business-driven data modelling approach. Data-driven model will be done by IT based on stakeholders' requirement list. This might cause problems when start working with the data, there will come up new, specific requirements and/or business rules. (Dine 2015)

There is also Business-driven development (BDD) methodology created for developing IT solutions to directly meet the business requirements. This agile methodology starts from the business strategy, demands and objectives that are transformed to IT solution. Starting point is the business process models where these are artifacts for IT design (Tilak 2005). Main phases of this methodology are defined in figure 13.

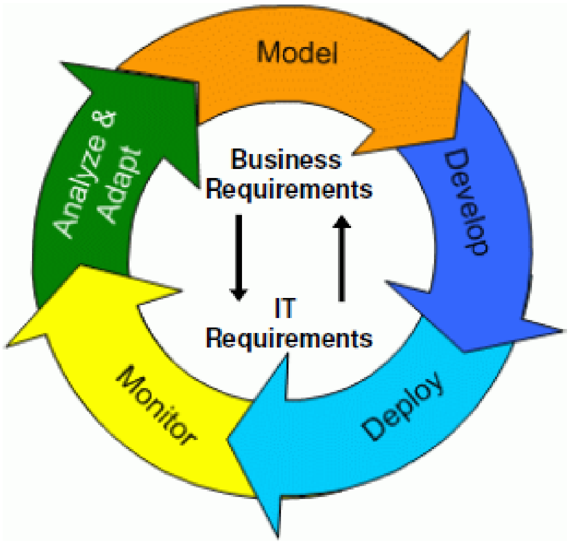


FIGURE 13. Business-driven development methodology (Koehler et al 2008)

Proper data modeling leads to lower support costs and increases the reusability possibilities by aligning current and future business requirements. Data models need a frequent maintenance, always when requirements or business processes are changing. Each change to model need to be considered from all integrations point of view. To check possible impacts to the data model should be treated as normal step in all enhancements related to integrations, even though it will not be easy to align data models between cloud and on-premise applications.

2.6 Data Integration

Core area for this thesis is data integration and recognized challenges between the source and target system. Data integration is managing the movement and consolidation of data within and between applications and organizations. Integration should provide the data in the format and timeframe needed, with lower costs and complexity, trigger automatic alerts and actions and support business intelligence, analytics, metrics, master data management and operational efficiency. (DMBOK2 2017, 8 / 1.1)

The goal of a data integration is to offer uniform access to the data sources. For reliability point of view data quality need to be in high level. Quite typical problem with the data is that the source systems are independent and do not have similar structures and the data sources do not belong to same administrative entity which can mean that data format or accesses can change when source decide to do so. (Doan, Halevy, Ives 2012, 6)

Primary driver in data integration is manage data movement efficiently which requires in large companies dedicated organization to be responsible for actual execution. Data integration includes three processes: extract, transform and load. Enterprise level design and standard tools are needed for the efficient and cost-effective solution. It also enables reuser of the code to implement compliance rules and simplify verification. (DMBOK2 2017 / 8 / 1.1)

System integrations are very often causing problems, outcome is not as expected. For these problems can be categorized to system, logical or social and administrative reasons. System reasons are technical aspects to integration, e.g. technical solution. This challenge will not be covered in this study. Logical reasons are related to data structure, format and modelling. Third reason; social reason is about data location and ownership reluctance to share the information. (Doan et al 2012, 6-8)

2.7 Service management and control environment

When implementing SaaS application and related integrations also processes and control environment need to be considered. Processes and controls are mainly company’s internal definitions and requirements, or some global methodologies or principles can be followed. Each company will decide on a management level what rules and practises are to be followed. In this chapter only relevant aspects for study in question in the case company are considered.

The case company is following fully ITIL methodology and principles in the IT Service Management. ITIL is a set of detailed practises to be implemented for IT service management to ensure fluent services to business. IT Service Management (ITSM) definition is presenting a set of specialized capabilities for providing value to the customers in the form of service (ITIL Service Strategy 2007, 15).

Controls are needed to ensure the successful testing and preparation of new applications and integrations for serving the business. In addition, controls will facilitate and support the successful execution of the daily activities and operational transactions of the IT systems. Today's business environment requires higher level of excellence to be able to survive in competitive market and IT environment, and adequate controls will ensure this outcome. (Kyriazoglou 2010, 246 – 247)

Control Objectives for Information and related Technology (COBIT) control objective AI6 (2005, 93) describes that all changes need to be formally managed in a controlled manner to assure risk mitigation and guarantee stability and integrity of a production environment. This control mitigates the risk that business development will happen without deep understanding of the ITSM processes and best practises. In addition, it mitigates the possible communication gaps between different parties. Quite commonly when application management services (AMS) are outsourced, adequate communication between AMS partner and business development is not in place without predefined controls.

Even before new SaaS application is acquired, there need to be enough information to ensure all needed business requirements are fulfilled with efficient approach. Business' functional and control requirements need to be translated into an effective and efficient design of the solution and integrations. Focus should be in identifying technically feasible and cost-effective solution. This requires clear definitions from business and technology as well as possible feasibility study. (COBIT 2005, 73)

COBIT control objective AI2.2 (2005, 78) points out the importance of detailed design where several items to be considered, e.g. requirement definition and documentation, interface definition, user interface, source data design, processing requirements, output data requirements, control, security, availability, and testing. In the case company requirement definitions are often defined only in high level supporting only main process flow. Nearly always there are sub-processes or country / business specific processes that are not considered early enough and are causing later problems when integrations are not supporting all needed functionalities.

To follow COBIT control objective AI2.10 (2005, 78) there need to be strategy and plan for the maintenance and other releases. In these plans there are many important aspects that should be considered, e.g. release planning and control, resource planning, bug fixing, minor enhancements, documentation maintenance, emergency changes, interdependencies with other applications.

According to the ITIL Service Transition (2007, 17) service scope includes the management and coordination of the processes, systems and functions to build, test and deploy a release into production and establish the service. Needed main activities are release planning, building and testing, deployment planning and actual deployment. An implementation project needs to take into account all these aspects also from the integrations' point of view.

In addition, ITIL states that to be able to ensure reliable and efficient IT services, there is need to have well defined release management and processes in place. The scope of release management includes processes, systems and functions to get a release into production and establish the specified service. Release management objective is to ensure that clear release and deployment plans exists, efficient and on schedule deployment is possible, there is capability to deliver required tasks, change will have no unpredicted impact to production services, and practises and outcome are in place. (ITIL Service Transition 2007, 84)

Business testing is a crucial element to provide new service or enhancement fit for purpose and fit for use, it will be guarantee for a quality assurance. Inadequate testing is in many cases underlying root cause for inefficient service. If enhancement to the integration is not tested sufficiently, it may rise additional incidents and clarification calls due to failures and mismatches, harder diagnosis for errors in the production environment, more costs because errors are corrected urgently in the production and inefficient use of the services (ITIL Service Transition 2007, 115). According to ITIL Service Operations (2007, 174) comprehensive and realistic testing environment need to be in place for all systems and components – which mirrors the operational environment in terms of volume as well as characteristics.

According to COBIT control objective AI4.4 (2005, 86) knowledge transfer and skills enable effective and efficient delivery, support and maintenance of the application and associated infrastructure. It is stated in ITIL Service Transition (2007, 180) that cooperation, understanding and mutual respect between a project and AMS partner are critical to ensuring that new and on-going delivery of services to the customer are optimized. The flow to gather wisdom for the future coming enhancement requests is presented in figure 14.

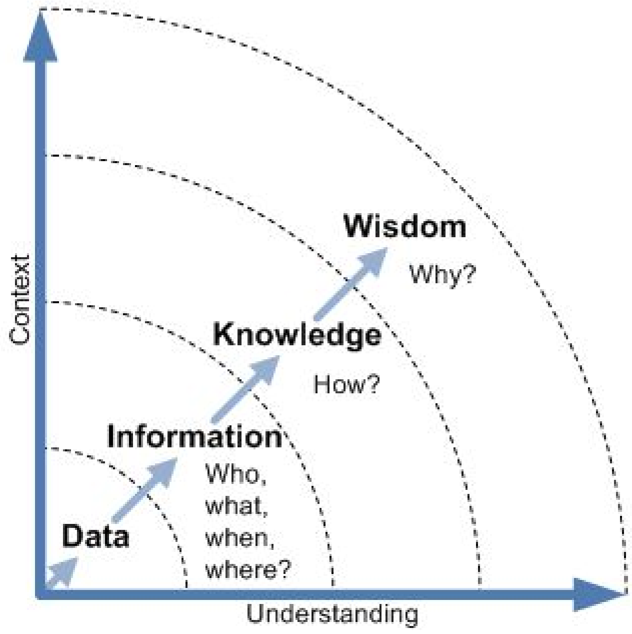


FIGURE 14. The flow from data to wisdom (ITIL Service Transition 2007, 147)

COBIT AI4 control objective (2005, 85) is focusing on providing effective user and operative manuals and training materials to transfer the knowledge for successful system operation and use. ITIL Service Operations (2007, 31) principles point out the importance of the documentation; definition and maintenance of the process manuals for all processes and technical procedures manuals. Documentation should be archived to the predefined tool and it need to be available for the support and other purposes. Additionally, from project point of view documentation should also be tested to ensure completeness and quality (ITIL Service Management 2007, 174).

Project management needs also adequate controls to ensure project delivery within agreed time schedule, budget and quality. In COBIT control objective PO10 has been created to support and give guidelines to manage of all IT projects, to ensure the correct prioritization and co-ordination. This can be done with predefined frameworks, management guidelines and adequate planning (COBIT 2005, 67).

2.8 General Data Protection Regulation

Many countries have own regulations for the data privacy. These regulations are a bit different in each country or area, but main driver is to protect personal or sensitive information. New regulation for the EU General Data Protection Regulation (GDPR) was established in EU countries on May 2018. Preparation work had taken four years, regulation was approved in the EU Parliament on April 2016 with two years transition period which led to final enforcement date 25.5.2018. This new regulation will replace old data privacy directive which was not binding as new GDPR regulation is. (Regulation (EU) 2016/679)

GDPR is designed to harmonize data privacy laws across Europe. The aim is to protect all EU citizens from privacy and data breaches, it will give people more control over their personal data. The regulation will clarify rules for companies how personal data should be handled and what data even can be processed. Regulation is having global impact, even company is not located in EU but is holding personal data on customers, prospects or employees based within the EU regulations must be followed. To be considered that the GDPR is not affecting only EU citizens but also data processed in the EU (LeBlanc 2018).

Personal data has been categorized into general personal data and sensitive personal data that is more important and sensitive to individuals. Sensitive personal data trigger additional and more onerous obligations as compared to general personal data. This data can be related to racial or ethnic origin, political opinions, religious or philosophical beliefs, trade union membership, health or sexual life or offenses. (Lambert 2017, 111 – 115)

Because case company for this thesis is also European company, this legislation impacts to all business transactions inside the company. From this study point of view also integrations need to be considered. Before new legislation in the case company there were no accurate instructions what kind of data should and can be transferred to another system. In some cases, data which was not actually required in target system, were transferred. New legislation gives very strict rules when organization can collect and process personal data. Organization needs to comply with the registration requirement, quality of the data, security, ensure processing conditions, provide information to individuals, implement compliance procedures and respect the data transfer restrictions (Lambert 2017, 131 – 132).

Personal data includes any information related to a natural person that can be used to directly or indirectly identify the person. It can be anything which identifies person, e.g. a name, an identification number, location, an email address, bank details, company id, computer IP address. As GDPR also supports the data minimalization principle, only needed data can be updated (Wright 2018). Making sure the data is anonymized may increase requirements for an integration and in addition slow down deployment.

According Lambert (2017, 137) there are several key rules for data processing for companies

- processing should happen in a lawful and transparent manner (lawfulness, fairness and transparency)
- needs to have specific purpose for processing the data (purpose limitation)
- collected data can be only necessary, irrelevant data should not be processed (data minimization)
- data needs to be accurate and up-to-date (accuracy)
- not allowed to use same data if not compatible with original purpose
- not allowed to store the data longer than necessary (storage limitation)
- to ensure the security, appropriate technical and organizational safeguards needed (integrity and confidentiality)

The breach in GDPR can be fined up to 4 % of annual global turnover or 20 m€ whichever is higher (Regulation (EU) 2016/679). This mean significant influence on corporate if any malpractices or offences will appear and that is why new regulation need to be treated much more carefully than previous laws. There are not yet many precedents available of authorities' interpretation for the new regulation, but any company do not take the risk to offend the regulations.

To secure data in the location where it is stored is not enough, companies need to understand that also in integrations between systems or applications that leads to need to consider data protection regulations also in other systems. It is usual to have same data integrated to several systems or applications and for different usage. It is mandatory to gain control over data that data can be corrected or removed in cases when it is no longer required, or individual will request data correction or removal. Same rules apply if personal data is shared with other parties.

3 METHODOLOGY

This chapter will give overview of used research strategy and methods. Due to working method during the study, action research was seen to be most efficient way to proceed. Explanation how used methodology was implemented in this study, will be shown in the Chapter 4, where Data acquisition sub-chapter will describe actual work steps and outcome of the steps.

3.1 Research strategy

Maxwell (2005) presented the model for qualitative research that includes five components which are tightly linked to each other's. Any component of the design may need to be reconsidered or modified during the research. Research questions are heart of the model and are connecting all the other components of the design. (Maxwell 2005, 3 – 5).

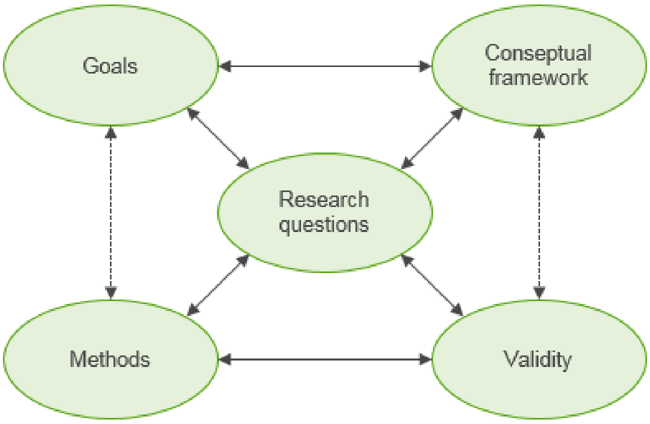


FIGURE 15. An interactive model of Research Design (Maxwell 2005)

In this thesis action research method is used as a strategy. Action research methodology's goal is to understand the practice and to solve immediate problem(s) in the organization. The origin of action research is unclear. According to Masters (1995) 'The history of Action Research' many authors state that action research is originated with American psychologist Kurt Lewin in the mid-1940s, but also different opinion exists.

Professor Emeritus Coghlan (2017) stated the action research is based on two assumptions which are also cornerstones of organizational development; (1) involving client to provide more valid data about how the system works and (2) understanding of the system comes only when one tries to change it. Results for the study are tested in action and by the people who are in the position to know their conditions better than conventional researcher can – research is constructed with people rather than on or for them. (Coghlan 2017, 11 – 15)

Action research has four themes (Saunders et al 2009, 147 – 148)

- research in action – resolution of organizational issues
- involvement in research – researcher is part of the organization within which the research and the change process is taking place
- iterative nature – each iteration round includes diagnosing, planning, action and evaluation phases (figure 16)
- implications beyond the project(s)

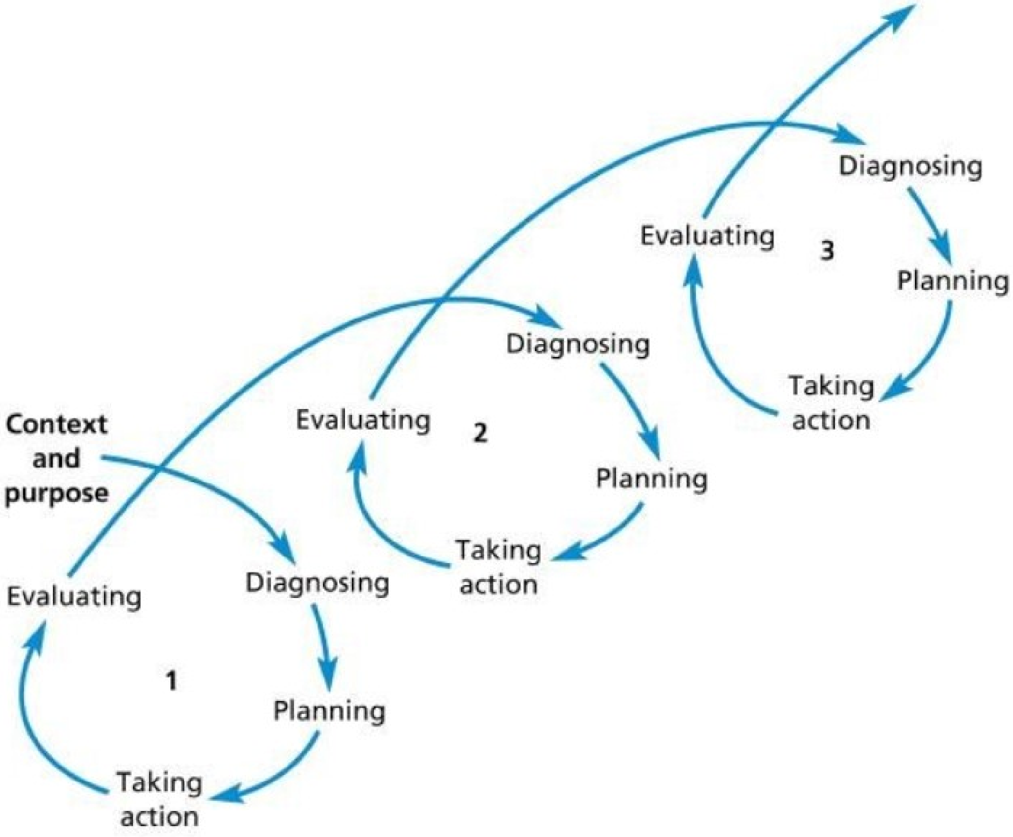


FIGURE 16. The Action Research spriral (Sauders et al)

3.2 Research methods

Research methods used are mainly qualitative, consisting of interviews, existing documentation and practical knowledge and experience from the on-going integration renewal projects (later Project A and Project B). There were more than one data collection technique and analysis procedure in use and due to this research chose multiple methods research (figure 17).

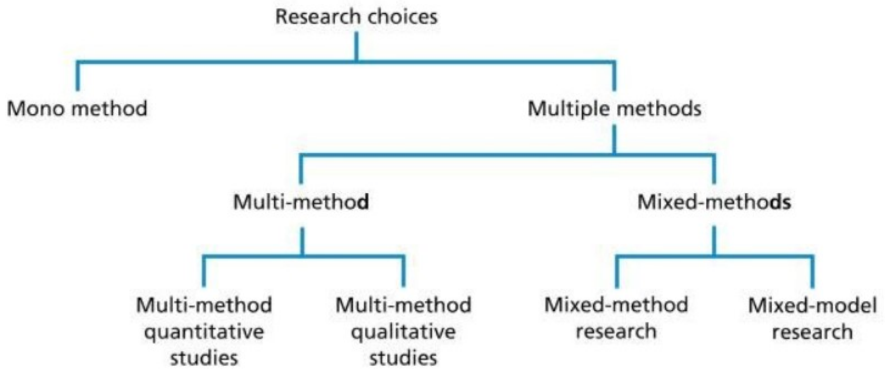


FIGURE 17. Research choices (Saunders et al 2009, 152)

Interviews consist of selected project people, both internal and external. Interviews were based on partly same questions and partly with individual questions according to the responsibility area of interviewee in question. Interviews were mainly semi-structured interviews where list of the themes and questions were covered, but questions were varying from interview to interview. Also, unstructured, in-depth informal interviews were established.

In addition, integration assessment from project A was used as a basis. Assessment in question was done already in the beginning of the study year when it had been noted that there were lot of problems with the new cloud application integrations. Actual assessment work and report was executed by an outsourced consultant company and was supervised by this study researcher.

Project A included 34 integrations that were deployed to the production in the moment of original project go-live and there were problems occurred in more than half of those. This report will be known as secondary data because it has not been collected specifically for the thesis purpose but general purposes to make decision if renewal project is required. Assessment documentation is confidential and will not be attached to the thesis, only main findings will be introduced.

Due to nature of this research, researcher had been participating fully to the renewal project planning and to the first phase as project lead, this can be called participant observation. Part of the primary data has been collected by the researcher. Based on Saunders (2009) participant observation roles can be categorized into four different roles where researcher identity and participation to activities are different (Saunders 2009, 292 – 293). In this thesis researcher role has been complete participant as other members are not aware of the thesis aspect and researcher's participance has been full.

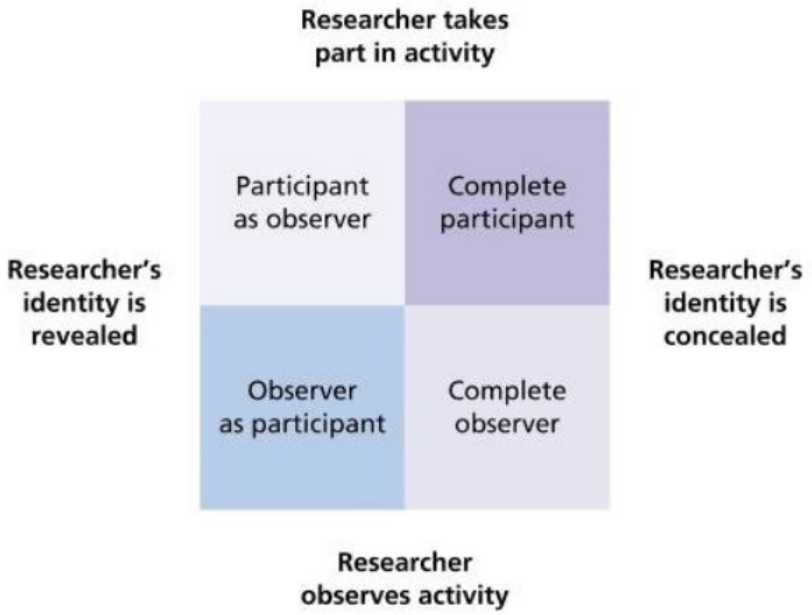

Researcher takes part in activity
Participant as observer
Complete participant
Researcher's identity is revealed
Researcher's identity is concealed
Observer as participant
Complete observer
Researcher observes activity

FIGURE 18. Typology of researcher roles (Sanders et al 2009, 293)

4 RESEARCH RESULTS AND ANALYSIS

This chapter will first give insight to actual reference projects and research phases done during the study. There have been two global level projects analysed. The first project included several stages, that should be considered as own iteration rounds. The second global project was started in summer 2018 and will continue until the end of 2020. It will have several sprints, where findings from previous sprint corrective actions will be utilized in the following sprints. Only first sprints can be considered as part of this study because project will continue even the study will be closed. After introducing reference projects and giving phase presentations there are findings arranged according to the main challenges. These main challenges are described in chapters 4.2 – 4.7. Each challenge area includes several pain points recognized and analysed.

4.1 Action Research process

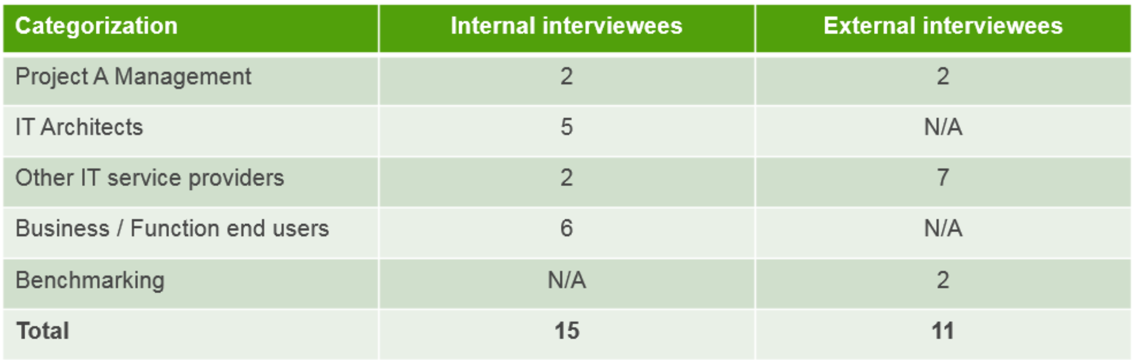
This chapter will focus on action research phases and analysis of different steps of the action research spiral – each of below sub-chapters should be considered as own iteration round with phases for diagnosing, planning, action taking and evaluation. Findings from the diagnoses and evaluation are analysed and connected to the theoretical framework and are presented in the following chapters.

4.1.1 Starting situation

Project A go-live was in April 2017. It was already known in the go-live that all integrations are not working as expected by the business. Original idea was to finalize incorrect integrations within couple of months after the go-live. Corrective actions were not successful and in the end of the year it was agreed to set-up separate study to investigate how many integrations are facing problems and analyse what kind of errors are appearing. From action research point of view this assessment reflect the base status.

Assessment diagnosis phase included study of the project A documentation as well as project key personnel's and other stakeholders' interviews. Related interviews were done by the external consulting company and whole assessment study was supervised by the case company's representative, researcher of this study. Interviewees were selected from the different stakeholders and categorization was done based on the role of interviewee.

TABLE 1. Interviewees categorization



Categorization	Internal interviewees	External interviewees
Project A Management	2	2
IT Architects	5	N/A
Other IT service providers	2	7
Business / Function end users	6	N/A
Benchmarking	N/A	2
Total	**15**	**11**

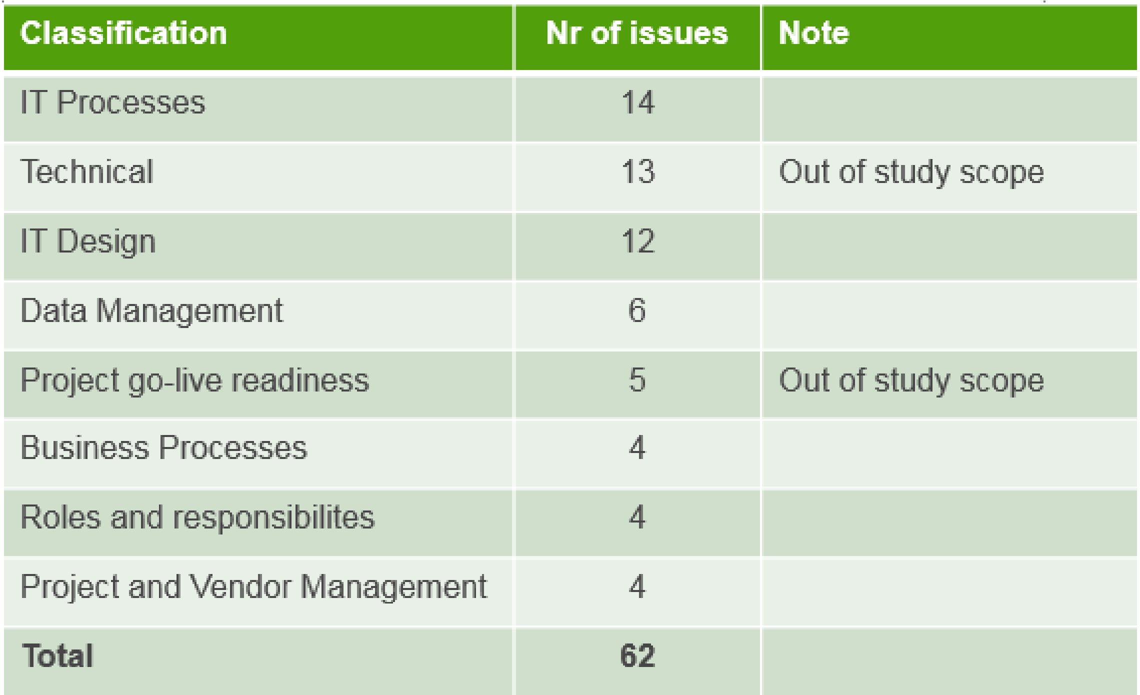
Interviews were done during October – November 2017 and were based on free discussion without any predefined questionnaire. Main point was to raise issues or concerns which were not considered sufficiently during the project. In the final report all interviewees were anonymous, this approach enables people to give honest comments and feedback without any pressure.

Issues raised up from the interviewees were categorized based on priority and pain level but only if interviewee was willing to give such evaluation. Because interviewees were from different target groups also priority and pain levels were commented on different angles and for that reason categorizations are not totally comparable. General comments were not having priority and pain level categorization as well as items noted from the project end report documentation.

During the assessment study there were 34 integrations in the production and new integration implementations were already in the pipeline. From production integrations, 17 were facing problems and were causing errors and mismatches in production environment. For all errors there were not even workaround defined. Total outcome was that 50 % of integrations were not reliable six months after go-live of the project.

Based on noted issues in the assessment, findings were classified into eight different groups according to the content of the finding. Some of issues could have been classified into several groups, but those have been considered only in one place which have been identified as a main pain point. Two of those groups, technical aspects and project go-live readiness, are not part of this study and will not be investigated further because study scope was limited to process and business view.

TABLE 2. Assessment issue classification



Classification	Nr of issues	Note
IT Processes	14	
Technical	13	Out of study scope
IT Design	12	
Data Management	6	
Project go-live readiness	5	Out of study scope
Business Processes	4	
Roles and responsibilites	4	
Project and Vendor Management	4	
Total	**62**	

Assessment report was presented to the steering group on January 2018. Meeting outcome was that steering group decided to continue to the planning phase where new assessment study team should give proposal to the corrective actions and next steps based on the original assessment. Team was nominated and started planning activities on February 2018. The assessment findings were investigated and prioritized together with the business representatives. For each finding team prepared proposal for a possible new approach or a corrective action. Assessment study team proposed approach to proceed with several stages due to the controllability and amount of the corrected integrations. Assessment proposal was presented to the steering group and was approved for execution.

4.1.2 First iteration round

Project stage 1 should be considered as first iteration round with all relevant phases; diagnosing, planning, taking action and evaluating. This stage was started immediately after the steering group project approval. The first iteration round was taking almost six months and researcher was leading the integration renewal project's first stage.

Diagnosing and planning

Scope for the first stage was agreed with the relevant stakeholders by taking into account business and function prioritization. Due to the large scale of integrations, only prioritized integrations were assigned to the first stage. Issues noted and classified during the assessment study were considered against final scope of the stage. In the diagnosis phase project team investigated issues collected to the assessment report. All issues could not be considered in this stage due to fact that the stage scope did not include all relevant aspects. Missing issues were agreed to be included to the following stages.

Taking action

Proposed actions defined after the assessment work were executed and solutions were tested by the project team. During the execution and when more details were discovered, it was found out that all proposed solutions were not working as expected. With new detailed level information project team created new solution proposals and executed those with better results. Also, totally new issues that were not included in the original assessment were raised.

Evaluation

Final evaluation for the stage in question included questionnaire to the project team representatives and researcher own observations. Questionnaire was free format report where all project members had possibility to comment and criticize issue solving and raise up possible new issues. As an outcome of the questionnaire and observation, the end report for project stage 1 was created and presented to the project steering group. The end report included comparison between original and realized scope, lessons learned and project team open feedback. Issues which had beneficial solution were agreed to implement as new ways of working in the following stages and new ideas were agreed to be tested there.

4.1.3 Second iteration round

Next project stage with new content scope was next iteration round for the whole integration adjustment project and it contained issues faced during the assessment and new or slightly modified issues that had been realized during the first stage. This new iteration round included again all themes; diagnosing, planning, taking action and evaluating.

Diagnosing and planning

Stage two started similar way as stage one, by agreeing the final scope for the stage with all relevant stakeholders. Scope content was dictating what issues from the assessment and from previous stage could be considered and retested. Stage one end report outcome, lessons learned, project team feedback, and researcher's own observations were basis for the new stage planning.

Taking action

Stage two was led by outsourced project manager, because researcher who was leading stage one was nominated to Project B. Researcher was anyway having supervisor role in the Project A, but this time focus was more in observing and supporting new project manager than actually participating to the development work. In the supervisor role researcher gave the needed information and support to ways of working in actual execution phase. Issue solving, and resolution testing was continuing as planned in the execution.

Evaluating

Continuous evaluating for integration adjustment project is done and outcomes are validated by the researcher and are presented in the stage steering meetings. Again, corrective actions to the project processes are implemented immediately if such cases appeared. Evaluation will continue until the whole stage will be closed, that will happen latest in the end of the year 2018 and lessons learned will be collected to support the next stage which is planned to continue almost whole year 2019.

4.1.4 New SaaS application implementation

Project B was established early 2018. It was new global SaaS application implementation project where target was to implement new application to nearly all group companies within 2,5 years. As previously presented, also this project included a lot of integrations to different ERP or other systems. Researcher was nominated to the project as IT Lead when project organization was changed after summer 2018. First roll-out was supposed to be in September 2018, but none of the integrations were ready even those have been built for several months.

Diagnosing and planning

When starting in the new project, existing pain points and findings were studied, and issues were compared to earlier projects integration implementation findings. There were lot of similarities in the problem areas even though systematic questionnaire or analysis could not be done in this case because time schedule was tight due to target to get first company into production within one month. By taking into account learning points and best practises from earlier project and stages, integration implementation plan was created.

Also, in this project there were no clear business requirements for the integrations, not even any clear picture how many or which integrations are needed. Suppliers were implementing integrations without any specifications or prioritization of the tasks. Detailed level definitions were not considered or documented, integration development supplier was trying to develop integrations without full understanding of the global processes and business requirements. Supplier work was not led by case company representative and because several project members were giving instructions to developers, nothing was finalized.

SaaS application provider did not really understand fully business requirements and business representatives did not understand new process logic and just thought that application will work similar way as the previous on-premise application. Only mapping tables were received from SaaS supplier without understanding of an application process functionality and a usage of each data field. Different basic process handling was impacting heavily to integrations, all best practises from SaaS supplier could not be taken into use due to limitations in old ERP system.

Taking action

New integration implementation plan was used as a basis for new scheduled build activities. Each integration implementation was designed, documented and prioritized according real business requirements. It was clear that all integrations could not be built within the original schedule due to missing design, but implementation work continued with more precise design and documentation work progressed and whole project integration work was again in target schedule within two sprints.

Evaluation

After each sprint ways of working and checklists were reconsidered and possible new findings and plan to prevent such issues in next sprint were established. Based on the evaluation topics following sprints that can also be treated as following iteration rounds, will have more efficient approach to integration build.

4.2 IT processes mismatches

In the case company IT services are currently mainly provided by the outsourced suppliers. To fulfil ITSM commitment described in Chapter 2.7, implementation project need to ensure that all related functionalities and processes are planned to support needed requirements and handed over properly to the line organization. In this study there were findings where all the aspects were not realized as it should be in the best practise point of view.

Traditionally even small enhancement work in an application has meant need for IT organization to entirely develop a new functionality. IT developer had full understanding of the data modelling and linkages between new requirements, existing data elements and integrations. And as described in Chapter 2.5, data modelling is one of critical factors of avoiding data quality or data content problems. Currently many SaaS applications give possibility to transfer simpler development execution to the business people. Configuration capabilities have been set up to support small enhancement work without deeper understanding of a programming or software development. Due to this new possibility, also in the case company business development organizations have been built up to implement such enhancements. People in these organizations are often business key users who participated implementation project, but do not have actual IT programming background and are not familiar with the data modelling and other technical linkages. Finding from the case company was that new approach to do business development can lead to the situation where changes will cause problems elsewhere if IT landscape and possible implications to the other applications and integrations are not fully understood.

As described in Chapter 2.7 changes to application environment need to be done in a controlled manner. If ITSM processes and best practises are not followed, it will lead to the situation where business development is doing enhancement in the application or integration with agile way, without release planning, full testing and deployment management. Business development prioritizes only quick business process enhancement and how to get source system enhancement working most efficient way without any delay, but without full understanding of possible impacts to the data management and integrations.

In case of such development done by a business development organization, AMS partner is not aware of any changes and cannot react properly if there are problems in the integrations. In these cases, business development organization should be considered as IT build supplier against AMS partner and development organization needs to understand that supplier role and role requirements will belong to them. It will be critical that business development follows the ITSM processes (e.g. release management, tools, knowledge transfer to AMS partner) to support overall coordination and follow-up and to guarantee high level services also after deployment. This is in many cases difficult to get realized because by default business organization do not need to follow ITSM processes and business management is not willing to take new responsibilities or tools into their scope. If business development organization will be set up, also the role towards the whole service need to be understood well before actual production implementation. With such organization, RACI, a model used to define who is responsible, accountable, consulted and informed, should be created.

As stated earlier, release and deployment management is one important service transition process to be taken into account when implementing new application or integration. The SaaS application has by default supplier's standard upgrade releases that will be deployed to the customer according to the predefined schedule. Frequency can be monthly or quarterly or yearly depending of the stability and size of the SaaS application. Internal process to support these upgrades need to be planned already during the project. There need to be clear definition of the following responsibilities; release notes to be investigated from the functional and integration point of view, agree possible activation of the new functionalities and arrange needed resources for the testing. In addition of the supplier upgrades, release and deployment management is needed for the company's own development, that can also be functional, or integration related. As well as supplier releases, internal enhancements need proper release management in place to secure seamless transfer to the AMS partner and to guarantee on-going service management levels.

In the case company's assessment study and observations during the Project A different stages it was seen that missing release and deployment management caused lot of extra work to the service management. In the beginning there were no clear rules and practises to transfer enhancements to the production. Release process was developed during the study time, but official release management process was still missing.

Knowledge transfer to AMS partner process should include integration related business process understanding to ensure that new and on-going service delivery is optimized. It will not be adequate to understand only technical content of the integration but also actual business requirement behind the code value need to be understood. Business process understanding supports AMS partner work when trying to solve integration related problems with different parties who might define same topic differently as described in figure 14.

To be able to support AMS partner's efficient work, adequate support tool need to be in place. Many SaaS application suppliers do have own ticketing tool that is supporting supplier efficiency and are not willing to use any other tool for the support. Also, incident classification may lead to situation where there is need to create ticket in the different support tools (e.g. in cases where there are several AMS partners with own tools). Process for supporting tool usage need to be defined already during the project. Additionally, definition should include the process description how to compare and match support requests and incidents between the different tools, especially when main AMS partner has clear end to end responsibility.

End to end integration documentation is critical to exist. Many times, in the case company integration documentation has been defined in three places but including only definition viewpoint from the party in question. In practice source system, ESB and target system are having each own documentation but overall understanding that should combine the data is not existing. To be able to support ITSM and service operations, full picture for the whole integration should be available and shared with AMS partner.

Current approach in the case company is that development is required to be done with the agile approach. Many business people have misunderstood agile, and demanding development starting even no clear requirements are defined. Also, the agile development has meant business people to do development without any proper communication and planning for releases. Agile should be treated as mindset, not methodology. Generally, it will be less suited for the integrations where there is low level of uncertainty – predictability, planning and control are in place. To keep in mind that it will be waste of time and money to build integration without proper planning and actual requirements with trial and error -mode.

As a summary following cases can be listed as mismatch topics in case company's IT processes

- Lack of ITIL service processes (build, test, deployment) understanding during the project
- Lack of release and deployment management process understanding, including SaaS supplier release management
- Unstructured enhancements done by business development
- Imperfect IT controls follow-up in the development work
- Missing overall coordination and follow-up to guarantee high level services after deployment
- Inadequate end-to-end business process knowledge transfer to AMS partner
- Inadequate instructions and processes to different support tools usage
- Inadequate documentation of the whole business process affecting integrations
- Agile development process treated as methodology, not as mindset

4.3 IT Design mismatches

Based on discussions and questionnaire results, the main challenge in the integration work is underestimation of the integration complexity. Many times, in the case company project representatives do not really understand that integrations are more than the simple mapping table, and even mapping is occasionally missing. Also, approach is quite commonly decision that integrations will remain the same as those were with the previous on-premise application. This decision has been made without even understanding content of the data elements in the old and in the new application, and possible mismatches between those. There might be big difference what certain code value or process means in the new SaaS application compared to the old system environment. If data elements will not be investigated carefully, it will lead to situation where target system is getting information which is not the same as required and was earlier got from the old integration, even the field or process name is same as previously.

The case company's best practise in enterprise level has been to use an enterprise service bus (ESB) as a middleman. ESB creates a service that includes rules and principles for integrating different applications together. Different applications are connected by putting communication bus between them and ESB controls message transfers. In this context integrations include three parties: source system, ESB and target system. The role of ESB should be clarified already during the implementation project. Should ESB be only service which is doing needed mappings and forwards as-is data to the target system or should ESB also enrich the data with additional queries from the other systems or calculate some new values based on the source system data before sending it to the target system. From business users point of view transparency for the data is many times critical and when enriching the data in the ESB this requirement is not fulfilled. End user do not have full visibility to the changes ESB has done and enrichment or calculation might have negative impact to the system performance and leading to possible timeouts in the integration. These risks need to be considered carefully before finalizing the IT design.

Another aspect in the ESB role is that should some data be stored in a place where ESB is only dividing similar data to the relevant parties. In case company all integrations related to the projects in question were point-to-point integrations, by using ESB as a message broker. There are situations where almost same data was transferred several times from the source system to several different target systems. Should be considered possibility to have only one integration from the target system to ESB, including all data and then divide only needed data elements to the different target systems. In this scenario data privacy topics to be considered to guarantee that stored data is not available for unauthorized personnel or data is not transferred to the parties who especially are not requiring it.

During the assessment and even in the project A first correction stage it was found out that the case company was lacking global understanding of the integrations and enterprise level target architecture. When each of the three counterparties have own architect and they try to promote their own agenda, there was nobody to check whole corporate view, what is the reasonable and the most beneficial solution globally but not necessarily most efficient from the one-party point of view. No-one of the application area architects did not want to take additional workload needed to provide required elements to the target system, even decision was needed to be able to continue with the development. There should be enterprise level person in place whose responsibility is to check integration solution from global point of view and to give final decision and approval to be able to proceed. Approved solution needs to be in line with the defined IT architecture and technology standards (COBIT 2005, 81).

Mainly the SaaS application provider is giving opportunity for several environments (e.g. development, test, production). Tenants' refresh schedules are quite commonly predefined based on the supplier own rules and separate release coordination is required from customer. New configuration or development activities, including testing, need to be adapted based on given schedules that are not necessarily based on customer company's own wishes. The whole process requires structured approach to guarantee efficient development process and knowledge transfer to the AMS partner.

In addition, if test tenants are refreshed from the production, it need to be clear how data scrambling is done in the other environments. If there is any privacy data related content in the test environments, proposal is to scramble the data. Nearly always test environment has wider access rights than production environment but if data scrambling is not done, environment should have same authorizations as in the production to be able to secure the data privacy. In the case company this has prevented effective testing because project personnel and testers are planning and developing whole chain but cannot see or test functionality in the testing environment because lack of the authorization.

When planning IT design, it will be critical for a developer to understand business process and application functionality to be able to build solution most effective way. If requirements have been giving only by defining needed data elements as done in many times in the case company, enhancement may fail. Business representative assumes something which has not even been requested and developer cannot guess what the actual requirements behind individual data elements are. Specification requirement document including description of related business processes, functionalities and data elements is needed to give global understanding of the requirement. Own template for enhancement specification to be established with all relevant sub-chapters to fulfil business requirements and support business representative when doing definition.

As a summary following cases can be listed as mismatch topics in case company's IT design

- Integration complexity underestimation
- Misunderstood data elements and data architecture
- Incomplete mapping documentation
- Decision about ESB role (e.g. transparency, visibility, performance, GDPR aspects to be considered)
- Lack of enterprise level architecture responsible
- Insufficient rules for tenant management (incl. GDPR requirements)
- Inadequate business process understanding

4.4 Data Management mismatches

Businesses are dependent on the data, it is key aspect for everything in the business, to run operations, to understand, adapt to market and have basis for decisions at the right time. During the research it was seen that because global data governance model is not in place, data management has not been considered carefully when new application has been implemented, clear ownership and responsibilities for the business data are not defined in all cases. Understanding of managing the data is not clear, even definition, dimensions or level of represented details for certain subject are different within the different stakeholders.

Formal management of the data is the basis and equals to continuous quality management for other products. Data should be managed through its lifecycle by setting enterprise level standards and quality to the transactional processes. When adequate governance model is in place also other aspects can be considered. Reliability of the master and transactional data, which will be the key for further operative processing and related integrations, is crucial. Any uninformed decision or action related to the data can result to poor quality of the data. Cross-business and cross-functional level communication process need to be established to avoid miscommunication or lack of an information.

Basic assumption for the data is often ’raw material for the information’, like layered pyramid where at the base level we can see data and at the top of that information, knowledge and wisdom. Assumption is quite commonly in the case company is that data just exists, but that is not the truth – data has to be created and company needs to ensure the quality of the data. Data is a form of the information and information is a form of the data. (DMBOK2 2017, 1 / 2.2)

Quality aspects, as described in Chapter 2.2, are crucial factors to provide reliable data. One aspect in low quality data is misunderstood or misused data. In case company there were information gaps between the data source and actual requirement, there is same denominator, but actual requirement is not the same. Due to this, adequate communication process need to be established, data owner need to be sure where data is used in target system and is the data content exactly what was expected form the target organization point of view.

New application implementation requires nearly always data migrations from the previous system. Before starting any migration activities to the SaaS application, to conform data quality cleaning activities are needed. Incorrect data should never be migrated to the new application, assumption to clean data later will never happen or is happening only partially and incorrect data will stay in the new system and will impact negatively to the integrations and target system functionality. In the case company cleaning activities have been seen in many cases too time consuming and difficult during the tight project schedule and this is reason why this activity has been failing. Cleaning includes detecting and correcting the data errors to ensure quality in the new application and related integrations.

Mainly data models are not the same in source and target systems and therefore data mapping is crucial to keep data modelling up-to-date. During the assessment it was recognized that in the case company mapping was documented only partially, and actual mapping was visible only via a system code. Mapping should always be documented carefully and should include also information about possible enrichment done in the integration (e.g. in ESB layer). Documentation ensures effective incident resolution, support for the future developments and visibility for the integration data. Similar data mapping is needed in case of the data migration from the old system environment. In the migration mapping this should be part of the project documentation and future maintenance is not needed.

One part of the data management is global understanding on where and how company's data is used. There need to be clear visibility and documentation how target system is using the data which have been provided via integration to the target system. Data owner, who is commonly key user of the source system, is finally responsible for misuse of the data in the target system. If the target system is feeding third system with data given from the original target system, whole chain is needed to be understood by the data owner. There is linkage also to the data privacy rules and data owner have overall responsibility of data usage inside the company.

In the case company insufficient understanding of the application functionality and usage of the data in different functionalities have led to situation where errors occur in the integration because requirement definitions did not consider each application functionality separately. Data content in the integration can differ when doing same business process via separate functionalities. This makes it very important that requirements are defined from the all angles.

As a summary following cases can be listed as mismatch topics in case company's data management

- Missing data governance model (strategic and operational)
- Missing data ownership and responsibilities
- Data inaccuracy due to inadequate quality
- Missing data management throughout the lifecycle, including data quality
- Missing cleaning activities before migration
- Inadequate data modelling and communication between parties
- Missing end-to-end documentation of data usage

4.5 Business Processes mismatches

As described more precisely in Chapter 2.7 control environment demands clear requirement definitions and design already in the feasibility phase of the project. These definitions will be adjusted during the planning phase when also overall knowledge is increasing. Definitions need to include business process functional and control requirements to be able to design most efficient and cost-effective solution. It has happened in the case company projects that detailed definition with all needed functionalities have been raised only when solution design has been finalized and actual build of integration has started. This has caused need to open the whole solution and start redesign activities.

In the study it was found out that many cases in the case company only basic scenarios were tested and many country or business specific scenarios were not even specified. Adequate test scenarios are beneficial for the business, it is not only IT requirement to secure future services with the AMS partner. Defining all possible scenarios affecting integrations will take time and effort from the business but will be advantage in the later stages. Due to new GDPR regulations test cases need to be reconsidered, even existing ones. In addition of business scenarios, IT supplier needs to provide test cases for volume testing and possible regression testing, especially in cases when any change in a component is touching already existing integration. After defining all relevant scenarios, acceptance criteria for testing need to be defined (COBIT 2005, 74). Business need to have clear understanding of what kind of case failure is showstopper for the production deployment and to understand possible business risk to take when approving to production something which is not fully fulfilling the requirements. If incorrect solution will be taken into production, workaround should be described and communicated to relevant parties. Additionally, possible impact to AMS partner's SLA's to be considered.

All test execution cases should be also archived to have evidence for successful or unsuccessful test. This is important in later stages when possible incident is raised. There need to be clear understanding if integration has worked earlier in similar situation and now got broken due to other enhancements or was functionality something which has never worked properly but has agreed workarounds. If common tool for the testing is available, it should be used even though some business areas or functions are willing to use only own spreadsheet to follow test progress, usage of own documentation should not be accepted. When using own documents, there is no visibility for the overall situation and later those documents cannot be used to support AMS partner work in case of the incident. Based on the study, coverage of the testing scenarios and testing tool usage has not been in good level in the case company.

Test case planning should be started early enough, already when collecting business requirements. Test library could be re-used during the implementation project and should include ready data settings for each case to guarantee efficient testing with relevant data. Test library will need frequent updating and responsible person to creating, cataloguing and maintaining test scripts. Test library should include information if test scenario is affecting to any integration, then it would be possible to recognize testing needs related to the integrations. It would be also beneficial to prioritize test scenarios in sense of a service and business risk. In case where all test scenarios cannot be executed this is important information for the acceptance criteria definition. In the case company test library was partially existing but not categorized.

Effective testing management requires adequate tool to be used, in the case company there is global tool for the purpose, but the tool usage is not always efficient. The tool should be used to manage, monitor, control and report the test execution. During testing on-line transparency is needed for test execution, also reporting and monitoring is needed. If testing tool is not up-to-date, it will be impossible to have full picture of defects and prioritization of those. Evidences for testing scenarios to be archived in the testing tool if possible, in case test data has elements where data privacy requirements, archiving of scenarios need to be restricted and common tool cannot be used for the evidence archiving. This is to be considered separately and restricted archiving tool or folder for evidences to be organized.

The SaaS application has often own test tenants which will have frequent refreshes from the production, this is case also in the case company. If test data is not scrambled and is including personal data, requirement is to limit user accesses to same level as they are in production. Commonly in test tenants’ accesses are much wider and integrations are tested quite freely, this cannot be done if the data contains privacy data. In addition, test integrations cannot be used or investigated by non-authorized personnel. This can lead to situation that all defects are not even noticed during the testing due to end user inadequate technical knowledge.

In the assessment one observation was that business users were expecting application to work exactly same way as previous system even such functionality has not been defined to be requirement during the project. When implementing SaaS application, it will be topic to be raised from the beginning in the project meetings, business need to be ready to change the way of working and cannot expect system to behave exactly similar way as old system did. As SaaS application has been built to gain benefits from the large customer basis, it is not necessarily supporting individual company based different approaches. If requiring changes to system basic functionality company will lose part of benefits of the SaaS solution and application itself may not work as expected after normal, frequent upgrades. Also, to be noted that all customized changes to the application and integrations need separate regression testing during all upgrades because those are not tested automatically by the supplier. If upgrades are frequent, it can be time and resource consuming effort from the business, unless automatic testing tool have been implemented and automation can execute similar tests during each upgrade.

When implementing any new solution, country or business level process harmonization should be considered, this was not done enough deeply in the case company. Different country level approaches to be harmonized if change is possible from the local legal point of view. This is especially important in case of SaaS application, due to fact that special configuration is not even possible or requires extra effort and loosing of SaaS benefits. In addition, country specific rules are not easy to implement and upkeep in integrations.

As a summary following cases can be listed as mismatch topics in case company's business processes

- Inadequate business process definitions
- Inadequate test tool usage
- Inadequate testing and/or missing test scenarios
- Insufficient documentation of test evidences (GDPR aspects to be considered)
- Unrealistic expectations of SaaS application functionalities
- Insufficient process harmonization

4.6 Role and Responsibility challenges

Assessment findings raised up the requirement to have clear after go-live roles and responsibilities defined already during the project. Roles and responsibilities definition need to include run services as well as developing new enhancements. When new development is done after the project, there should be clear instructions how it should be done, especially if development supplier is not same as it was during the project or business development organization has been established. It is critical that base rules for integration tools and practises have been adequately documented and explained to the new partners to avoid additional questions and inappropriate solutions after go-live. There have been gaps with these topics in the case company.

A new development partner need to follow same documentation principles as done in project phase or agreed during the project. Responsibility and needed level of the documentation need to be clearly defined in the work order. These were practical issues notified after project A, where all needed documentations and enhancements after the project were not defined enough detailed and AMS partner could not get the information without drilling down to the system code.

Another stakeholder proposal from the assessment was that integration build should be centralized to the one development partner to keep process and efficiency in high level. When only one partner is doing development, there is no need to go through basics for integrations with the new partner during each development. This approach should be considered at least with minor enhancements and in these cases, there would be also beneficial to have same partner as in the run services to guarantee smooth knowledge transfer to the line organization and same time increase the wisdom as described in the figure 14.

As a summary following cases can be listed as mismatch topics in case company's roles and responsibilities

- Inadequate role and responsibility definitions after the project (including supplier tool usage)
- Missing definition about development partner documentation responsibilities
- Missing guidelines to development supplier selection

4.7 Project and Vendor Management challenges

SaaS application supplier knows its own system, but normally is not able to support integration from other system point of view. It is very easy for the supplier to state that integration solution with another system is standard and can be built in short time without any deeper understanding. This statement should be investigated already in the early project phase and check that needed resources are available from the source and target system sides, there cannot be assumption that no changes are needed in another side of an integration but each and every integration need to be checked separately.

Tight schedule is always something that is coming up after the projects, same in Project A assessment. Too tight schedule was noticed in the end of the implementation and was impacting to the test execution. Functionalities and integrations are approved to the production with very light testing and even noticing the functionality is not working at all. Missing functionality was written to the project backlog and incorrect development was deployed to the production. If there is no solution available how to solve issue in question, should be considered several times before releasing malfunction to the production, and question should be raised also to the integration target system and identify what incorrect data will cause to them. Incorrectly working integration can produce even more errors to the target system and manual corrections are many times difficult and time consuming. Project management should admit if tight schedule is risking the reliability of the application and in high / medium risk cases even consider postponing the go-live.

Many SaaS application implementation projects do not have dedicated full-time resources for integration architecture planning and based on observations this is causing lot of delays before even starting actual build work, as well as build resources are commonly limited. This was case also in the case company. COBIT control objective PO10.8 (2005, 68) states that project resources should be defined with adequate responsibilities and performance criteria's. Many times, these requirements are not fulfilled due to several overlapping projects that are resourced with same person.

As a summary following cases can be listed as mismatch topics in case company's project and vendor management

- Lack of vendor understanding of existing integrations
- Consequences due to tight schedule
- Lack of integration architecture and build resources

5 DISCUSSION

This chapter will present key findings and answer the research questions presented in Chapter 1.2. It will also propose recommendations to the case company and alongside with prepared checklists to support project management work in the planning phase, IT Lead work and data management related work. To be noted that checklists are meant for the case company and situations where global solution impacting whole corporation will be implemented and integrated with several other systems. Detailed checklists are presented in appendices.

5.1 Summary

Main research question for this thesis was to answer what could be the most effective approach to manage, identify and understand challenges when implementing SaaS application and provide needed integrations to the other systems. Secondary questions were related to SaaS application processes and data identification and responsibilities. Aim was to identify main pain points and create list of best practises to support project management when planning and executing implementation project with lot of integrations to existing systems. Basis and reference for the study were case company's closed and still active projects to global functions and where influence is visible nearly for all other functions and business areas.

One global project in thesis scope was closed at spring 2017, but with awareness that lot of integrations were not working properly, and aim was to do corrective actions immediately after the go-live. Due to incapability to correct all incorrectly working integrations, in the end of 2017 there was separate assessment study raised to recognize what kind of problems are still occurring in the integrations. This assessment work has been used as a reference of this study and is described more detailed in Chapter 4.1.1. Actual integration adjustment work started in the spring 2018 and included several stages. These stages could be seen as action research iteration rounds where redefined processes and practises were tested. Each iteration round outcome brought on new findings and observations with new solutions that were tested in the following round. In addition, new global project was launched in spring 2018 and in integration execution work started at autumn 2018 where these same best practises were established.

Theoretical framework for the study was based on main topics found from the assessment. Because it was question of the one company's challenges in actual environment, framework gave foundation for the pain points in the research and was supporting researcher to make conclusions in the study. Main findings are based on actual projects and outcomes and observations done there. Additionally, to be noted that case company is following ITIL -processes very straightforward, and due to this reason functionalities or processes that are working well have not been raised in this study.

Conclusions for the study bring up shortages in IT and business processes, IT design work, data management as well as overall project and vendor management. In addition, there were unclarity with some roles and responsibilities. Mismatches in all these subgroups were collected and categorized according corresponding research question. Proposed approach for the future has been discussed in Chapter 5.2 and listed in detail in the appendices.

5.2 Recommendations

Thesis objective was to create supportive best practise listing for project management, check list for IT Lead in the project and data management related recommendations. Items that have been creating problems or challenges in the reference projects or iteration rounds have been collected based on research results and analysis in Chapter 4 and final lists are presented in appendices of this study.

The case company related main finding to integration set-up was general underestimation for complexity when trying to integrate totally different systems or applications with each other. Best practise listing in Appendix 1 includes 10 topics to be considered already in the project detailed planning phase. Thoughtless decision to have integrations continue as previously should not be given if adequate study has not been done. There should be more effort put to the integration planning already from the beginning of the project, that ensures integration build work will be all the time in line with process development. Understanding of related business processes and possible harmonization is required to get effective approach to integration build. Also, business need to understand that in SaaS application implementation project, configuration possibilities are restricted, and all wishes cannot be provided. Even though all development should be done agile way, project representatives should understand that agile is mindset, not methodology. Integration implementation without proper planning and requirement definitions will not work. Project need to ensure adequate enterprise level architecture validator availability for the integration design work. Also, data quality is seen a big challenge, project need to understand everything is based on high quality data, starting from possible data migrations from the old systems and where needed cleaning activities should be done.

More detailed checklists for IT Lead are proposed in Appendix 2. All topics are not relevant to all projects but proposed to be checked to avoid pitfalls. One of the main topics to be noted is that integrations require work, resources should be reserved in time. Standard corporate level rules and practises for integration architecture should be established and the role of the ESB to be clarified in global level. There will be always exceptions, but basic way of implementing integrations should exists, e.g. is ESB enriching data or doing calculation or being responsible for the delta investigation. Currently same discussions are done over again with each integration and same arguments are faced. If common rules would exist, there would not be need for such dialogue.

After new data privacy legislation implementation in May 2018, privacy topics need to consider much more detailed, taking into account target system usage of the data and more tight rules and requirements for the test environment. Data scrambling should always be pre-requirement for test tenants if data includes any personal related or sensitive data. If SaaS application provider do not have such functionality, process and tools for scrambling should be investigated and established in the corporate level. When data is same as in the production, also access rights to test environment should follow production accesses to avoid possible risk for data privacy regulation offences.

Documentation is one topic which should have more attention. For integrations full understanding of all stakeholders' development, including source system, ESB and target system to be carefully documented with definition of possible business process connections. Also, data mapping and usage in target systems need to be described and understood. All documentation to be handed over and archived for AMS partner purposes.

Due to easiness to do business development via simpler configuration in the SaaS application, separate business development organisations have been established. If such new way of working will be set, proposal is to make sure that all needed ITSM processes will be followed also in the future development to avoid mismatches in integrations. Proper coordination and communication is needed between business development and AMS partner. If business development is taking traditional IT role by doing configuration, also responsibilities towards relevant IT controls to be agreed clearly.

Before starting testing, instructions and processes should be clarified with all parties and common testing tool to be used for controlling and monitoring. Business testing related to integrations should have more attention before approving any integration to the production, whole test set proposed to be prioritized and clear approval criteria defined in the early phase. Test scenarios should be prepared with a comprehensive list of the business cases, not only main ones. Automated testing tool usage should be increased to be able to have more extensive testing. Additionally, in cases where new customization has been implemented to SaaS application, regression testing is many times needed to ensure business functionality after changes in the process or integration. Regression test should include in addition of application functionality also integration to target system and possible influence there.

In case SaaS supplier is having own support tool that needs to be taken into use, it will be important to define process how to combine and manage service requests in company’s own support tool and supplier tool. There needs to be clear instructions and audit trail between these separate supportive applications. Also, responsibilities and processes to have updated data in both systems to be considered.

Appendix 3 concentrates on data management and related responsibilities. In the case company data management is proposed to be more structured. Data governance should be set-up in the corporate level. Current approach where individual businesses or functions are setting up individual projects for the data management without any enterprise level governance model is not effective. This will lead to situation where competitive governance models are existing inside the corporation, depending which organization feels owning the data. Corporate level data standards would be beneficial, not reinventing the wheel in every organization. Data ownership and responsibilities towards it need to be clearly defined.

Data quality has been seen one of the biggest problems impacting integrations. Managing the data efficiently in the corporate level requires data quality team or responsible person whose role is to engage business and technical team professionals and is also driving the work of applying quality management techniques to the data to ensure that data is fit for variety of purposes. This team or person should be involved to support projects or individual enhancements where global data delivery requirements have been identified. Role is to establish processes and best practises for high priority data in projects and in addition operational responsibilities for reporting, analysing, quantification and prioritization the data issues. Also supporting those who need data in their job, to ensure the data meets their needs with relevant creation, update and deletion processes. This kind of approach would ensure that each project or individual requestor do not need to reconsider the process and do same mistakes as done earlier.

5.3 Critical evaluation

As this thesis was based on the case company's global functions' projects, findings in the study are not all-embracing and will not support project management in all cases. The study will not consider items that are in good condition in the case company. If e.g. ITSM processes are followed, topics are not raised up in this study. Viewpoint in this research was project practicalities and process, not including technical aspects how integrations can or should be provided.

The trend will lead more and more to the SaaS application integrations in the case company, cloud first strategy will continue. It will be extremely important to find the best practises how to implement cloud solution especially when new application will be deployed with several integrations to the existing systems. To be able to support business enhancements and to guarantee go-live in given timeframe, whole project group need to have standardized, efficient ways of working and understanding of the whole end to end requirements.

Findings and proposed solutions for realized errors or pain points have been partially implemented in previous and on-going iteration rounds. Anyway, some of the findings, e.g. data governance and management, are by nature such that management commitment to renew global processes are required and those are not yet implemented. Due to the scope limitation in this study, work was concentrated to cover only project practicalities with process and design related topics. In the future coming studies scope should be widen to cover also technical aspects and find best practises there to support SaaS application implementation globally.

REFERENCES

Bibi, S., Katsaros, D., Bozanis, P. 2012. Business Application Acquisition: On-Premise or SaaS-Based Solutions?. IEEE Software 29 (3), 86-93.

Business Application Research Center BARC. 2018. Data Governance – Definition, Challenges & Best Practises. Read on 23.9.2018. https://bi-survey.com/data-governance

COBIT 4.0. 2005. United States. IT Governance Institute.

Coghlan, D., Retrieving Action Research as Research in OD. Organization Development Journal 35 (2), 11-15.

The DAMA Guide to The Data Management Body of Knowledge. 2010. NJ: Technics Publications.

DAMA-DMBOK: Data Management Body of Knowledge (2nd Edition). 2017. Basking Ridge, NJ: Technics Publications.

DAMA-DMBOK2 Framework. 2014. The Data Management Association. Released 6.3.2014. Printed 11.10.2018. https://dama.org/download-dmbok2-framework

Delez, T. 2018. Data Governance explained in one slide. Released 27.7.2018. Printed 23.9.2018. https://www.fact-international.com/blog/2018/7/27/the-one-slide-data-governance

Delez, T. 2018. Poaching your way through data jargón. Released 19.9.2018. Printed 23.9.2018. https://www.fact-international.com/blog/2018/9/19/poachingjargon

Dine, S. 2015. Business Driven vs Data-Driven Data Model Design. Read 16.7.2018. https://www.eckerson.com/articles/business-driven-vs-data-driven-data-model-design

Doan, A., Halevy, A., Zachary, G. 2012. Principles of Data Integration. Morgan Kaufman, MA:Elsevier.

EU General Data Protection Regulation (GDPR): An Implementation and Compliance guide. 2017

Gartner. 2009. Gartner Highlights Five Attributes of Cloud Computing. Read on 26.8.2018. https://www.gartner.com/newsroom/id/1035013

Hai, H., Sakoda, S., 2009. Saas and Integration Best Practises. Fujitsu Sci.Tech Journal 45 (3), 257 – 264.

Henderson, J., Venkatraman, N. 1993. Strategic alignment: Leveraging information technology for transforming organizations. IBM Systems Journal 38 (1), 472-484.

Hurwitz, J., Bloor, R., Kaufman, M., Halper, F. 2009. Cloud computing for Dummies. Hoboken, NJ: Wiley Publishing.

ITIL Service Operations. 2007. Great Britain: The Office of Government Commerce.

ITIL Service Strategy. 2007. Great Britain: The Office of Government Commerce.

ITIL Service Transition. 2007. Great Britain: The Office of Government Commerce.

Kariazoglou, J., 2010. IT Strategic and Operational Controls. UK: Cambrigeshire, IT Governance Publishing.

Kidd, C., 2018. Data Governance vs Data Management: What's the Difference?. Released 2.5.2018. Printed 23.9.2018. https://www.bmc.com/blogs/data-governance-data-management/

Koehler, J., Hauser, R., Kuster, J., Ryndina, K., Vanhatalo, J., Wahler, M. 2008. The role of Visual Modeling and Model Transformations in Business-driven Development. Electronic Notes in Theoretical Computer Science 211, 5 - 15

Ladley, J. 2012. Data Governance: How to Design, Deploy and Sustain an Effective Data Governance Program. Morgan Kaufman, MA: Elsevier.

Lambert, B. 2017. Understanding the New European Data Protection Rules. Auerbach Publishers Inc.

LeBlanc, K. 2018. Europe's GDPR to Set New Standards. Information Today. 35 (5), 24.

Marinescu, D.C., 2015. Cloud computing: Theory and Practise. Morgan Kaufman, WA: Elsevier.

Masters, J., 1995. The History of Action Research. Action. Reserarch. Electronic Reader. Read on 12.8.2018. http://www.aral.com.au/arow/rmasters.html

Maxwell J. 2005. Qualitative Research Design: An interactive Approach. Second Edition. USA: Sage Publications

Regulation (EU) 2016/679. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). Read 2.9.2018. https://eur-lex.europa.eu/eli/reg/2016/679/oj

Saunders, M., Lewis, P., Thornhill, A. 2009. Research methods for business students. Pearson 5th ed.

Sebastian-Coleman, L., 2018. Navigating the Labyrinth: An Executive Guide to Data Management. Basking Ridge, NJ: Technics Publications.

Simsion, G., Witt, G. 2005. Data Modeling Essentials. Amsterdam: Morgan Kaufmann.

Snyder, B., Ringenberg, J., Green, R., Devabhaktuni, V., Alam, M., 2015. Evaluation and design of highly reliable and highly utilized cloud computing systems. Journal of Cloud Computing. 11 (4). https://journalofcloudcomputing.springeropen.com/articles/10.1186/s13677-015-0036-6

Software as a service market (SaaS) growing at a CAGR of 28.3% to 2025 - microsoft, SAP SE, ADP, workday, amazon, salesforce. 2017. M2 Presswire. Read on 25.8.2018. https://search.proquest.com.elib.tamk.fi/business/docview/1933083850/fulltext/4C02586D2A484B2APQ/1?accountid=13208

Tilak, M. 2005. Business-driven development. IBM DeveloperWorks. Printed 28.10.2018. https://www.ibm.com/developerworks/library/ws-bdd/ws-bdd-pdf.pdf

Tupper, C. 2011. Data Architecture: From Zen to Reality. Amsterdam: Morgan Kaufmann.

Wang, L., Ranjan, R., Chen, J. & Benatallah, B. 2017. Cloud computing; Methodology, Systems and Applications. Boca Raton, FL: CRC Press.

Wright, M. 2018. What is GDPR, and why should you care. Independent Banker 68 (10), 35 – 37.

APPENDICES

Appendix 1. Best practises for integration set-up

1. Do not underestimate time needed for integration set-up

2. Even existing integrations need investigation, project should not give statement that nothing will be changed. Also, business needs to understand that always it will not be possible to get new application work exactly similar way as previous system.

3. Process harmonization between different countries or businesses should be considered. With harmonized processes also common integration design can be built.

4. Agile development does not mean developing without proper planning and requirements. Agile should be treated more like mindset, not methodology.

5. Integration planning to be started same time with actual functional planning. Integration planning is not just mapping document but includes also understanding of business processes and impact to integration. Only mapping document is not relevant, because same data element can have different meaning in different applications.

6. Ensure all business scenarios have been defined. Many of scenarios do have implications to the integrations.

7. Ensure knowledge of enterprise level target architecture for the integrations. Ensure global level validator availability in integration related architectural decisions.

8. Ensure integration related data quality together with business representatives.

9. Before migrating data from previous system, ensure enough time and resources for the data cleaning activities.

10. Do not migrate data just based on field name, ensure overall understanding of data fields in new system. Incorrect data content can impact to integration reliability.

Appendix 2. IT Lead checklist for integration implementation

1. Define main rules for data transfer, e.g. point-to-point integrations or one integration from source system.

2. Define ESB role – should role include data enrichment, calculation or just transfer the data.

3. Ensure adequate resources for integration planning and building (source system, target system, ESB).

4. Ensure adequate technical understanding of integration build already during the project (e.g. training possibilities for architects) to avoid incorrect solutions when building new integrations. Especially important if several different integration tools or practises exists.

5. Master data should not be enriched in middleware or target system if that is avoidable. Source system should have responsibility of the data including requirements from the target systems.

6. Check how target system is using the data, especially if data contains personal or sensitive data.

7. Ensure GDPR related data is not included to (existing) integrations, if not really needed in target system. Due to GDPR, only needed relevant data can be part of integration.

8. Ensure end-to-end integration documentation is created. Quite often different parties are doing integration definition only from their point of view and big picture not defined. When later changes are realized (e.g. in cases of mergers, acquisitions), need to understand whole chain and possible customizations in each step. Ensure documentation includes possible data enrichments done in the middleware.

9. Agree physical location (archive) for the integration documentation. Documentation should be available for all stakeholders.

10. Ensure documentation process also after the project go-live. Need to ensure document to be up to date after future enhancements. Knowledge transfer to responsible person after the project to ensure documentation quality.

11. Identify all business scenarios where customizing has been done and are impacting integrations. Based on this list to be investigated if regression testing is needed in case of version upgrade changes this functionality.

12. Ensure testing tool usage for integrations. Test library, test execution, test evidences, defect management and test management reporting via tool only.

(continues)

13. Ensure adequate test scenarios impacting integrations. To be confirmed by business that all business specific, country specific etc. process scenarios are considered. Also, possible regression test cases to be defined by IT supplier. Prioritize scenarios according business and service risk in case there is no time to execute or do corrections to all.

14. Reserve enough time for testing to be able to test all integration scenarios, not only most common ones.

15. Ensure testing tenant usage without refreshes during the testing. If tenant refreshes are fixed, to be checked that no integration tests are in the middle of execution during the refresh.

16. Integration volume testing should be done. Individual transactions with enrichment will go nicely but when volumes include thousands of transactions, there is possibility to have performance issues.

17. Check if testing data includes personnel data which should be considered from data protection regulations point of view and if needed, organize different archiving for integration testing evidences.

18. Check authorizations to test environment if data scrambling is not done after refresh. If data contains any personnel data and data scrambling is not in place, accesses need to be limited to same level as in production.

19. Define acceptance criteria for integration testing. Define when deployment to production is possible without full integration functionality. Identify what missing scenarios can be finalized when integration is already in production.

20. Ensure ITSM processes are followed also related to integration changes. If business development organization has been established, also they should follow ITSM way of working, incl. demand and deployment process, release management, ticketing tool usage.

21. Ensure business development organization has adequate process for knowledge transfer to AMS-partner.

22. Create RACI matrix to support responsibility understanding in business development organization. Define integration related roles and responsibilities after project go-live.

23. Define supplier ticketing tool usage process also in case of integration related issues. Process to be defined how to combine two (or more) ticketing tools if corporate is having own tool. Process definition should include also sub-process to compare and match tickets and their content in different tools.

(continues)

24. Plan and set up release management process both for functional and integration related changes. To be noted that there are two types of releases; supplier regular upgrades and own enhancements. Both types of releases need also knowledge transfer to AMS partner.

25. AMS partner need to understand business processes related to integration, not only technical content. Critical in cases where business processes are affecting the way how application transfers data forward.

26. Ensure process to investigate SaaS provider regular upgrades to identify when any actions or regression testing is needed for integrations.

27. Define future development model and responsibilities to avoid confusion and to guarantee efficient handover to AMS partner. Definition should include process to identify future development in source system structures and ensure there will be enough time to do similar changes in integrations.

28. Ensure development partner knowledge of integration build basics. If new developer will be hired, it will be partner responsibility to do needed documentation and give needed knowledge transfer sessions.

29. Define process for changing business processes. Business rule changes are not always incorporated to system and from there to integration. There needs to be process in place to notify coming changes and possible consequences.

Appendix 3. Data identification and responsibilities

1. Data governance rules and practises to be implemented. Cross functional commitment and coordination need to be in place.

2. All changes to data elements should have communication process in place. All relevant parties where data in integrated, should be aware of coming changes.

3. Data quality principles to be followed: criticality (risk level if data is not correct), lifecycle management, data error prevention, root cause analysis, governance, using standards, measurement and transparency

4. Errors from previous system master data should not be migrated to new system, careful cleaning is needed.

5. Avoid workarounds with data. Data quality in source system is essential and if adequate outcome is result of workaround, it can impact next integration who needs similar data.

6. Differences between data elements in old system and new application to be studied. Data element name can be the same, but actual content differs.

7. Data need to have clear ownership. Owner need to be business or function, it cannot be IT or application. Ownership is not just about securing own needs of the data but have cross-functional commitment and coordination.

8. Roles and responsibilities towards data elements to be defined

9. Data needs to be prioritized and relevance to other systems to be recognized (e.g. regulatory reporting, financial reporting, business policy, ongoing operations, business strategy)