A Systems Approach to the design of a Management Information System for a growing IT service company

Student:	Wouter van den Berg
Student number:	s0171875
Company name:	Ymor
Company supervisor:	Bas Jorritsma (Delivery Manager)

Date: Februari 16th, 2015

UNIVERSITEIT TWENTE.

YMOR

Abstract

The managers of the Delivery department at Ymor desire a new management information system (MIS) that will provide them with information on their department's level of performance. The goal of this research project was to design such a system. I have developed and used a new MIS design method that is based on Mason and Mitroff's work (1973) where they define the five key variables of any MIS. For one of these key variables, namely the organizational context of the MIS, I have used a different model than Mason and Mitroff: the Viable System model (VSM) by Stafford Beer. The purpose of my method is to help the designer determine the most appropriate value of those five variables in such a way that they form a coherent whole. In this case, it turned out to be necessary to first define 'performance' for the Delivery department to bring more structure to the information needs of the managers. It is hoped that my MIS design method increases our understanding of the processes and goals of MIS design projects, especially in small to medium enterprises.

Key words: MIS design, inquiring systems, systems theory

Preface

I have carried out this research project in partial fulfillment of the requirements for the degree of Master of Science in Industrial Engineering and Management at the University of Twente. I would like to take this opportunity to thank the people who helped me along the way.

I would like to start by expressing my gratitude to the two Delivery Managers at Ymor, Bas Jorritsma and Stephan Badal. They showed a lot of interest in the project and were always willing to assist me with any issues that I encountered. I am especially thankful for the weekly meetings with Bas Jorritsma. I have learned a lot about the management profession during my time at Ymor because of him.

My second word of thanks goes to my supervisors at the University of Twente, Dr. Fons Wijnhoven and Dr. Hans Heerkens. My meetings with them were approximately once a month and I feel that their support was vital to my progress. They have shown patience during the many times that I felt lost and gave me the advice I needed to find my way again. Because of Fons' enthusiasm and knowledge of the field, I never left a meeting without new ideas to explore. Perhaps I have even spent a little too much time thinking divergently, but I was lucky to have Hans as my second supervisor who kept asking the right questions to bring my attention back to the matters at hand.

I am also grateful for the other people at Ymor who spent time with me and assisted me during my months at the office. They are too many to mention here, which only proves that the willingness to help is indeed one of Ymor's core values.

Finally, I want to express my love and thanks to everyone at home for their support.

Table of contents

List of figures

Note: I have chosen to use the label 'figure' for every kind of illustration, graph or table that I have added to the text. It is my hope that this makes it easy for the reader to find the right data every time that I have made a cross reference.

Introduction

1.1. Background

The context of this research project was formed by a small to medium-sized IT-service company called Ymor. The mission of Ymor is to assist large Dutch organizations with so-called Application Performance Management (APM). APM is the monitoring and management of the performance and availability of software applications. Ymor provides roughly three different APM services.

The first service is called 'Ymonitor'. Most of Ymor's revenue is earned with this service. It involves setting up a system that monitors the performance and availability of a certain application for the client. 'Performance' is defined as the amount of time the application takes to fulfill a user request (i.e. lower is better). This means that Ymor takes a so-called 'end-user perspective' on the performance of a client's application. Ymor does this by using specialized software to set up computers that simulate user behavior. These computers then measure the amount of time it takes for the application to process each action that the simulated user tries to take. This is done every 15 minutes on every day of the week. The data is then presented to the client in a dashboard that Ymor developed, which is also called Ymonitor. The client can log in on this dashboard and so has access to data on the actual application performance experienced by the end-user. Without this service, these client organizations would have to rely on their own application performance monitoring. The problem with these organization's internal monitoring is that they are often not focused on the end-user experience. Instead, they measure the local performance of subsystems of the IT infrastructure. The underlying rationale of this approach is that if all parts of the system perform well according to their key performance indicators (KPIs), then the performance of the system as a whole must be good as well. Unfortunately, this is often not the case. That is why Ymor provides the Ymonitor service and has done so with success in the past eleven years.

The other two kinds of services that Ymor provides are called 'Yvalidate' and 'Troubleshoot'. Simply said, Yvalidate consists of performance tests that are meant to show the changes of an application's performance under different degrees of data-traffic intensity (i.e. load). The Troubleshoot group of services consists of projects where Ymor's most senior experts are tasked to solve a complex and/or persistent problem with a client's application performance.

In the summer of 2013, the management of Ymor chose a new strategic direction for the years to come. It was decided that the company would not settle with building performance monitoring systems and running performance tests for their clients, only so that the latter can use that data to manage their applications. Instead, the managers at Ymor envision their company as one that plays a much larger role in its clients' APM activities, namely by relieving them of most or all of their APM-worries all together. According to Ymor's management, this means that Ymor doesn't sell performance monitoring or performance tests anymore, but rather specific APM results for which they can use monitoring and testing as a means rather than an end.

What was immediately apparent to people at Ymor is that the company needed to grow in order to make the new vision a reality. After all, in order to take over more APM activities of other organizations it is necessary to have enough people to do the required work. However, it's not only a matter of *more* work but also of a *different kind* of work: if Ymor wants to start emphasizing their role as APM consultant to their clients in addition to their traditional role as APM engineer, new kinds of expertise are required. Examples are: change management, IT strategy, business process management, et cetera.

In short, Ymor is in a state of transformation. Thanks to the new strategy and the resulting policies, new people are joining the company every month. These developments form the background of this research project.

1.2. The problem situation

I now turn to the problem situation that this research project was meant to solve. I have included a visualization of the cluster of problems at Ymor in order to describe the situation effectively, see Figure 2. Most elements in that figure represents a problem and the arrows indicate a cause-effect relationship between them. The problems have been numbered in order to make referencing more easily. Some elements are just effects and not necessarily problematic. These are added to the figure to clarify the cause-and-effect chain.

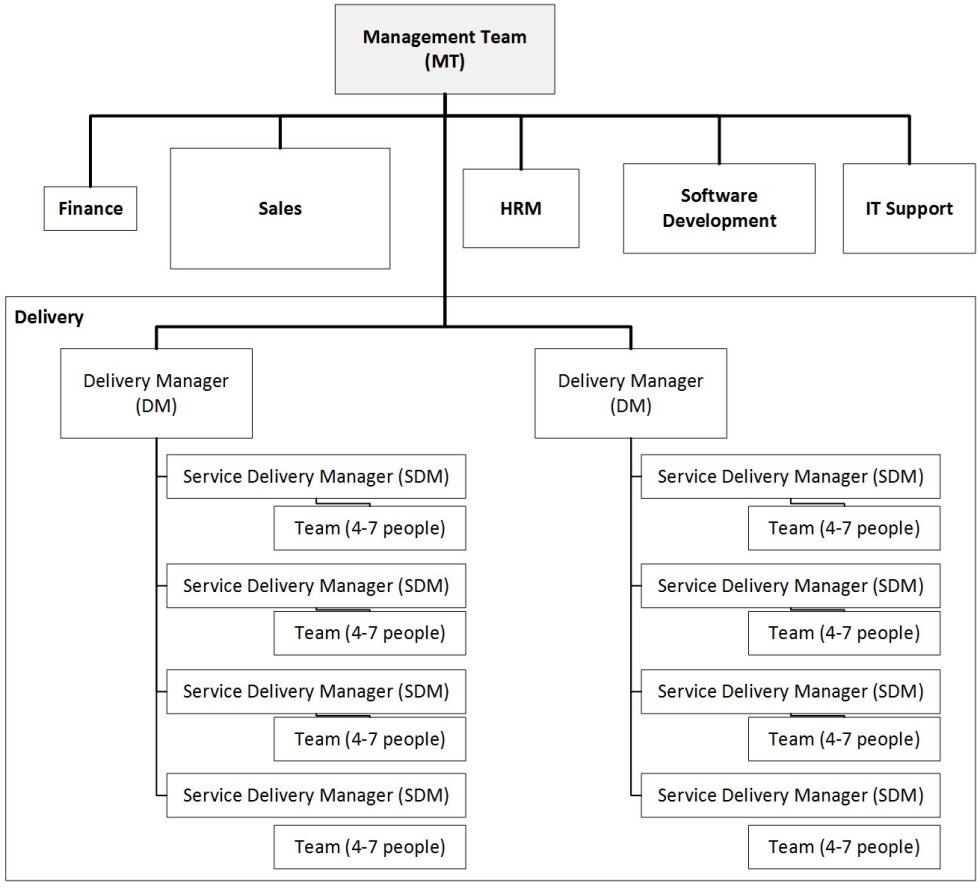


Figure 1 – A simple organizational chart of Ymor, focused on the Delivery department

The figure may look complex at first, but there are three themes that run through it. I used three colors (purple, green, yellow) to indicate them. Hopefully this increases its readability. The yellow problems all have to do with under-capacity and growth of the Delivery department. The majority of people at Ymor work for this department: they deliver Ymor's services. Other large departments are Sales and Software Development. See Figure 1 for an organizational chart. Going back to the problem cluster in Figure 2, the green problems are related to issues with management control of the Delivery department, while the purple problems concern the organizational design of Ymor as a whole. Finally, the problems in white are about the general business results and company strategy. As the figure indicates, the three colored groups of problems together cause the white group at the top.

I will now move on to short discussions of each colored cluster.

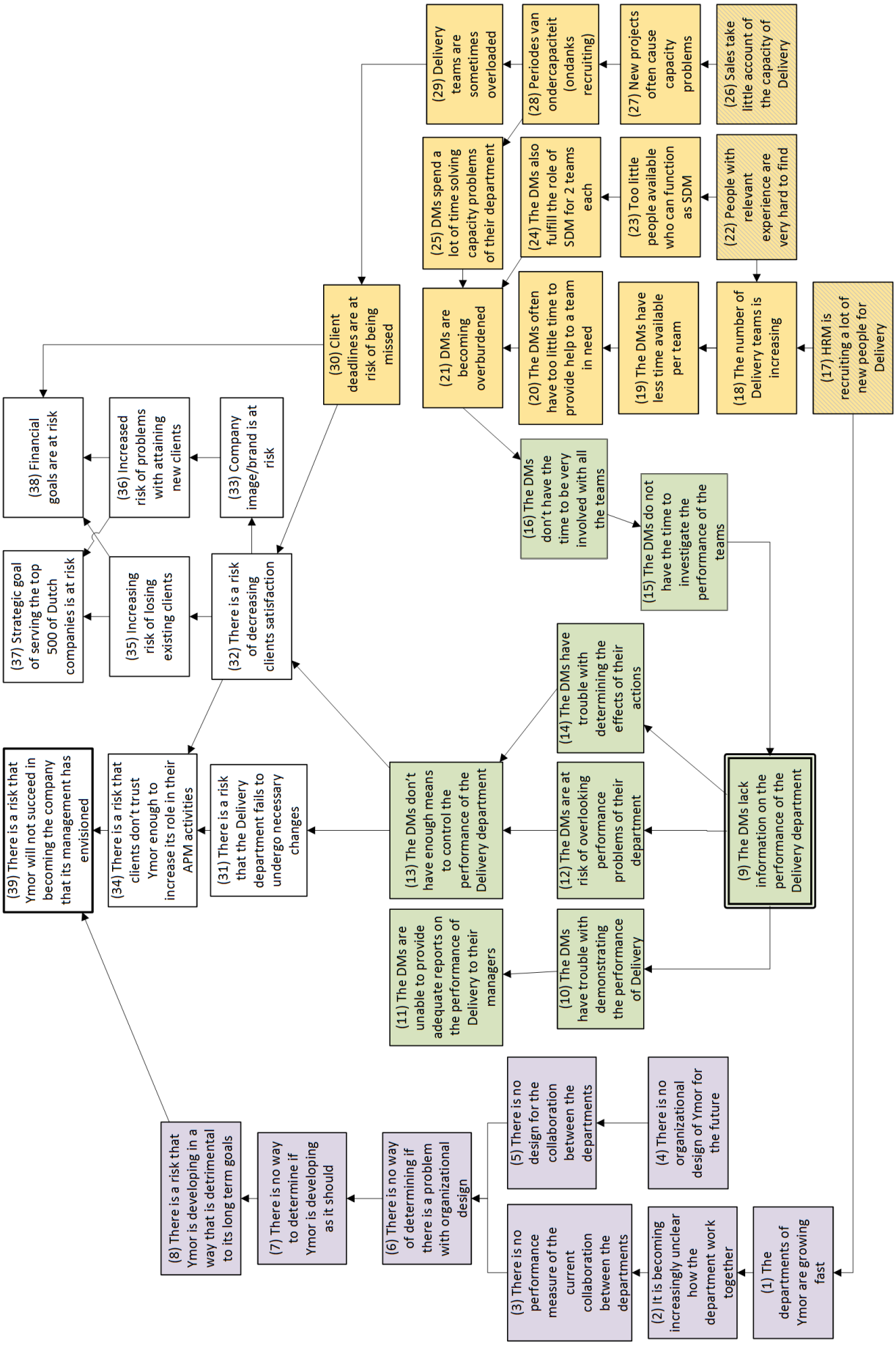


Figure 2 – A visualization of the cluster of problems that I observed at Ymor

1.2.1. *Under-capacity of the Delivery department (yellow)*

At the bottom of this cluster of problems one can find element #17, which says: "HRM is recruiting a lot of new people for Delivery". Of course, this fact is not inherently problematic. In fact, as I established in the previous paragraph, growth is part of Ymor's new strategy. But as the figure shows it does cause a number of problems. Before I go over them, I would like to point out that this element has a striped pattern that makes gives it a darker yellow. This is meant to indicate that this fact or problem lies out of my sphere influence. This also goes for the other elements with this pattern in the figure. Thus, element #17 can be read as "HRM is recruiting a lot of new people for Delivery *and there is nothing I can do about that*".

Because of the growth of the Delivery department, the number of Delivery teams has been increasing significantly. Every team has a manager, who is called Service Delivery Manager (SDM). The next management layer consists of two Delivery Managers, who are responsible for the Department as a whole. Because people with relevant experience are very hard to find (#22), there are not enough SDMs to cover every team (#23). That is why the two DMs are each required to play the role of SDM for two teams (#24). In other words, in addition to their role as DM, they are required to act as SDM. Fortunately, these DMs have more than enough previous experience with this role. However, as the number of Delivery teams increases, so does the load of their DM-responsibilities. The DM have less time to spend per team and often have to work extra hours to help teams in need (#19, 20). It should be no surprise, then, that the combined responsibilities of being DM and SDM (for two teams) is making them overburdened (#21).

What makes this problem worse is that Sales takes little account of the capacity of the Delivery department (#26). Their goal is simply to sell as much projects/services as possible. The upper management of Ymor is aware of this, but gives higher priority to Sales' goals and therefore has not intervened. Because the demand for the services of Delivery is uncertain and non-deterministic, this results in periods of under-capacity for the Delivery teams (#27, 28, 29). This in turn causes the DMs to spend a lot of time solving capacity problems, either between teams or for the department as a whole (#25). Although they can often work something out, it is not uncommon that client deadlines are missed (#30).

1.2.2. *Organizational design of Ymor as a whole (purple)*

Not only the Delivery department is growing, all of them are (#1). Collaboration between the different departments is becoming more complex now that the company is growing. The result is that it is becoming increasingly unclear how this collaboration is functioning, and if its functioning well (#2, 3). Furthermore, there is no organizational design for Ymor in the future (#4), which means there's also no design of how the collaboration between the departments should develop on the long term (#5). The result of the above is that there is no way of determining if there is a problem with organizational design (#6), which means its also not possible to determine if Ymor as a whole is developing as it should (#7). This introduces a risk that Ymor is developing in a way that's detrimental (or at least not beneficial) to Ymor's long term goals (#8).

1.2.3. *Management control of the Delivery department (green)*

Next to problems with capacity (yellow) and a apparent lack of organizational strategy or plan (purple), there are some issues with the management control of the Delivery department. As Figure 2 shows, these problems have come to the surface because of the company's growth, i.e. the yellow part. The DMs do not have enough time to be closely involved with all the teams (#15), and hence are not able

to frequently observe their performance (#16) like they used to, e.g. through attending their team meetings and conversation. This causes problem #9: the DMs lack information on the performance of the Delivery teams and hence on the performance on Delivery as a whole. This problem in turn effects the situation in multiple ways. Firstly, because of this lack of information the DMs have trouble with communicating about the department's performance (#10). This goes for communication among the DMs themselves but, more importantly, also for their reports to top management, i.e. the Management Team (#11). Secondly, without the required performance information the DMs are running a risk of overlooking performance problems (#12). The result is that they lack the means to adequately control the performance of their department (#13). The third point also adds to the control issue: without performance information the DMs have no way of determining the effects of their interventions (#14).

As Figure 2 shows, these problems pose a risk for the company goals. Without adequate management control, the Delivery department runs the risk of not undergoing the changes that are necessary to achieve the long term goals (#31, 34, and 39). In fact, if the company keeps growing and management control of Delivery becomes even more problematic, the performance of the company as a whole could well decrease (#32-38).

1.3. Choosing the core problem

With the problem situation explored, it is now time to choose the core problem. The core problem is a problem that is far down in the causal chain of the problem cluster. Solving this problem should therefore dissipate the problems that it caused. The core problem should also be feasible, both technically and socially. This means that the core problem must be agreed upon with the research client, which in this case is Ymor (or more specifically, one of the DMs at Ymor).

The problems at the bottom of Figure 2 are all candidate core problems, i.e. #4, 9, 17, 22 en 26. However, the latter three (which are the yellow ones) all have the striped pattern that indicates that they are outside of my sphere of influence. This means that the remaining candidates are

- #4: there is no organizational design of Ymor in the future
- #9: the DMs lack information on performance of the Delivery department

The result of the former is that Ymor is growing without a clear idea of the desired form, which introduces the risk that the company is growing in ways that is detrimental for the strategic goals. The result of the latter is that the DMs lack adequate control over the performance of their Delivery department, which ultimately also poses a risk to Ymor's attainment of its strategic goals.

After I presented these two options to my supervisor at Ymor, who is one of the two DMs, we concluded that the first option should be dropped because it is the responsibility and wish of the Management Team to solve that problem. Which left us with only one core problem candidate, i.e. #9. This problem was therefore chosen as the core problem that would be attempted to solve through this research project. I have shown this in the figure by using a thicker outline on the element, to stress its importance.

1.4. Assumptions about the solution

Both the DMs already had an idea of how to solve the core problem described above. Their plan was to provide themselves with more information about the performance of the Delivery department by

introducing some kind of management information system that measures and monitors four so-called *key performance indicators* (KPIs). While there may be numerous other ways to approach this problem, I decided to go along with theirs because there was already a lot of devotion to this idea at Ymor and there seemed no glaring problems with it.

The DMs had already selected the following four KPIs:

- The degree of capacity utilization
- The deviation of the time budget, which is in hours
- The average of the grades for project success given by the clients
- Employee satisfaction

The last KPI seemed to be more like a variable that the DMs were interested in instead of an actual indicators, but more on that later. In any case, this was the list that the DMs had formed prior to this project. Capacity utilization of the Delivery department refers to the degree to which the employees use their time working on projects for the clients. This is where the revenue comes from so it is important that not too much time is spent on other internal activities such as meetings and whatnot. The measure can also be used to distinguish situations where under-capacity is caused by too many projects from situations where under-capacity is the result of too many internal activities.

The second KPI concerns the degree to which the amount of hours used by the Delivery teams to complete a project differs from the amount that was budgeted. A high difference can mean either that people spent more time than planned, or less. Less would indicate a higher efficiency while more would mean a lower efficiency.

The third on the list is a measure of project success, which is the part that the Delivery department plays in attaining a key performance variable on the level of the whole company: customer satisfaction. In other words, project success and customer satisfaction are not the same. The former is part of the latter. Project success is the responsibility of the Delivery department, while all departments share the responsibility for general customer satisfaction. Note that even at Ymor there seems to be some confusion about the distinction between the meanings of these terms. In any case, the goal for this KPI is achieving a grade of 8 or higher on a scale of 1 to 10.

The fourth item on the list is employee satisfaction. As I have said, it is not really a KPI but a key performance variable. The measurement of this variable was outsourced in Q4 of 2014, shortly before the end of this research project. The DMs state employee satisfaction is important because they are convinced it has a strong, albeit indirect influence on project success. The precise nature of this supposed causal relation has not been made entirely clear, but common sense informs us that it involves higher motivation by the employees to achieve the best possible results for the clients and the company.

It seemed uncertain whether this list of KPIs was complete or not. That is why an important part of this research project has been the identification of other important KPIs, which I will describe in a later chapter. Whatever the KPIs would be, it was clear at this point that the end-point of this research project would be a design of a management information system that implements these KPIs and so provides information on the performance of the Delivery department to the DMs.

1.5. Goal and research methodology

Until now I have only described the practical goal of this research project, but of course there is also a scientific goal. This means this project should not only result in a design but also in new insights, i.e. new knowledge. But new knowledge about what?

According to Hevner et al. (2004, p. 76), this could be one of things: a construct, a methodology, a model or an instance. Constructs consist of new notations or symbols. The example that Hevner et al. use is the well-known Entity-Relationship notation that is used to model the structure of databases. In other words, constructs form the language with which we describe reality and design our solutions. These descriptions are 'models', the second category. In other words, we use constructs to make models. For example, using the Entity-Relationship notation (a construct), we can make an Entity-Relationship diagram (a model) that describes the structure of a certain system or a type of systems. The third category is 'methods'. Methods are specifications of design processes. Of course, methods are very useful because, for example, they inform us how to use constructs to effectively make models. But methods can also specify how to design an instance. 'Instance' is the fourth category described by Hevner et al. In this context, an instance is simply an implemented (and hopefully working) information system. In other words, research projects that focus on an instance are meant to provide new insights about existing systems.

With these four categories, I can now restate the goals of this project. As stated before, the first goal was to design a management information system that provides the DMs with their desired information on the performance of their department. The actual implementation of this system was outside of the scope of this project. Or in the words of Hevner et al.: the result was to be a model, not an instance.

The second goal was to develop a new method for information system design and demonstrate its usefulness by applying it during this design project. This has been the scientific goal of this project. This means that the design of the management system for Ymor had two purposes: firstly, to solve a problem at Ymor, but secondly, to gain new knowledge about the general usefulness of this new method for information system design.

1.6. Research questions

Two research questions will have to be answered during this research project. The first is concerned with the actual design of the management information system (MIS) for Ymor:

Research question 1) *What are the functional requirements of the MIS for the Delivery department at Ymor?*

The second research question focusses on the scientific goal of this research project. It reflects the wish to learn more about how MIS design projects should be approached. The question is as follows:

Research question 2) *To what extent is this design method useful for the design of MIS in other contexts*?

1.7. Structure of this report

In chapter two, I introduce the theoretical framework that I employed to derive my MIS design method. The basis of this framework is formed by the model of Mason and Mitroff (1973). My design method uses their model of MIS in conjunction with the Viable System Model (VSM) of Stafford Beer

and the Balanced Scorecard (BSC) of Kaplan and Norton. All three of these models are briefly described.

Chapter three is called ‘Analysis’. In this chapter, I describe my attempt to answer the first research question, which is concerned with finding the functional requirements of the system under design. The first paragraph of this chapter contains a description of the steps of my MIS design method. This is only a small part of the chapter. In the two paragraphs that follow, I illustrate how I applied my method and what the result of each steps was. I conclude the chapter with an answer to the research question.

After revealing the functional requirements, it was time to detail the actual design. I have described the design in chapter four. The first paragraph contains a discussion of the KPIs and of how the system will measure them. In the next paragraph I have specified the data structure of the system under design using Entity-Relationship modeling. The third paragraph describes the necessary information flows between the system and its supposed environment. Finally, in the fourth paragraph, I have given examples of the management reports that the system should be able to produce. This concludes the design of MIS for the Delivery department of Ymor.

The last chapters are chapters five and six. The former contains numerous recommendations to the DMs of Ymor concerning the implementation, use and future expansion of the designed system. The latter, i.e. chapter six, contains a discussion of the design process of this project and concludes with an answer to the second research question. This means that a few statements are made concerning the general usefulness of my design process.

2. Theoretical framework

The design method that I developed is based on other models and theories. Together, these form the theoretical framework of this project. The first paragraph contains an introduction to the Mason and Mitroff's (1973) model of management information systems (MIS). This model describes five key variables that together compromise a MIS, and provides the possible values of each variable. The authors presented this model as a basis for MIS-research, but I took it as a basis for a MIS design method. How and why I did this is discussed in this paragraph as well.

One of these five key variables is the organizational context of the MIS. For this variable I defined a set of possible values that is different from the set that Mason and Mitroff originally presented. The reason for this is that their distinction of organizational contexts seemed a bit too general and therefore unsuitable for a smaller company such as Ymor. I will return to this point later. The model that I used to distinguish a different set of possible organizational contexts is the Viable System Model (VSM). It will be introduced in the second paragraph. The VSM is a powerful cybernetic model of control in organizations. With it, one can distinguish the organizational context of the MIS (to be designed) in much more detail.

The third and last model that is introduced in this chapter is the Balanced Scorecard (BSC) model. It became apparent during my research at Ymor that it was necessary to explore and define the meaning of the 'performance' of the Delivery department in more detail. (I explain the reasoning behind this in chapter 3, section 3.2.5.) That is why paragraph three contains a discussion and comparison of three alternative business performance models, of which one is the BSC model. The choice for the BSC model is justified at the end.

Finally, the fourth paragraph concludes with a short summary of the whole theoretical framework and the role each model plays in it.

2.1. Five key variables of MIS

In their article from 1973, Mason and Mitroff describe the five key variables that according to them compromise a management information system, or 'MIS'. This is their list (1973, p. 476):

1. The **psychological type** of the users of the system;
2. The **method of evidence generation** (or: the nature of the guarantor of evidence);
3. The **organizational context** within which the MIS operates and problems occur (see below)
4. The **class of problems** that the system is supposed to provide information about
5. The **mode of presentation** of evidence (i.e. information) by the system.

Each variable can have different values. For example, Mason and Mitroff distinguish four psychological types and according to their model that variable has four possible values. Furthermore, the authors state that there are different degrees of compatibility between values of the variables. To give another example: certain methods of evidence work better with certain psychological types. This means that a good MIS design process gives form to these variables in such a way that they work together as best as possible.

In each of the following sections I describe the possible ways a variable can be given form according to Mason and Mitroff, starting with the first in the list above and proceeding in numerical order. At the end of this paragraph, I explain why this model forms the basis of my MIS design method and also why I replaced part of it with the Viable System Model, or ´VSM´.

2.1.1. *Psychological types*

Mason and Mitroff refer in their article to the four personality types defined in Myers (1962)[1], which is a Jungian typology. They write that these four types are distinguished based on a) how they perceive the world, and b) how they evaluate their perceptions. For both characteristics there are two possibilities. When it comes to perceiving, there is the 'Sensing' type versus the 'Intuition' type. When it comes to evaluation, they distinguish the 'Thinking' type versus the 'Feeling' type. Each of the four personality types is a different combination.

Concerning the differences between the 'Sensing' and 'Intuition' types, Mason and Mitroff write the following:

"*The virtue of Sensing types is that they are guided by the facts and are careful not to extrapolate them, while the virtue of Intuition types is that they see through the facts and extrapolate beyond them (one is reminded of Freud cautioning scientists that in order to see beyond their facts they have to be prepared to ignore them). Whereas the Sensing type may be too data-bound (he tends to go on collecting data forever because he is afraid to risk a generalization that "goes beyond the available data"), the Intuition type may be too data-free; he may spin out a hypothetical conclusion a minute, none of which is based on available data."* (1973, p. 477)

When it comes to the difference between 'Thinking' and 'Feeling', Mason and Mitroff state:

"A Thinking individual is the type who relies primarily on cognitive processes. His evaluations tend to run along the lines of abstract true/false judgements and are based on formal systems of reasoning. A preference for Feeling, on the other hand, implies the type of individual who relies primarily on affective processes. His evaluations tend to run along personalistic lines of good/bad, pleasant/unpleasant, and like/dislike. Thinking types systematize; feeling types take moral stands and are interested and concerned with moral judgements." (1973, p. 477)

The point is that the answer to the question 'what is information?' depends on the psychological type of the person that is being asked. This is the reason why this variable is considered a key aspect of a MIS.

Finally, it is important to stress that the model is a relatively simple description of a very complex reality. These types are not accurate descriptions of real persons. Nonetheless, Mason and Mitroff state that the four types provide a useful heuristic tool for the study (or design) of MIS.

One of the variables that is influenced most by the psychological type of the MIS users, is the method of evidence generation.

2.1.2. *Methods of evidence generation*

The purpose of a MIS is to provide the user (i.e. the manager) with evidence (i.e. information) to decide on a course of action (Mason & Mitroff, 1973). There are different ways of generating evidence. Which one should be chosen depends both on the class of problems that the user faces, as well as his or her psychological type (see previous section). Mason and Mitroff formulate it as such:

"A manager will tend to place his reliance on some methods of generating evidence to the exclusion of others because for him the "guarantees" that the evidence produced by these "inquiring systems" is true are much stronger. Some managers, for example, will rely on observations of basic events (e.g.

[1] This model has since been developed and expanded, but the core seems to be unchanged. See for example Bayne (1995).

accounting data); others will rely on abstract reasoning from basic premises (e.g. O.R. models); still others will seek clarification through debate. Consequently an important factor in MIS design is the type of evidence generating system used and the type of guarantees behind it." (1973, p. 480)

For the different methods of evidence generation, Mason and Mitroff refer to the five 'inquiring systems' of Churchman (1971). Each of these is based on a different epistemology and is named after the philosopher that supported it. The five inquiring systems of Churchman are:

1. The *Lockean* inquiring system
2. The *Leibnizian* inquiring system
3. The *Kantian* inquiring system
4. The *Hegelian* inquiring system
5. The *Singerian* inquiring system

Again, which of these inquiring system is most suited depends on the psychological type of the users and the class of decision problems that they face.

The Lockean inquiring system is rooted in empiricism. They focus on the use of data to make truthful statements about the world by means of induction. This assumes an objectively, knowable world or truth that exists apart from human experience. According to this kind of inquiring system, information about the world is true if a community of people agree on its truthfulness. A common example of implementations of the Lockean inquiring system are the IT systems that make use of databases to answer inquiries. Mason and Mitroff state that Lockean inquiring systems are best suited for structured problems, i.e. where there is a strong consensus between those involved about the nature of the problem (1973, p. 481).

The Leibnizian inquiring system, in contrast, is rooted in rationalism, which has been defined as 'any view appealing to reason as a source of knowledge or justification' (Proudfoot & Lacey, 2009). Leibnizian inquiring are not focused on data but construct so-called fact nets about the world using the rules of logic and reason. The guarantor of truth in these systems is the precision of the statements and their mutual logical consistency. In practice, Leibnizian inquiring systems are characterized by the use of models and calculation of (optimal) solutions to structured problems. An example of an implementation of a Leibnizian inquiring system is the type of systems insurance companies use to calculate the insurance premiums of their customers. According to Mason and Mitroff, Leibnizian inquiry is best suited for structured problems for which a solution can be calculated (1973, p. 481).

Next is the Kantian inquiring system. This type can be seen as a combination of the previous two. The central idea behind a Kantian inquiring system is that giving meaning to data implies that a model has been built in the inquiring system *a priori* (Churchman, 1971). The labels on the data could not have meaning without such a model. That is why the Kantian inquiring system is a multi-model inquiring system (Mason & Mitroff, 1973, p. 481). For every problem, the system generates at least two alternative models that represent the problem in different ways. The decision maker (i.e. the user of the MIS) then compares each perspective on the problem and determines which one is the best. In this case, the 'best' model is the one that the data agrees with the most. In other words, for Kantian inquiring systems a strong match between data and theory is the guarantor of truth. Examples of implementations of Kantian inquiring systems can be find in situations where people from different disciplines (i.e. with different world views) are working together. With such systems, each of the members of the multidisciplinary team makes their view on the nature of the problem known to the others as best as possible. As Mason and Mitroff put it, the hope is that this enables the problem owner to select a problem representation that is best for his situation. One of the problems with Kantian inquiring systems is that determining what is 'best' in the context of problem-solving can be

problematic. However, thanks to the use of multiple models, this inquiring system is the first that has been discussed so far that Mason and Mitroff deem suitable for problems with an unstructured (or 'wicked') nature (Mason & Mitroff, 1973).

If the different models or representations of the problem are not complementary but conflicting, it is called a Hegelian inquiring system. In other words, in Hegelian inquiring systems, at least two conflicting world views are created on purpose. These are then applied to the same (Lockean) data set. The point of the Hegelian inquiring system is that the same data set is used to support two perspectives that are not only different but even antithetical. In fact, the guarantor of truth in this kind of inquiring system is conflict, because conflict ensures that hidden assumptions about the world (i.e. the world views) are brought to the surface. It is hoped that, when this happens, a synthesis of the two world views becomes possible, resulting in a more encompassing perspective on the problem. In this sense, the Hegelian inquiring systems seems the opposite of Lockean inquiring systems because the latter is based on agreement while the former is based on disagreement (Mason & Mitroff, 1973). For this reason, Hegelian IS are better suited for unstructured (i.e. 'wicked') problems than for structured problems, where conflict seems to be a waste of time.

The fifth and last inquiring system described by Churchman is called the Singerian inquiring system. What exactly characterizes this kind of inquiring system is complex and sometimes unclear. It has been interpreted and described quite differently over the years (James F. Courtney, 2001; James Forrest Courtney, Haynes, & Paradice, 2005; Wijnhoven, 2012). Mason and Mitroff point to this complexity as well but state that the main feature of the Singerian inquiring system is its continuous learning and its adaption through feedback (Mason & Mitroff, 1973). The feedback is based on the degree of agreement between the people involved about the nature of the problem. If there is much agreement, the questions become more precise until disagreement is introduced. In contrast, when there is much disagreement, new variables are ´swept in´ to the inquiring process that can account for the disagreement, thereby creating agreement again. As Churchman states, the process is dialectical because "two opposing processes are at work in the inquiring system. One is the process of defending the status quo, the existing 'paradigm' of inquiry, [while] the other is the process of attacking the status quo, proposing radical but forceful paradigms, questioning the quality of the status quo" (1971, p. 199). The guarantor of truth with this type of inquiring system is the degree of progress that a certain perspective enables. The inquiring system supports the use of all the previous forms of inquiry, depending on the situation and the amount of agreement. In situations where there is lots of uncertainty and disagreement about the nature of the problem, agreement-seeking inquiry is assumed to stimulate more progress because it enables action. However, in situations where there is strong consensus, the Singerian inquiring system prescribes that the risk of deception is too high and progress can only be made by questioning the status quo.

2.1.3. *Organizational contexts*

The third key variable of a MIS is its organizational context. This is the environment within which the MIS is supposed to function. Mason and Mitroff distinguish three possible contexts: 1) the strategic context, 2) the management control context, and 3) the operational context. Within the strategic context of an organization, company goals are determined and the necessary resources for the attainment of these goals are identified. Furthermore, plans are made for acquiring and using these resources to reach these goals efficiently. In the management control context, the focus lies on controlling the execution of these plans and checking whether resources are in fact used effectively and efficiently. Finally, the operational context of an organization is concerned with the actual execution of the planned activities.

This is as far as Mason and Mitroff go when it comes to distinguishing the three contexts from each other. Their descriptions seem to give little direction to MIS design efforts, especially in smaller companies like Ymor where these organizational structures are not as apparent as in large corporations. That is why I have chosen to use the Viable System Model (or 'VSM') instead. This model is introduced in paragraph 2.2.

2.1.4. *Classes of problems*

The fourth key variable of a MIS is the classes of problems about which the users need information to come to a solution (i.e. make a decision). Mason and Mitroff assume that a MIS is only used for decision problems and they define those as follows:

"[A decision problem is] *to choose from among a set of acts $A_1, \ldots, A_m$ that A_i which optimizes (in some sense) the decision-maker's (Z's) return U_{ij} , where U_{ij} is the utility or value to Z of the outcome O_{ij} corresponding to the doublet (A , S_j) where {S_j} is the set of the 'states of nature'."* (1973, p. 479)

Based on this definition they distinguish two classes: structured problems and unstructured problems. A decision problem is structured if the sets of {A}, {O}, {U} en {S} are known. They describe three types of structured problems. The first type is called *problems under certainty*. For these type of problems, the relationship between the choice for an {A} and the occurrence of {O} is deterministic and known. The second type of problems is called problems under risk. For these problems the relationship between {A} and {O}'s are stochastic but also known. Lastly, the third type of problems are called *problems under uncertainty*, and for these problems the relationship between {A}'s and {O}'s ore unknown.

The second class of problems are unstructured (or 'wicked') problems. For these problems, one or more of the sets {A}, {O}, {U} and {S} are unknown or at least not known with enough confidence. This seems to imply that there are varying degrees of wickedness possible between the unstructured problems.

2.1.5. *Modes of presentation*

The last key variable of a MIS is the mode of presentation of information that it supports. Mason and Mitroff distinguish between impersonal and personal modes of presentation. Examples of the former are computer-generated reports, models and graphs. Examples of the latter are (group-)discussions, stories, art and drama. The authors propose that some modes of presentation are more compatible with certain psychological types than others, although they never explain this relation any further. Their point is that the conventional mode of presentation, i.e. computer-generated reports, are not necessarily always the best choice. The designer of a MIS would do well to also consider the other options.

2.1.6. *Conclusion*

Mason and Mitroff's model of a MIS has been used as a basis for my MIS design method. It is chosen for the following reasons. Firstly, there are not many design methods where the underlying epistemology of the MIS is made explicit. Although there are quite a few researchers that consider Churchman's book on inquiring systems to be a seminal work, it does not seem to have been implemented much yet. The design method that I developed and used should be seen as an attempt to change that.

Secondly, I believe that the *simultaneous* consideration of the five MIS-design variables improves the chance that the system will be internally consistent and compatible with its environment, i.e. its users and its organizational context. As I explain at the start of chapter three, the idea that these variables are all interdependent can pose practical problems to the order with which the MIS design steps are taken. But acting like these interdependencies do not exist is likely to pose greater risks of ineffective designs.

However, as I have said in a previous section, Mason and Mitroff's model of MIS is not without its flaws. They only briefly discuss the different possible organizational contexts of a MIS and this typology is quite general. As an attempt to improve on this and make my MIS design method more informed, I have chosen to replace this part of their model with the Viable System Model, i.e. VSM. This model is introduced in the next paragraph.

2.2. Viable System Model (VSM)

The VSM is a cybernetic model of 'viable systems', e.g. organisms, people, organizations, nations, et cetera. It is developed by Stafford Beer in the 1970s. His first version of the VSM was in fact a mathematical model, but later Beer has reworked it into a visual variant. Figure 3 shows the VSM in its most abstract form, while Figure 4 contains a version of the VSM with its usual level of detail.

Stafford Beer worked in the field of cybernetics and was a systems thinker (Ramage & Shipp, 2009). According to some, he was the first to apply cybernetics principles to management (Rosenhead, 2006). The term cybernetics can mean a number of different things, but in this report it is understood as the science of control and communication in systems. *Management cybernetics,* then, is concerned with control and communication in organizations. The VSM is a model of the organizational structure of viable systems and can be used to explain how different control and communication responsibilities are divided over the parts of a company. It can also be used to design new organizational structures (Espejo & Reyes, 2011, p. 110).

As I have said, a key concept in the VSM is the 'viable system'. A system can be defined as a collection of organized parts that together form a whole. A *viable* system is defined by Beer as a system that has an independent existence (or an identity) and can adapt to its environment in order to protect its existence. It is clear that companies and other kinds of organizations in a business sense are all examples of viable systems (although some might be less viable than others).

The simplest version of the VSM is illustrated in Figure 3. It shows the three basic elements of the model: the external environment, and the organization divided into two parts: the meta-system and the operational system. The arrows symbolize interactions between the elements. The forms of the three elements are consistent with the conventions suggested by Beer (1994). The environment is the element at the left with the red '**E**'. It has a somewhat strange shape, which is meant to underline the point that the borders of the environment are not static but dynamic. The environment of an organization can be its clients, customers, service providers, suppliers, et cetera. The element at the bottom right with the blue '**O**' stands for the Operational System and is one of the two basic elements of the organization. The Operational System consists of the parts that 'do the work', i.e. that perform the primary activities of the organization. When the system under examination is a company, the parts that form the Operational System could be departments, teams or individual persons (depending on the size of the company). Lastly, the element at the top right with the green '**M**' stands for the Meta-System and forms the second basic element of the organization. Another name for the Meta-System is simply 'Management'. The term 'Meta-System' is more general and is used because it operates on

a higher level of abstraction and speaks a meta-language that is relative to the Operation System that it controls. The Meta-System can be said to provide a service to the Operational System. It makes sure that the whole is in harmony, i.e. that different activities performed by different subsystems are aligned to each other and serve a common purpose. It is also largely responsible for the adaptability of the whole system because of its ability to observe the environment and anticipate on change.

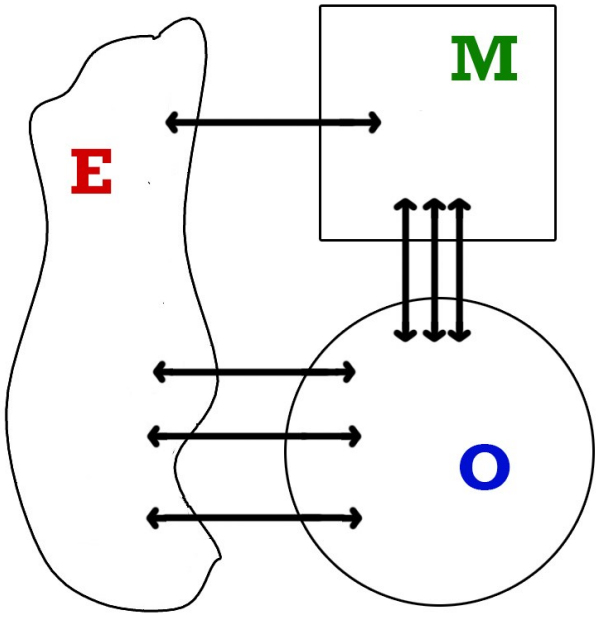


Figure 3 – the VSM in its most abstract form

What makes the VSM special is that it proposes that every viable system (i.e. 'M' plus 'O' in Figure 3) is composed of five subsystems and that *these five subsystems are necessary and sufficient for the whole system's viability*. This means that every part of a company can be seen as part of one or more of these subsystems. Although this is quite a radical claim, I have not find much research that have falsified it. In fact, the model has often been successfully applied to many different kinds of organizations (e.g. Beard & Santos-Reyes, 1999; Schwaninger, 2006; Espinosa & Walker, 2013). This however requires that we give a broader meaning to control and information than the purely rational meaning that cybernetics has traditionally given to these terms. Of course I already did this when I discussed the model of MIS by Mason and Mitroff (see Churchman's inquiring systems in an earlier section).

Figure 4 shows the VSM, like Figure 3 but this time with the five subsystems I mentioned above. The external environment is divided into different sub-environments, of which the future (the one with the question mark) is one. The other sub-environments are different markets, (groups of) customers or suppliers. The Meta-System is shown as a rectangle and contains three of the five subsystems. These are called System 5 (i.e. 'Policy'), System 4 (i.e. 'Intelligence') and System 3 (i.e. 'Control'). The Operational System in Figure 4 is drawn as a circle and contains System 1 (i.e. 'Implementation'). Although System 2 (i.e. 'Coordination') is not within the circle it is actually seen as part of the Operational System as well. Finally, the many lines in Figure 4 indicate flows of information between the five subsystems.

What the functions of these five subsystems of the whole viable system are is discussed later, in chapter 3. What is important here is that these five subsystems are the seen as the possible

organizational contexts of a MIS. In other words, the five subsystems replace the three contexts that Mason and Mitroff described in their article.

The final property of the VSM that I would like to point out using Figure 4 is its recursive-ness. As the image shows, the Operational System actually consists of 'smaller' viable systems, each with their own Meta-System and Operational System. This means that every viable system consists of smaller viable systems and is in turn part of a larger viable system. This is what makes the VSM applicable to all organizations regardless of size. I return to the subject of recursion in the VSM in chapter 3.

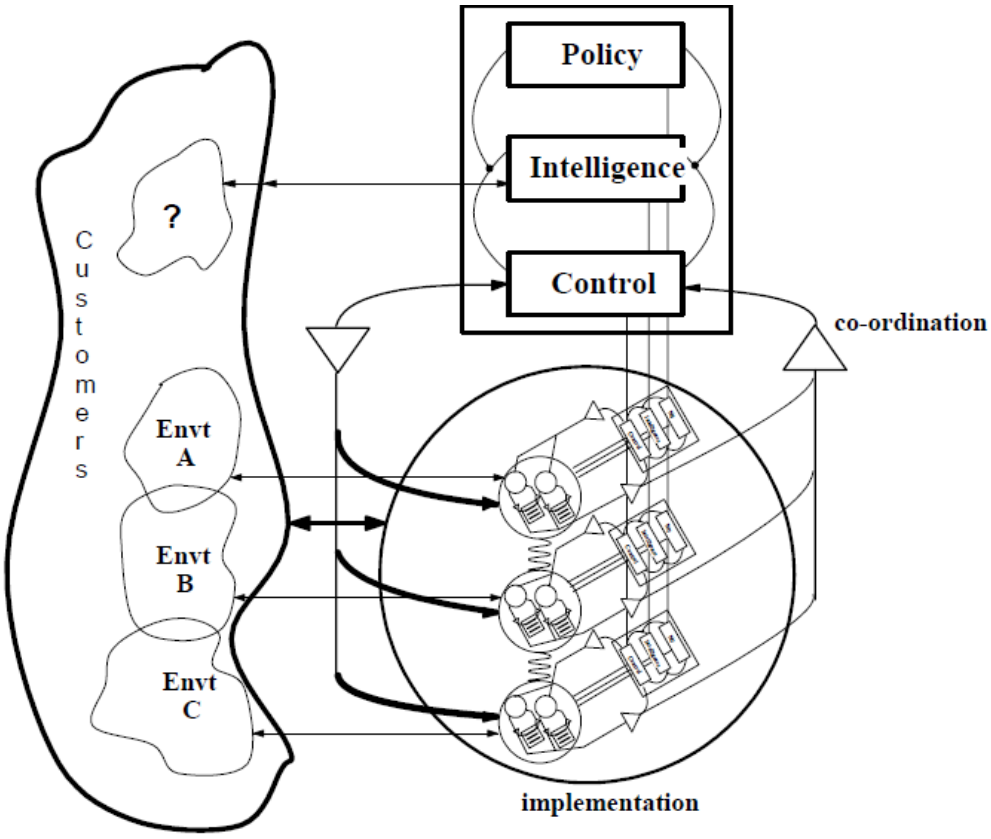


Figure 4 – the Viable System Model with more detail, taken from Espejo and Gill (1997).

2.3. Balanced Scorecard (BSC) model

The goal of this paragraph is to introduce the Balanced Scorecard (BSC) model and to justify the choice for this model over others. The BSC model is a model of performance for companies. It was necessary to use such a model for this research project because it turned out I needed to improve my understanding of what was meant with 'the performance' of the Delivery department. Why is explained further in chapter 3. This is also were I describe how I applied the model in my analysis. The result of this analysis was a list of performance variables that can be used to express the level of performance of the Delivery department.

So the BSC model is a model that globally describes what performing means for organizations. What I mean with global is that it distinguishes four categories of business goals, but only presents guidelines to help determine what the goals in this specific situation are. The performance variables are extracted from these goals and are in turn used to derive key performance indicators (KPIs). The DMs already had made a small list of KPIs (see chapter 1), but the process with which they arrived at this list was unclear. Furthermore, they deemed it likely that the list was incomplete, although they were not sure

what could be missing at that point. For these reasons I decided to perform an analysis with the BSC model to check if the list of KPIs was complete and to add KPIs if it turned out not to be.

Although I used the BSC model, I did consider other alternatives. In the following section, I describe how I determined the list of alternative performance models. I have filtered this list until three models remained, of which the BSC model was one. This was done due to time limits to the research project. The second section of this paragraph contains a description of each of the three models. In the third and final section, the models are compared and the choice for the BSC model is justified.

2.3.1. *Alternatives and filter criteria*

An explorative literature search resulted in a list of alternative models (see Figure 5). Every model describes different categories of performance, although quite a few times the models seem to overlap with each other.

The list was then filtered based on a few criteria. The first was that there had to have been scientific studies that implemented or tested the models after their introduction. The reason for this criterion is that such studies are excellent opportunities to validate the model and learn new things about the assumptions underlying it. Those models that had not been (or barely been) tested or validated since their introduction were left out, because it seemed to me that they bear a higher risk of containing mistakes or validation issues than the ones that have.

The second criterion was that the models should match with service providing companies, or at least be compatible. A lot of the models in Figure 5 focus explicitly on manufacturers and are therefore not suitable for use here. The reason is that manufacturing companies rely on tangible goods and discrete transactions, while service providers such as Ymor are based on intangible goods such as knowledge and information (Vargo & Lusch, 2004). A model for manufacturers would likely emphasize the wrong aspects of Ymor and overlook others.

Name	**Authors**
Performance measurement matrix	Keegan, Eiler and Jones (1989)
Results and determinants framework	Fritzgerald et al. (1991)
Performance pyramid	Lynch and Cross (1991)
Internal/external configuration time framework	Azzone, Masella and Bertele (1991)
Balanced Scorecard	Kaplan and Norton (1992)
AMBITE performance measurement cube	Bradly (1996)
Brown's framework	Brown (1996)
The Cambridge Performance Measurement Framework	Neely et al. (1996)
The Integrated Performance Measurement System	Bititci et al. (1997)
The Dynamic Performance Measurement System	Bititci et al. (2000)
Performance Prism	Neely, Adams and Kennerley (2002)
Integrated performance measurement framework	Rouse and Putterill (2003)
EFQM Business Excellence Model	EFQM (2007)

Figure 5 – a table of performance measurement models found after an explorative literature study

The third criterion was not theoretical but practical: I needed to have access to the articles that described the model. There were a few cases were this criterion meant that a model was left out. Although this was unfortunate, these articles were not cited often so it is unlikely that they met the first criterion.

After applying these criteria, three models remained: the EFQM model, the Performance Prism and the Balanced Scorecard. What follows is a description of each model.

2.3.2. *Description of three performance models*

The models that remained as alternatives were:

1. the Balanced Scorecard
2. the Performance Prism
3. the EFQM excellence model

Every model has been studies in practice at least three times (criterion #1). In addition, each of the three models seems applicable to Ymor because they are not specifically for manufacturers and have applied to service companies in case studies (De Toni et al., 2007).

Balanced Scorecard

The BSC model was introduced in 1992 in an article by Kaplan and Norton (1992). In the years that followed many companies implemented their version of it. According to the yearly survey by Rigby and Bilodeau (2013), 38% of the 1208 companies that they surveyed used the BSC model in 2012. While this percentage seems to be different every year, it is consistently in the top 25 of management tools when it comes to adoption rate. The model also received a lot of attention in academic circles (Marr & Schiuma, 2003; Taticchi et al., 2010).

The BSC model consists of four perspectives on organizational performance. The model prescribes that different KPIs need to be developed for each perspective. The perspectives are:

1. the 'financial perspective'
2. the 'customer perspective'
3. the 'internal processes perspective'
4. the 'learning and growth perspective'

Kaplan and Norton write the following about why they identified these perspectives:

"[the four perspectives] *tell you the knowledge, skills, and systems that your employees will need (their learning and growth) to innovate and build the right strategic capabilities and efficiencies (the internal processes) that deliver specific value to the market (the customers), which will eventually lead to higher shareholder value (the financials)."* (2000, p. 168)

In later works, Kaplan en Norton began to focus on the role that strategy plays in the BSC model (Kaplan & Norton, 1996, 2001). The authors later wrote that, shortly after the introduction of their model, they begon to realize that the business strategy should be the basis from which the KPIs should be derived (Kaplan, 2008a). The image in Figure 6 was meant to illustrate this.

It is no coincidence that the financial perspective is at the top in Figure 6. As the earlier quote indicates, the BSC model sees good financial results as the final goal of companies. The goals of the other perspectives are ultimately a means to an end.

Another assumption that Kaplan and Norton seem to make with their model is that it is an instrument for upper management to set and communicate goals to the rest of the organization (Kaplan, 2008a; Kaplan & Norton, 1996). In other words, it implies a top-down style of management where the people at the top largely decide what the company goals are and what should happen to reach them.

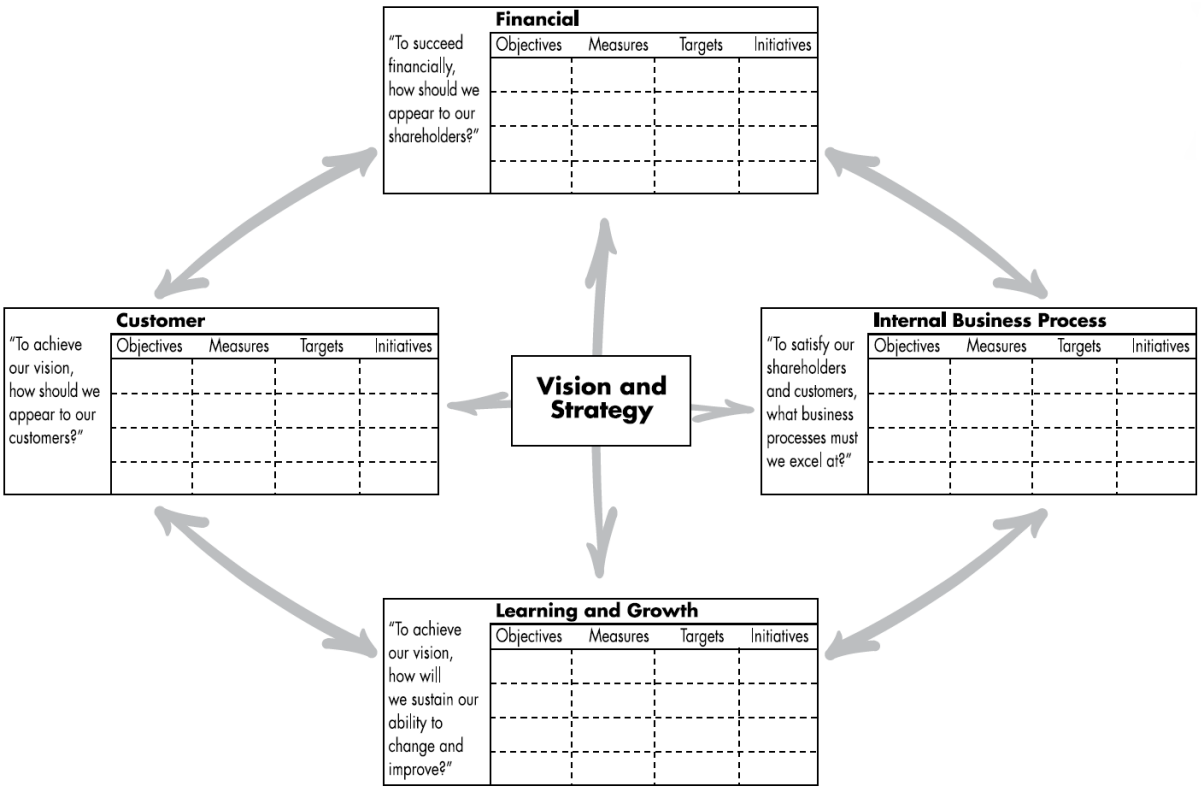


Figure 6 – An illustration of the Balanced Scorecard model (Kaplan & Norton, 1996)

Performance Prism

The Performance Prism is a model by Neely, Adams and Kennerly (Adams & Neely, 2000; Neely, Adams, & Crowe, 2001). The purpose of this model is comparable to that of the BSC, i.e. to aid managers of organizations with developing KPIs. It does so by presenting five questions:

1. Who are the key stakeholders and what do they need?
2. What strategies do we have to put in place to satisfy the wants and needs of these key stakeholders?
3. What critical processes do we require if we are to execute these strategies?
4. What capabilities do we need to operate and enhance these processes?
5. What contributions do we require from our stakeholders if we are to maintain and develop these capabilities?

These five questions correspond to the five faces of the prism in Figure 7. The faces on the side are called 'Strategies', 'Processes' and 'Capabilities' and are drawn separately from the prism for clarity. On the top is the face called 'Stakeholder Satisfaction', and on the bottom is 'Stakeholder Contribution'. Examples of common key stakeholders are given within the prism.

Here too it is no coincidence that 'Stakeholder Satisfaction' is placed at the top of the prism. According to the model, meeting the needs of the (key) stakeholder is the purpose of all companies and organizations. That is why the first question is about identification of these stakeholders. The authors of the Performance Prism acknowledge that different stakeholders can contradict each other, but suggest that the model could be useful to balance the degree to which every need is met (Neely et al., 2001).

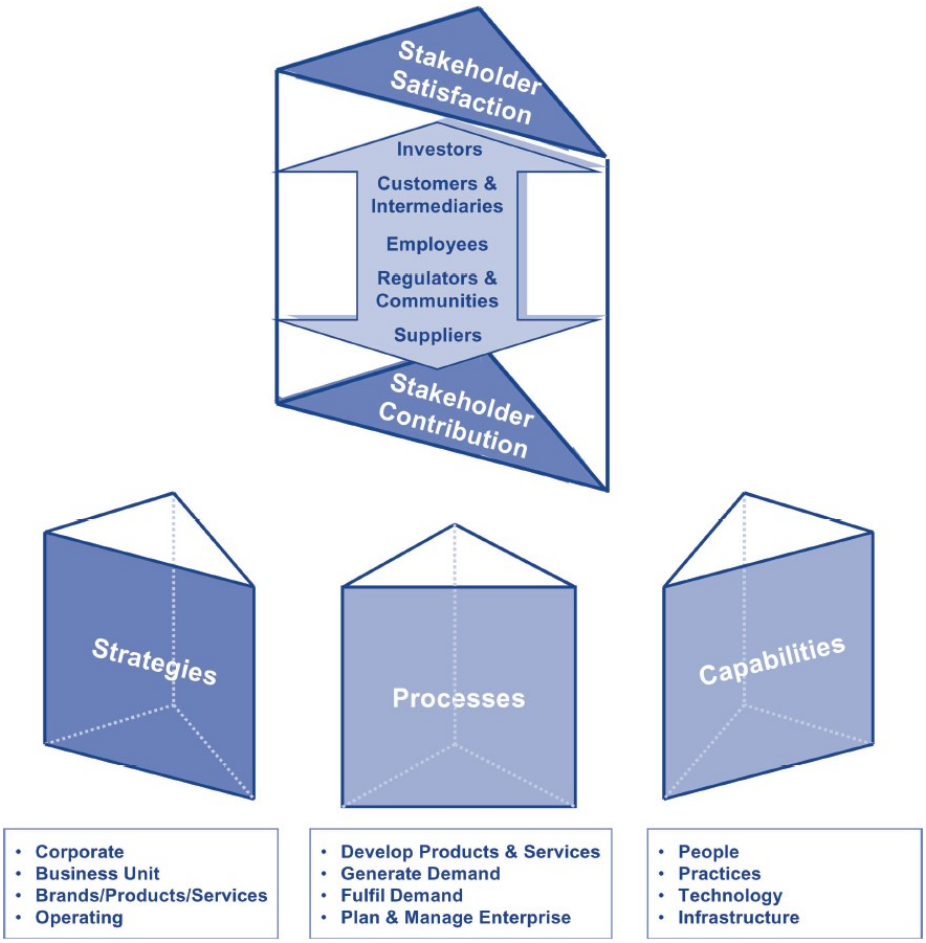


Figure 7 – An illustration of the Performance Prism model, copied from Neely et al. (2001)

EFQM Business Excellence Model

EFQM stands for the 'European Foundation for Quality Management' and was founded in 1989 to improve competitive position of companies in Europe. In 1991 they introduced the EFQM Business Excellence Model, which is shown in Figure 8.

The model identifies two kinds of performance: 'Enablers', which are the means, and 'Results', which are the ends. The EFQM model seems to be also based on stakeholder-theory because it divides the goals based on the stakeholders that are involved (EFQM, 2012a). It defines three groups: 'People', 'Customers' and 'Society'. What is interesting is that next to these stakeholders, 'Business Results' are also taken into account. When it comes to what is most important to a company, the EFQM seems to be a combination of the two previous models.

Like the previous models, the EFQM model distinguishes different categories means with which the desired end-results are supposed to be realized. In the case of this model, there are five: 'Leadership', 'People', 'Strategy', 'Partnerships & Resources' and 'Processes, Products & Services'. What they mean exactly and how they relate is unclear and subject to interpretation (Rusjan, 2005). The black lines seem to indicate some sort of relation between them but there are now arrows and the exact nature of the relation is not defined. Are they causal, and if yes, which way does the causal chain go?

It should be noted that the EFQM was originally design as a tool for self-assessment and benchmarking, not for the development of KPIs (Rusjan, 2005). The EFQM has written self-assessment methods that come in the form of large checklists that managers can use to score their organization

(Andersen et al., 2000). However, for the purpose of this project this is not problematic because the nine categories can still be used as a starting point for the search for adequate KPIs. The only downside is that the authors of this model have not developed a method to do so, while the authors of the other two models did.

Figure 8 – An illustration of the EFQM Business Excellence model, copied from the EFQM (2012b)

2.3.3. *Comparison and choice*

The BSC model and the Performance Prism are similar in a lot of ways. Firstly, both are designed to help managers develop KPIs for their organization. Secondly, both bring attention to different aspects of organizational performance. In that regard, Performance Prism could be called 'balanced' just as well. Thirdly, both models distinguish between capabilities, internal processes and customer value, although they use slightly different terms.

There are however some differences, the most important one probably being the role of stakeholders on the meaning of organizational performance. The Performance Prism states that stakeholders are most important and that the performance of an organization should ultimately be measured by the degree to which it meets stakeholder needs. In contrast, the BSC model puts the financial results of the company at the top and essentially regards everything else as a means to an end. As stated earlier, in this regard the EFQM model can be seen as a combination of the two.

Finally it should be noted that the BSC model has enjoyed much more attention than the other two models, both from academics as well as practitioners. This might have more to do with the quality of the authors' rhetoric than the model itself, as some suggest (Nørreklit, 2003). In any case, it has been tested and criticized and improved much more often than the others. Also, much more has been written on how to apply the model, i.e. method.

For these reasons the BSC model was chosen to more accurately describe what is meant by the performance of the Delivery department at Ymor. As we will see in the next chapter this will result in a number of organizational performance variables. KPIs are developed for these variables, but this subject will be covered in chapter four.

2.4. Conclusion

The five key variables of a MIS as described by Mason and Mitroff form the basis of the theoretical framework of this project and my MIS design method. These variables are:

1. The **psychological type** of the users of the system;
2. The **method of evidence generation** (or: the nature of the guarantor of evidence);
3. The **organizational context** within which the MIS operates and problems occur (see below)
4. The **class of problems** that the system is supposed to provide information about
5. The **mode of presentation** of evidence (i.e. information) by the system.

Each of these variables can be filled in differently. Figure 9 summarizes the possibilities that were discussed in this chapter.

Psychological Types

(a) 'Thinking'-'Sensing'
(b) 'Thinking'-'Intuition'
(c) 'Feeling'-'Sensing'
(d) 'Feeling'-'Intuition'

Classes of problems

(a) Structured
    (1) Decisions under certainty
    (2) Decisions under risk
    (3) Decisions under uncertainty
(b) Unstructured (i.e. *Wicked*)

Evidence generation methods

(a) Lockean (gguarantor: data)
(b) Leibnizian (guarantor: logic and models)
(c) Kantian (guarantor: multi-models)
(d) Hegelian (guarantor: conflict and synthesis)
(e) Singerian (guarantor: continuous cycles of doubt and agreement)

Organizational contexts

(a) System 1 – 'Implementation'
(b) System 2 – 'Coordination'
(c) System 3 – 'Control'
(d) System 4 – 'Anticipation
(e) System 5 – 'Identity'

Modes of presentation

(a) Personal
    (1) Drama/role-playing
    (2) Art
    (3) Personal or group discussions
(b) Impersonal
    (1) Company reports
    (2) Models

Figure 9 – Five key variables of MIS and their possible values; an alternative to Mason and Mitroff (1973)

This list is the same as the one by Mason and Mitroff expect for one thing, which is the possible values of the third variable, i.e. the organizational context. I decided to use the VSM to define the different

possible organizational contexts because it gives a richer description of organizational structures than the model that Mason and Mitroff used.

Thus the foundation of my design method is formed by a combination of the work of Mason and Mitroff and the VSM. The other model that was introduced in this chapter was the BSC model. Although I would not call it part of the foundation, it proved to be a useful addition to my design method because it helped define 'organizational performance' in the case of the Delivery department at Ymor.

In the next chapter I define the steps of my MIT design method, after which I use this method to decide what the most appropriate values of the five variables are for this case.

3. Analysis

The previous chapter described the five key variables of a MIS. It also contained a taxonomy of the different ways each variable could be given form. For example, one of the key variables of a MIS is the evidence generation method that it employs, and the options that we can choose are: Lockean, Leibnizian, Kantian, Hegelian or Singerian.

In this chapter, I use this theoretical framework for the construction of a MIS design method. The goal of this method is to determine the most suitable configuration of the five variables for the MIS under design. In other words, the method can be used to answer the first research question of this project (see paragraph 1.6):

Research question 1) *What are the functional requirements of the MIS for the Delivery department at Ymor?*

The first paragraph of this chapter contains an outline of the steps of the method. As we will see, the design steps can be divided into two phases of analysis. A description of the process and results of the first analysis phase is given in paragraph 3.2. The second analysis phase is discussed in paragraph 3.3. Finally, an answer to the research question is formulated in paragraph 3.4.

3.1. Method

In this paragraph I finally describe the steps that together compromise my MIS design method. The method is new; although Mason and Mitroff proposed the five key variables of every MIS, they did not specify the order in which these should be determined for a MIS under design. In fact, their point is that the five key variables are interdependent and therefore should be approached as a whole, i.e. in parallel and not sequentially. Unfortunately, all this is not very practical because we can only consider one thing at the time.

Considering one variable at the time has its difficulties because of the interdependency between the five variables. This means that deciding on the value of one variable restricts the possibilities of others. For example, different modes of presentation match different psychological types. This means that specifying the psychological type of the user(s) also largely determines what mode of presentation the MIS will have to employ.

However, Mason and Mitroff seem to imply that the strength of interdependency between different pairs of variables actually varies. I have attempted to illustrate the strongest interdependencies in Figure 10. The sequence of the steps of my design method are based on this.

The arrows in Figure 10 depict the direction of influence. For example, deciding on the organizational context of the upcoming MIS largely determines the kind of decision problems that will be relevant. The set of decision problems in turn determine what the most suitable evidence generation methods are. The other deciding factor for the most suitable evidence generation method is the psychological type of the users. This variable also directs the choice for the mode of presentation employed by the MIS.

Given these interdependencies, there seem to be two possible approaches to deciding on the best configuration of the five variables. The first is to start by deciding on the psychological type of the users, followed by the selection of the most suitable evidence generation method and the appropriate mode of presentation of the MIS. When this is done, the type of organizational context can be

determined. The last step of this approach is to identify the class of problems that the MIS will provide information about. The hope is that the class of problems will be compatible with the evidence generation method that was decided upon in a previous step. If this is not the case, the designer has no choice but to review the configuration and decide which variable should be configured differently to make the whole set more coherent.

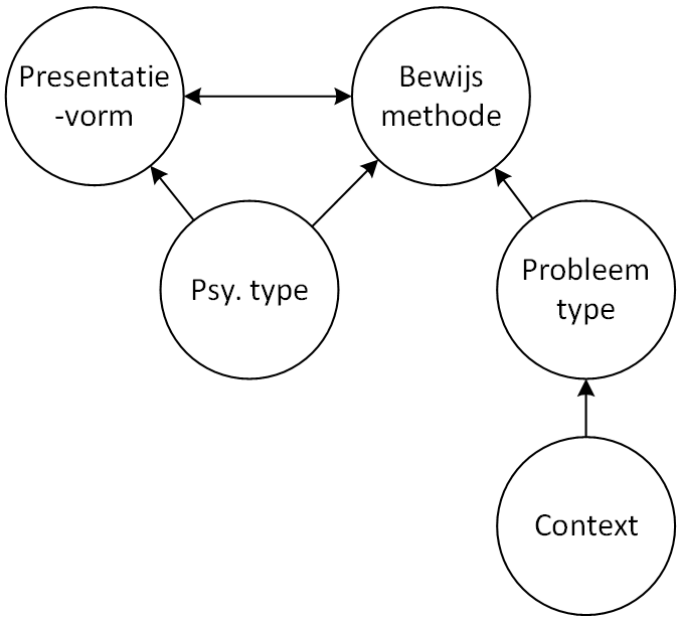


Figure 10 – A visual representation of the strongest interdependencies between the five key variables.

The second approach consists of a somewhat reversed order and starts by deciding on the organizational context of the MIS under design, followed by the class of problems. Next, based on these preliminary results, the most suitable evidence generation method for problems of this nature is chosen. At this point the number of possible modes of presentation that match with the rest of the variables is probably reduced significantly. Finally, the designer determines psychological type of the user(s) of the MIS under design. Here too it is hoped that the result is an internally coherent configuration of the five variables. If not, a second analysis is necessary to resolve the conflicts.

For this project I have decided to follow the first approach, which means I started by determining the psychological type of the users, the evidence generation method and the associated mode of presentation. The reason for this decision is that the DMs had already expressed their preferred configuration of the latter two. I had also observed the culture and people at Ymor enough to derive the likely psychological type of the DMs.

However, as I have said, there is a chance that the first attempt results in an incoherent configuration of the five variables, no matter which of the approaches is taken. What makes the configuration incoherent is that at least two of the variables are given values that are incompatible with each other. For example, suppose an evidence generation method is chosen that works well for problems of a structured nature. If the actual relevant problems then turn out to be of a very unstructured nature (i.e. 'wicked' problems), then we could say the class of problems is incompatible with the chosen evidence generation method. In those cases, a second analysis phase is necessary to determine how to resolve this conflict.

Unfortunately, this turned out to be the case with this project. After the first analysis I found that the evidence generation method that the DMs preferred was not compatible with the unstructured or 'wicked' class of problems that they face. I therefore performed a second analysis phase with the purpose of transforming the nature of these problems from unstructured to more structured ones. It is for this purpose that the BSC model was used.

Figure 11 summarizes the design process that I have just described.

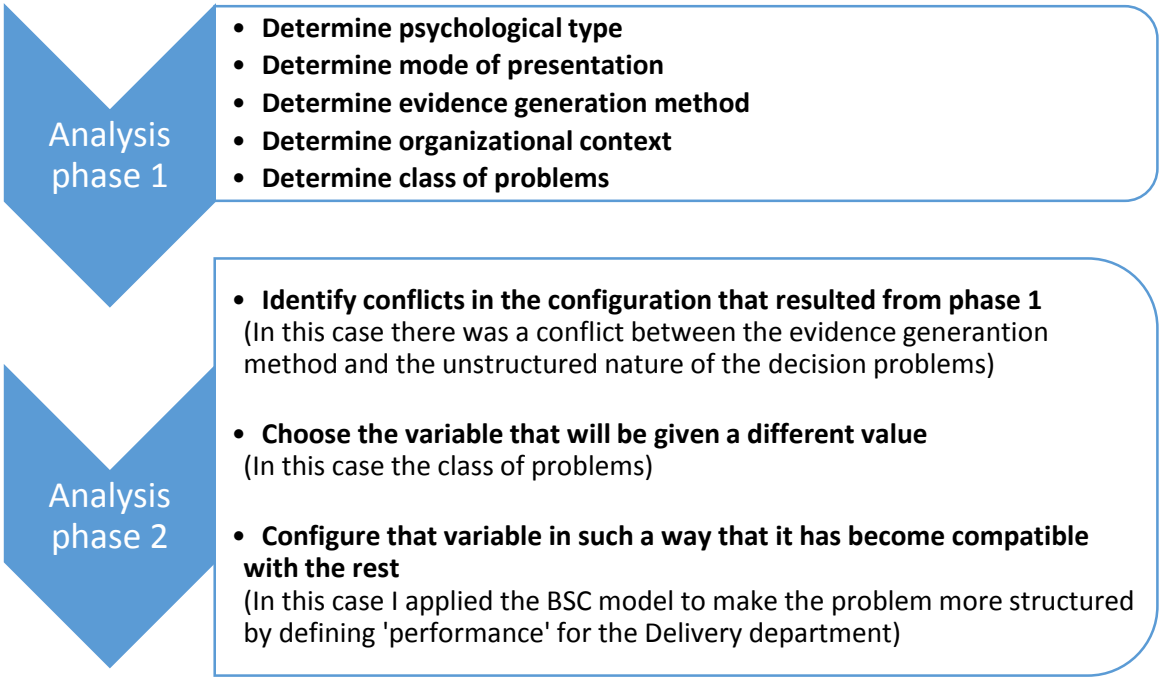


Figure 11 – A summary of the steps of the MIS design method applied during this project

3.2. Analysis phase 1

In this paragraph, I describe my first attempt to configure the five key variables of the desired MIS. The first step will be to determine the psychological type of the users. This is followed by setting the mode of presentation of information, after which I decide upon a compatible evidence generation method. The fourth step involves determining the organizational context of the upcoming MIS. I have described this process in more detail than the first three steps in order to clarify my use of the VSM. The purpose of the fifth and last step of this analysis phase is to find out what class of problems the MIS will be dealing with. All this is followed by a final section where I discuss the results of the analysis and explain the need for a second phase.

3.2.1. Step 1 –the psychological type of the users

As I wrote in chapter 2, Mason and Mitroff name four psychological types based on two characteristics. The first characteristic involves the way in which a person perceives the world. This can be either via 'Sensing' or 'Intuition'. The people at the Delivery department at Ymor, which includes the DMs, seem to correspond with the 'Sensing' type. This type is associated with empiricism because it prefers objective data and observations, i.e. 'facts'. In contrast, people of the 'Intuition' type have a more inductive attitude and are more interested in adding meaning. It is perhaps somewhat unsurprising that the people at Delivery lean towards the former type. After all, they specialize in making

performance issues less subjective by setting up systems that make measures objective data using specialized software. One could say the people at Delivery are being paid to think and act as a Sensing type.

The second characteristic is the way a person evaluates his or her observations. Again, two types are distinguished: the 'Thinking' type and the 'Feeling' type. Mason and Mitroff's description of the Thinking type seems to correspond most with the mindset of the people at the Delivery department of Ymor, including the DMs. To put it simply, people with the 'Thinking' characteristics generally rely more on logic and reason to make their conclusions, while people of the 'Feeling' type focus more on emotional and social dimensions. The people at the Delivery department are all trained IT engineers, which means they are trained to think in terms of logic and formal systems.

We have arrived at the Sensing-Thinking combination, although perhaps not by very scientific means. This classification is based on my own observations during my months at the company. I did not make use of a scientifically tested method, mostly because of my unfamiliarity with this field and the time constraints involved with this project. However, during the project my classification of Delivery people as 'Thinking'-'Sensing' seemed to be confirmed by a new observation. One of the senior employees at the Delivery department revealed that he had once been characterized as a Sensing-Thinking type by an organization that specializes in these kind of psychological analyses.

3.2.2. *Step 2 – the mode of presentation*

What is perhaps the most important piece of proof supporting the classification of the DMs as Sensing-Thinking types, is their preferences regarding the mode of presentation of information by the future MIS.

Mason and Mitroff state that nearly every MIS-related project and research implicitly assumes that the MIS consists of a combination of IT software and hardware, and that its mode of presentation is computer-generated reports. In other words, these are the considered to be the 'normal' kind of MSI. However, the authors point at the possibility that some situations call for MIS that employ different modes of presentation of information, such as group discussions, role-playing, stories, or art.

But the DMs at Ymor are very clear about their preference: they wish for a 'normal' MIS. They expect a system that is composed by IT software and hardware, and presents information in the form of reports based on compiled data. As I indicated, this preference of the DMs is an important pointer to the Sensing-Thinking type.

3.2.3. *Step 3 – the evidence generation method*

The desire for reports combined with the Sensing-Thinking psychological type points at a Lockean method of evidence generation. The wish for KPIs is another thing that corresponds to a Lockean MIS. After all, the Lockean method of evidence generation is characterized by the collection of data and forming consensus on the truthfulness of statements based on said data (see chapter 2). However, a Lockean MIS requires that the nature of the problem is well understood among those involved. There is no room for different interpretations. Whether or not this is actually the case with the decision problems that the DMs would like to solve is discussed in section 3.2.5. However, before I go into the class of problems it is first required to define the organizational context of the MIS under design.

3.2.4. *Step 4 – the organizational context*

The goal of this section is to decide on the organizational context of the MIS under design with the use of the VSM. According to the model the following questions need to be answered:

1. What viable systems can be identified at Ymor?
2. Which of these viable systems will the upcoming MIS be a part of?
3. Which of the five subsystems of this viable system will the MIS be supporting?

These questions are addresses below in the same order as they appear in the list. As question 1 indicates, the first step is to uncover the structure of Ymor as seen through the lens of the VSM. Because Ymor is an organization, it must be also a viable system. But as I explained in chapter 2, viable systems themselves are composed of other 'smaller' viable systems. Thus an important part of uncovering the structure of Ymor involves identifying its sub-organizations, and perhaps even its sub-sub-organizations, depending the number of relevant recursion levels. I have done this by exploring the structure of Ymor as a whole first, and then worked my way 'down' (so to speak) to find the particular viable system that the MIS will be a part of. To do this, I needed an adequate understanding of what the five subsystems of the VSM mean. As promised in chapter 2, I will now discuss these in more depth.

The five subsystems of the VSM

System 1 – 'Implementation'
System 1 is the largest part of the Operational System of a viable system. If the viable system in question is an organization, System 1 is composed of the parts that perform the core activities. These parts delivery goods and services to the external environment. As I have stated before, an important property of the VSM is that each of these parts is itself a viable system, with its own Meta-System and Operational System. See Figure 4 in chapter 2 for an illustration. This shows the recursive nature of the VSM. What this means is that according to the VSM, each sub-organization has a high degree of autonomy. After all, each sub-organization has its own Meta-System that is responsible for control and planning (see the descriptions of Systems 3 and 4 below). Only in situations where the sub-organization does not have the means to respond to an event does the Meta-System of the whole intervene.

Because the VSM is recursive, one could essentially keep identifying new levels of organization (e.g. sub-sub-sub levels) indefinitely. In practice however, most researchers that apply the VSM stop at the level where System 1 consists of individual people (Espejo & Gill, 1997). If we would go any further, we would find ourselves in the domains of psychology and biology.

The last important feature of System 1 is that it is the only part in the whole system that directly interacts with the external environment, with the exception of System 4 (see below).

Other names that are sometimes used for System 1 are 'Operations', 'Process', 'Producer' and 'Service Delivery'.

System 2 – 'Coordination'
System 2 is shown in Figure 4 as being neither part of Meta-System nor of the Operational System, while in other versions it is part of both, e.g. see Figure 12. The purpose of System 2 is to coordinate the different parts of System 1. It prevents these parts from getting in each other's way. Stafford Beer calls this 'oscillation' (Beer, 1984), and so the purpose of System 2 is anti-oscillation. Examples of

implementations of System 2 in organizations are planning systems, timetables and communication protocols. In smaller organizations System 2 sometimes forms spontaneously because the need for coordination is evident to all those involved. However, in general, as the organization grows, so does the need for formal processes and protocols.

Alternative labels for System 2 are 'Anti-Oscillation' (Beer, 1984), Regulation, Scheduler, Conflict Resolution, Agreement and Standards.

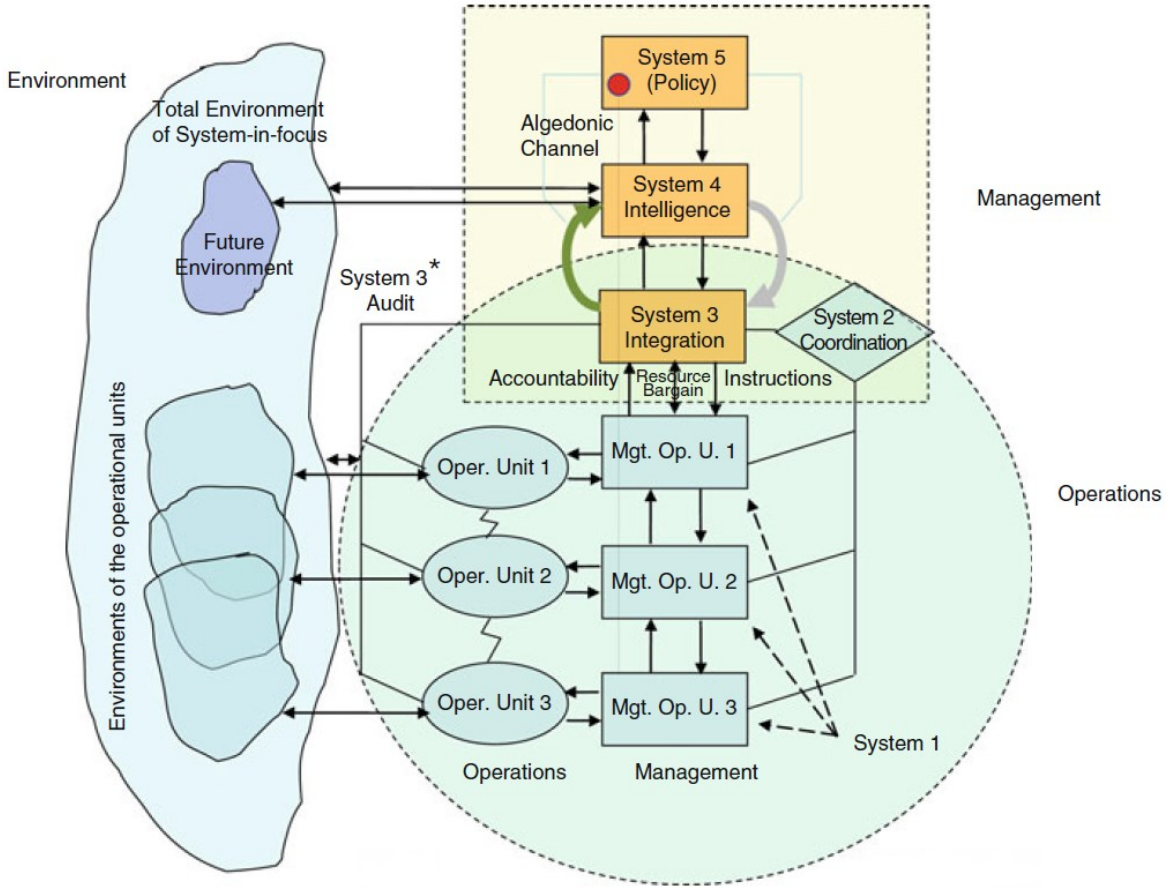


Figure 12 – A slightly different version of the VSM, taken from Ríos (2012)

System 3 – 'Integration'

System 3 focuses on the goals of the whole viable system and has the authority to make decisions that concern the parts of System 1 in service of those goals. Such decisions can for example concern resource allocation between the parts of System 1. There could be situations where all the parts of System 1 desire more resources to work more effectively towards their local goals, but it is up to System 3 to decide how these resources should be allocated to ensure optimal functioning of the whole.

To be able to function, System 3 needs to have adequate information about the performance of all the operational parts. It has two ways to attain this. The first is via the information channels that go from the parts of System 1 to System 3. These channels represent the accounting function of system 3. It should be clear that System 3 depends on the parts of System 1 to provide this information.

The second way is via audits. The audit function of System 3 is sometimes depicted as a separate subsystem with the label 'System 3*'. The audit function is the way in which System 3 investigates the

state of certain aspects in System 1. It completes the accounting function because it can check for information-gaps that are left by System 1's parts.

As both Figure 4 and Figure 12 show, System 3 is also connected to System 2. This is necessary because System 2 only enables coordination between the operational parts. In some conflict situations however, judgement is needed from the perspective of the whole, which is exactly what System 3 is. In that case, information is sent from System 2 to System 3 so that the latter can deal with the situation. The link also illustrates that System 3 is responsible for the design of System 2.

Although the authority associated with System 3 is often found at certain individuals in organizations, it is not the case that System 3 is simply a part of an organizational chart (or any of the five Systems for that matter). It is not simply the case that 'Person X is System 1, Person Y system 3", et cetera. Instead, the Systems are essentially control activities and these are often embodied by management activities such as meetings and discussions. Of course people play a central role in these activities, but that does not mean that these people should be equated to these Systems. People can play a part in multiple subsystems. For example, one person could perform an activity that falls under a part of System 1, but this person could also have certain management responsibilities that are associated with System 3. This is certainly the case at Ymor. The DMs are supposed to be focused on their department as a whole (i.e. Meta System of Delivery), but at the same time play the role of SDM for two teams, which is part of System one (at least from the perspective of the whole department).

Examples of other labels given to System 3 are 'Control', 'Optimization', 'Synergizer', and 'Here-and – Now management' (Beer, 1984).

System 4 – 'Intelligence'

System 4 is part of the Meta-System, just like System 3, although it plays a very different role. System 4 observes the external environment for signs of change that could affect the viable system, and makes plans accordingly. This relation with the environment is illustrated in Figure 4 and Figure 12 with black arrows going from System 4 to the environment. Thus, while System 3 is associated with the day-to-day aspects of management, System 4 has a management function that is more strategic of nature (Pérez Ríos, 2012, p. 41).

For System 4 to be able to estimate what internal changes are necessary to anticipate changes in the environment, it requires a model of the current state of the Operational System. But this is the responsibility of System 3, so this means that System 4 and 3 are in a continuous dialogue where the former expresses the need for change while the latter underlines the practical restrictions posed by the current state of the system. This is also referred to as the System 3-4 homeostat (Leonard, 2009).

System 4 is often recognizable in organizations as marketing and sales activities. Other labels for System 4 could be: 'Planning', 'Development', 'Learning and Adaption', 'Prediction, 'Exploration', and 'Outside-and-Then management'.

System 5 – 'Policy'

System 5 is the last part of the Meta-System and determines the identify of the whole system. It provides the existential ontology of the organization and defines its purpose. Or, to put it in terms of businesses: it sets the organizational vision, mission and values of the company. System 5 is the highest authority of the whole system. If the aforementioned System 3-4 Homeostat is out of balance, then it is the responsibility of System 5 to restore it by setting priorities.

System 5 often takes the form of meetings by the owners and/or members the board of directors. Next to ‘Policy’, other labels that are associated with System 5 are ‘Identity’, ‘Preserver of Values’, ‘Explicated Purpose’ and ‘Ultimate Authority’.

Viable systems at Ymor

Now that it is clear what the five subsystems are that together compose a viable system, we can start identifying different (levels of) viable systems at Ymor. I have done this by starting with a look at Ymor as a whole and seeing how the Meta-System and the Operational System is given form on that level. I then focused on identifying the different parts of the Operational System and found multiple departments. One of these departments was the Delivery department, so I then focused my attention on that. By talking to many of the people at Delivery I began to see how the Meta-System and Operational System are organized on this level of analysis. As with Ymor as a whole, the Operational System of the Delivery (sub-)organization was composed of even smaller viable systems, which in this case were the eight Delivery teams. These teams too had a Meta-System and an Organizational System consisting of individual team members.

I have summarized these findings in Figure 13 on the next page. It shows the three levels of recursion that I mentioned above. The naming convention is to assign the number 0 to the first level and to increase the number as we ‘dive deeper’ (i.e. recursion levels 0, 1, 2, et cetera). So the higher the number, the further one has zoomed in, so to speak. In this case zooming in means going from company to department, from department to teams, and from teams to individuals.

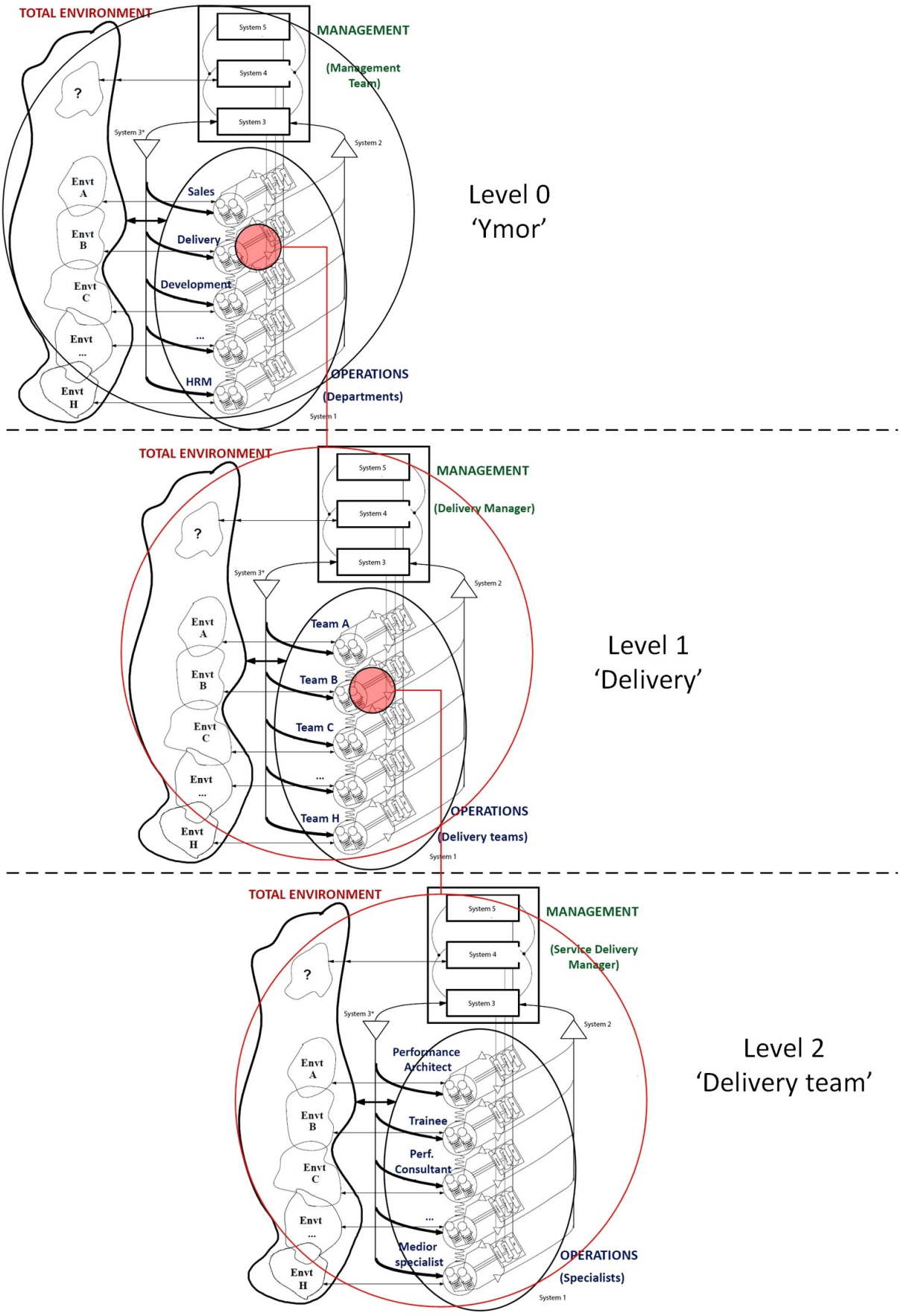


Figure 13 – Viable systems at Ymor identified on three different recursion levels

Ymor as a whole

The first viable system that can be identified is that of Ymor as a whole. To fill in the parts of System 1 of Ymor as a whole I turned to the organizational chart. The departments that it listed are Sales, TAM (Technical Account Management), Delivery, Software Development (or Development for short), IT Support, HRM and Finance. Does this mean that all of these are the viable systems that together form the Operational System of Ymor as a whole? The answer to that question depends on whether each department has a Meta-System and an Operation System, and (equivalently) if System 1 to 5 can be recognized in them. Or, to use the words by Leonard*: What makes [departments*] *viable systems is that any of them could be sold off as independent businesses active on their own"* (2009, p. 226). When I considered the departments in this light, I saw that every department listed in the organizational chart is a viable system, except 'Finance'. The Finance department actually consists of a part-time accountant that reports to the Management Team and advises other people at Ymor on financial subjects. It has no strategic goals and no way to observe or adapt itself to a changing environment. Simply put, it has no Meta-System to speak of. It seems to me that the financial department actually performs an information providing role and therefore could be considered to be part of System 2 or System 3* of Ymor. Furthermore, the departments 'Software Development' and 'IT Support' are merging and thus becoming one single viable system en share the same Management.

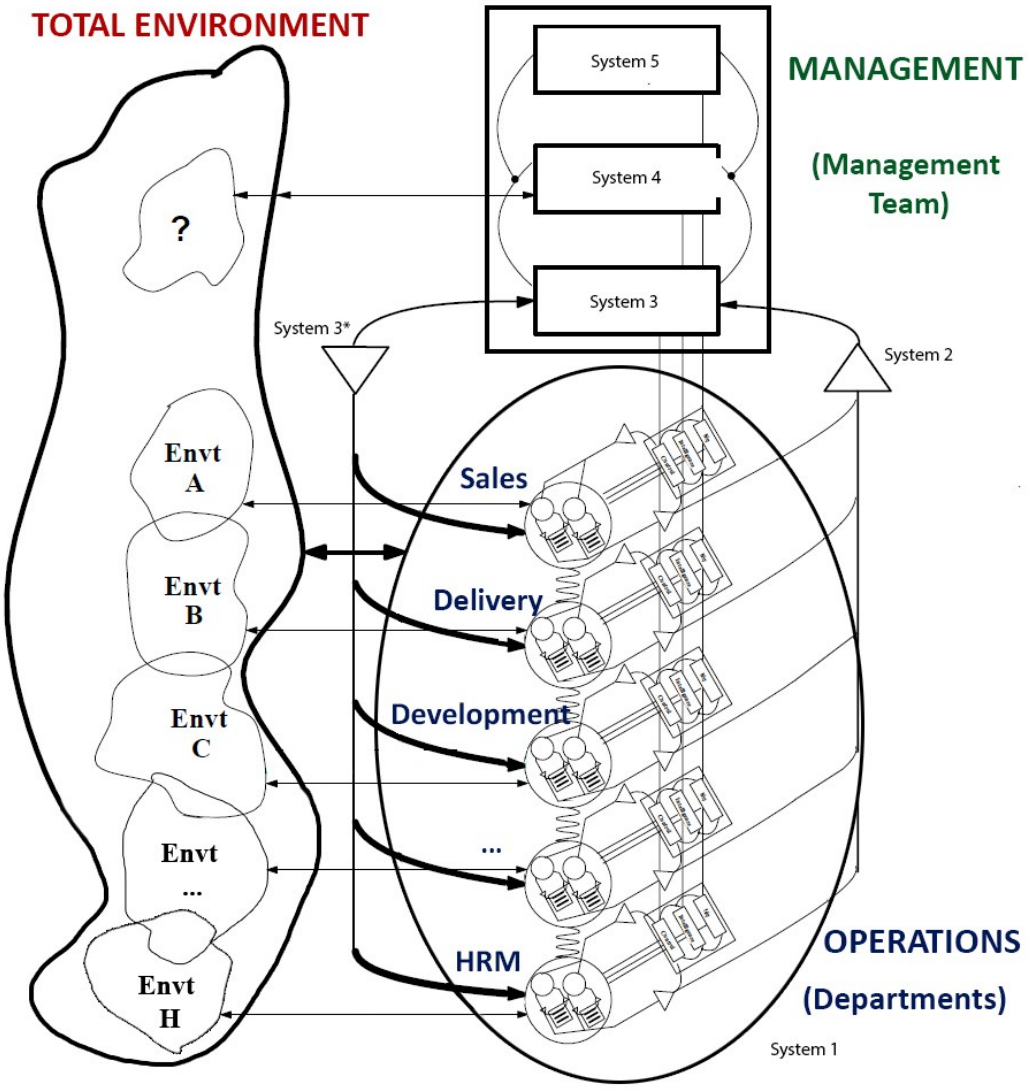


Figure 14 – Recursion level 0: Ymor as a whole

The three members of the Management Team (MT) embody most of the Meta-System on this level. They attempt to ensure that the combined effort of the departments results in the attainment of the company's goals (System 3). They also keep a lookout for changes in the market that might be important for the company's survival (System 4). Finally, the MT decides upon the company's core values and makes sure everyone makes decisions with these values kept in mind (System 5).

Delivery department

Looking at the different departments that together form System 1 of Ymor as a whole, it became clear that I needed to focus on the Delivery department. This department is the viable system that the future MIS will be a part of. Thus I chose to 'zoom in' to the viable system that is the Delivery department and analyzed its structure in the same way that I did for Ymor as a whole. The purpose of the Delivery system is to provide the services that the Sales system has sold to the clients, often with the help of the tools that developed by the Software Development system. The DMs are responsible for the Delivery system, which means they largely form the Meta-System of Delivery.

The external environment that the Delivery system interacts with is divided into different groups of client organizations. Every group is assigned to a different Delivery team, as can be seen in Figure 15 on the next page. As the figure shows, these teams are the different parts of System 1 of the Delivery system, which means that every Delivery team is itself a viable system. The teams pose the third and last level of recursion of the VSM that I will discuss (see below).

Before I do that however, I will quickly describe how I recognized the four other subsystems of the VSM at the Delivery department. As I established earlier, System 2 is about coordination between the parts of System 1. Most of the applications at Ymor can be seen as part of System 2 of Ymor as a whole. An example being the email system, or the system of agenda's. However, I also recognized a System 2 that was specific for Delivery. The most important example is the weekly Delivery meeting. This is where the team leaders, i.e. the SDMs, come together and discuss capacity issues and other forms of 'oscillation' that they experience during their activities. The goal of these meetings is to share the load, for example by temporarily sending a person from team A to assist team B. This is an example of System 2 at work because the parts of System 1 are coordination themselves.

However, in some cases that same Delivery meeting can also be recognized as a form of System 3. If there is a problem that the SDMs together are not able to solve, then the DMs are expected to make the final decision and give instructions to the rest. The DMs are responsible for Delivery as a whole, so they have to deal with things like ensuring that there are enough people with certain expertise, or checking if there are enough software licenses in stock, et cetera. They also need to monitor the performance of the Delivery department as a whole. *This means that, from a VSM perspective, the core problem described in chapter one is located in System 3 of the Delivery system.* In other words, System 3 of Delivery is not functioning properly because it does not have sufficient information on the performance of the Operational System (which in this case consists of eight Delivery teams). This means that System 3 seems to be the organizational context of the MIS under design.

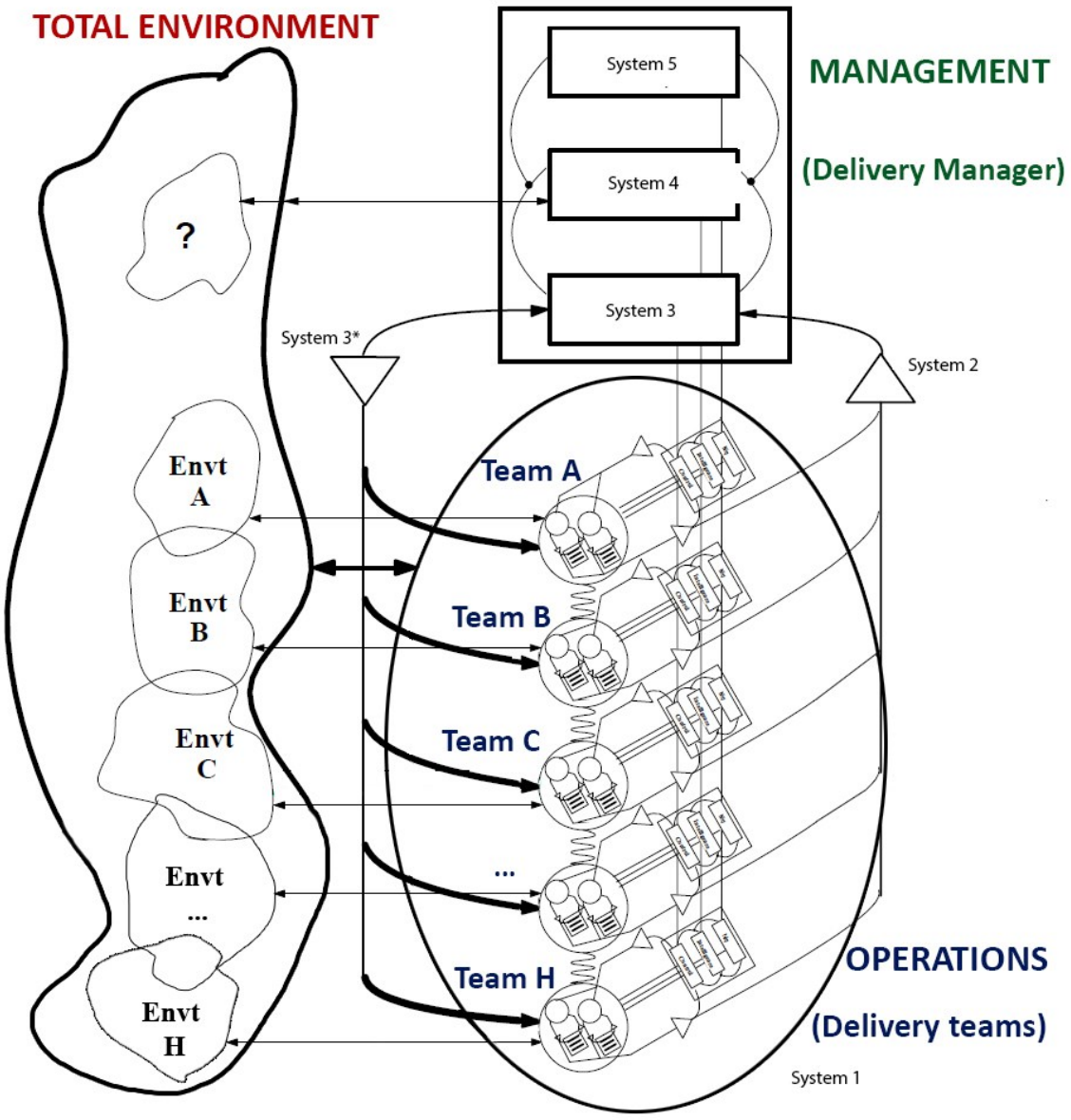


Figure 15 – Recursion level 1: the Delivery department as a viable system

But what about System 4 and System 5? As we know, System 4 is involved with the anticipation of changes in the world that could affect the organization, in this case the Delivery department. I have recognized two forms of System 4 in the activities of the DMs. Firstly, they are continuously on the lookout for new software instruments that could help them provide their services in a more effective and efficient way. Secondly, they are always communicating with Sales to make sure they know about future fluctuations in demand for their department's services as much as possible. Although information plays an important role here as well, the MIS under design will not be aiding these activities. There is no strong need for more information regarding these activities at this point.

Finally, System 5 is about defining and preserving the identity of the department. I did not clearly recognize this subsystem at Delivery. In any case, it does not seem to be a source of information problems and hence will not be part of the organizational context of the MIS under design.

Delivery teams

This is the last level of VSM recursion that I will discuss. While the MIS under design will serve System 3 on the level of the Delivery department, the team level is still relevant because it is contained within Delivery. This means that the MIS will have to collect information about the teams and their activities.

As Figure 16 shows, the parts of the Operational System of each team are individual people. Would we 'zoom in' any further, we would find ourselves dealing with the human body, which has little to do anymore with the core problem of this project. The number of people per team varies but usually lies between four and seven people. Most of the members have a different function profile. Some are trainees and just started working for Ymor, while some are 'performance architects' with years of relevant working experience. I will come back to the subject of function profiles in more detail later.

The Meta-System of each team is embodied by the (activities of) a single SDM, i.e. the Service Delivery Manager. The SDMs ensure that the members of the team work together in such a way that all the deadlines and team goals are met. They also anticipate on changes in the situation at the clients that could affect their team's workload. All except two of the SDMs manage a single team. As I have stated before, the DMs also play a SDM role and they are the ones that manage two teams instead of one.

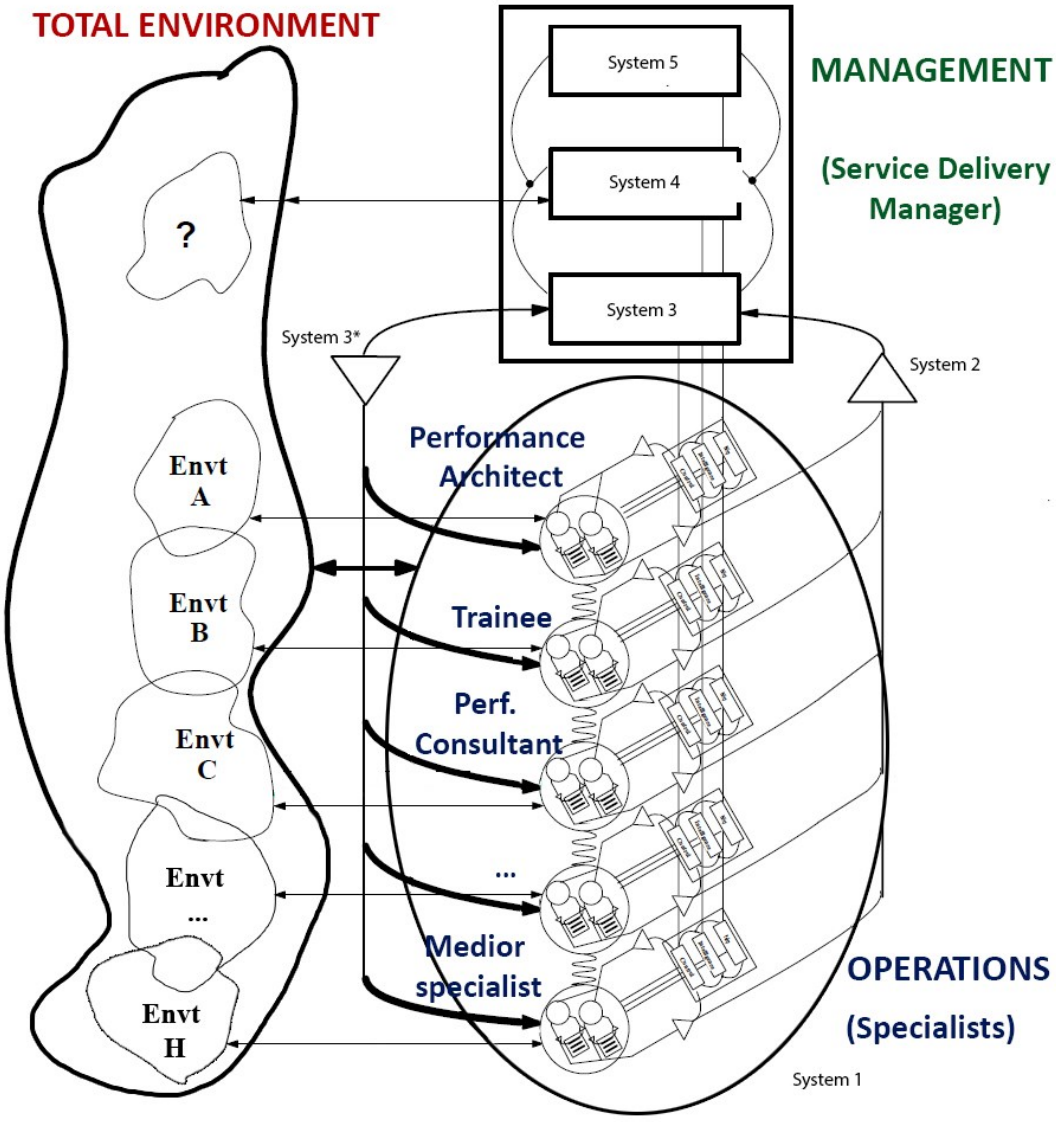


Figure 16 – Recursion level 2: a Delivery team as a viable system

Discussion and conclusion

The goal of this section was to decide on the organizational context of the MIS under design with the use of the VSM. For this I sought an answer to the following questions:

1. What viable systems can be identified at Ymor?
2. Which of these viable systems will the upcoming MIS be a part of?
3. Which of the five subsystems of this viable system will the MIS be supporting?

As I have showed, one can recognize a lot of different viable systems when looking at Ymor through the lens of the VSM. Firstly, there is Ymor as a whole, which is the viable system at recursion level 0. Within it are contained smaller viable systems, which are the numerous departments. Within each of these departments, other viable systems can be identified. I have chosen to focus on the Delivery department and recognized the eight Delivery teams as the smaller viable systems that together make up the Operational part of Delivery.

It also became clear that the future MIS would be situated within the Delivery department. I then took a closer look at this viable system and described Systems 1 to 5 on that level. I found that the core problem of this project was related to a dysfunctional System 3 of Delivery. This meant that the organizational context of the MIS is System 3 of the Delivery system.

This concludes the search for an appropriate value of the fourth key design variable of the desired MIS. Deciding on the organizational context was necessary to determine the fifth and last key variable: the class of decision problems that the MIS is supposed to provide information about. This variable is examined in the section below.

3.2.5. *Step 5 – the class of problems*

The class of problems is the last key variable of the MIS that needs to be determined to complete this analysis phase. The core problem chosen in chapter one functions as a starting point. As described in chapter one, the core problem is that the DMs have insufficient information about the performance of the Delivery department. According to the problem cluster, this results in an inability of the DMs to make effective decisions for the good of the department because they lack the information with which improvement can be guaranteed.

As I stated in chapter 2, there are two classes of decisions problems: structured problems and unstructured (or 'wicked') problems. Whether a decision problem is structured or unstructured depends on:

1. The extent to which the actions {A} that the decision makers can take are known;
2. The extent to which the possible outcomes {O} of these actions are known;
3. The extent to which the utility {U} of every outcome is known;
4. And the extent to which the set of possible states of the world {S} are known;

A decision problem is considered to be structured when the sets of {A}, {O}, {U} and {S} are known. If one or more of these sets are unknown or at least not with a high degree of confidence, then the problem is considered to be unstructured, or 'wicked'.

Moving back to the situation at hand, it seems that the decision problems that the DMs face are not structured but rather somewhat unstructured. The reason for this is that 'the performance of the Delivery department' has not yet been defined properly. In other words, there seems to be no strong

agreement on which states of the world (i.e. {S} actually matter because the meaning of 'performance' is open to interpretation.

Because of the lack of agreement on what constitutes as Delivery's 'performance', I conclude that the decision problems that the MIS should help with are moderately unstructured. All five key variables have now been determined for the MIS under design, and the first analysis is completed. I review the findings in the next section.

3.2.6. *Summary of results*

The results of this analysis phase are summarized in Figure 17. Firstly, I found that psychological type of the users was likely to be the 'Thinking'-'Sensing' type. This type prefers objective data and reports to other forms of information and presentation. Because this type leans to empiricism and the DMs in fact explicitly requested a MIS that produces reports, the Lockean method of evidence generation was chosen for this project. Furthermore, I found that the organizational context of the MIS under design is System 3 of the Delivery department, which means that the MIS will support the control and integration of the performance of the eight Delivery teams.

Until now I have mentioned the value of four out of a total of five key variables, and no internal conflicts are apparent. Unfortunately, such conflict is introduced when we consider the fifth key variable, i.e. the class of problems. As it turns out, the decision problems that the DMs face are of an unstructured or wicked type. According to Mason and Mitroff, the Lockean method of evidence generation is suitable for situations where there is strong consensus on the nature of the problems. Only then is it possible for those involved to use data to agree on the right decision. If the nature of the problem is unclear, which seems to be the case here, then no agreement is possible because it is unclear what kind of data is even relevant. We are thus forced to conclude that the decision problems in their current form are incompatible with a Lockean evidence generation method.

Key variables of a MIS	**Chosen value of the variable for this case**
Psychological type of the users:	'Thinking'-'Sensing'
Mode of presentation:	Computer-generated reports
Evidence generation method:	Lockean
Organizational context:	System 3 of the Delivery department
Class of (decision) problems	Onstructured, i.e. wicked

Figure 17 – A table that summarizes the results of the first analysis phase

There were now two options. The first option was to change the unstructured decision problem to a structured one. Option two was to change the evidence generation method from Lockean to one that is more suitable for problems of a wicked nature. Unfortunately, if the latter would be chosen, it would result in a MIS design that generates a kind of evidence that does not match the psychological type of its users. However, option one would mean that the decision problems might become over-simplified, making the 'evidence' provided by the MIS actually misleading because the 'wicked' part of the problem has been left out of consideration. Churchman has written extensively on the danger of this kind of deception (e.g. 1967, 1968, 1979). In short, both options have its downsides. Nevertheless, I decided option one is preferable, because the other option provides no solution to the core problem whatsoever, while option one does (albeit a limited one).

Option one means a second analysis phase is required. In the following section, I describe my attempt to transform the decision problems from unstructured to a more structured type.

3.3. Analysis phase 2

I concluded in the previous paragraph that the current form of the decision problems of the DMs is too unstructured (i.e. too wicked) to be suitable for a Lockean MIS. The reason for this wickedness is the ambiguity that surrounds the meaning of 'the performance of the Delivery department'.

To make the decision problems more structured, I decided to define Delivery's performance in terms of performance variables. I have applied the BSC model to do this. As we have seen in chapter 2, the BSC model prescribes four different perspectives on performance, which means that the following four questions have to be answered if one wants to define the performance variables for a (sub-)organization like the Delivery department:

Question 1. What are the financial performance variables of the Delivery department?
Question 2. What are the customer-related performance variables of the Delivery department?
Question 3. What are the performance variables related to the internal processes of the Delivery department?
Question 4. What are the resources-related performance variables of the Delivery department?

Answering these questions ensures that every perspective of the BSC model is accounted for.

The use of the BSC model resulted in the identification of seven performance variables of the Delivery department. However, before I discuss these results, I first explain what method I used to apply the BSC model. That is the subject of the first section.

The second section of this paragraph describes the analysis process and contains a discussion about the resulting seven performance variables. Three of these seven performance variables turn out to contain too much ambiguity, i.e. they still possess a 'wicked' quality to a certain degree. This means that defining these seven performance variables solved a lot of the 'unstructured-ness' of the original decision problem but not all of it. The three ambiguous performance variables of the Delivery turned out to be: 1) the department's 'innovativeness', 2) the degree of implementation of the Ymor core values at Delivery, and 3) the degree to which the required competencies are present at Delivery.

For reasons that I describe later, I chose to leave the first two out of the scope of the MIS design. For the third variable, i.e. the one concerning the required competencies, I attempted to resolve the ambiguity. This attempt and its results are described in the third section of this paragraph.

I have written my discussion of the end-results of both analysis phases in the final paragraph of this chapter.

3.3.1. *Methods for the BSC model*

I have looked at three different methods for the application of the BSC model and have chosen one of them. The first method I considered was the one by the authors of the BSC model themselves (Kaplan & Norton, 1998). The remaining two are created by others, namely Hudson et al. (2001) and Biazzo and Garengo (2012). These two groups of researchers both made an alternative to Kaplan and Norton's method because of supposed flaws in the latter. Their main concern was that the method of Kaplan and Norton was not suitable for use at small and/or medium enterprises (i.e. SMEs). Both groups believe their method is an improvement in that regard.

The method that I have chosen is the one by Biazzo and Garengo, most importantly because it is based on a bottom-up approach while the other two methods are top-down. What this means and why this is important is explained later. I will now first introduce each method separately. This is followed by a comparison of the methods, after which I justify my choice.

Option 1: Kaplan and Norton

Kaplan (2008) writes that there were many organizations that wished to implement the BSC model to develop KPIs next to financial measures. However, most were uncertain about how to derive KPIs for each category. Some decided to employ KPIs that were already using and simply divided them over the four categories. Others looked at the organizations they admired and took the same KPIs that were used there. But Kaplan and Norton quickly realized that the KPIs were most useful if they were derived from the company's own strategy. Of course this implies that the organization in question has explicitly defined its strategy, meaning there are written strategic goals that the company is striving to achieve.

The dictum that KPIs should be derived from strategy sometimes had an unforeseen effect on the BSC implementation project. What started as an attempt to formulate KPIs quickly transformed into an effort to improve and elaborate of business plans and strategy (Kaplan, 2008a). In other words, during these projects many organizations realized that their strategy was not clear or explicit enough, or incomplete, and hence that needed to be corrected first.

Based on these insights, Kaplan and Norton developed a method for applying the BSC model. This method is illustrated Figure 18. The figure is meant to be read from top to bottom. It clearly shows the belief that the development of KPIs should start with strategy. The four questions to the left each resemble a different phase of the method.

As the first question shows, the first phase is about defining the purpose of the organization in a broad way. (Here we might be reminded of System 5 of the VSM, which is responsible for the organizational vision, mission and values.) This is the starting point of the process and sets the direction.

The second phase revolves around the question "if [our] vision succeeds, how will [we] differ?" This question is meant to stimulate managers to think about the strategic goals of the organization. The strategic goals can be derived by taking the organizational vision and looking at the state of the world. The strategic goals are then the aspects of the world that need be changed to realize the vision. (Here we might be reminded of System 4 of the VSM, which is responsible for observing the state of the world and planning accordingly.) As the BSC model prescribes, the question is divided into four parts, one for each perspective on performance.

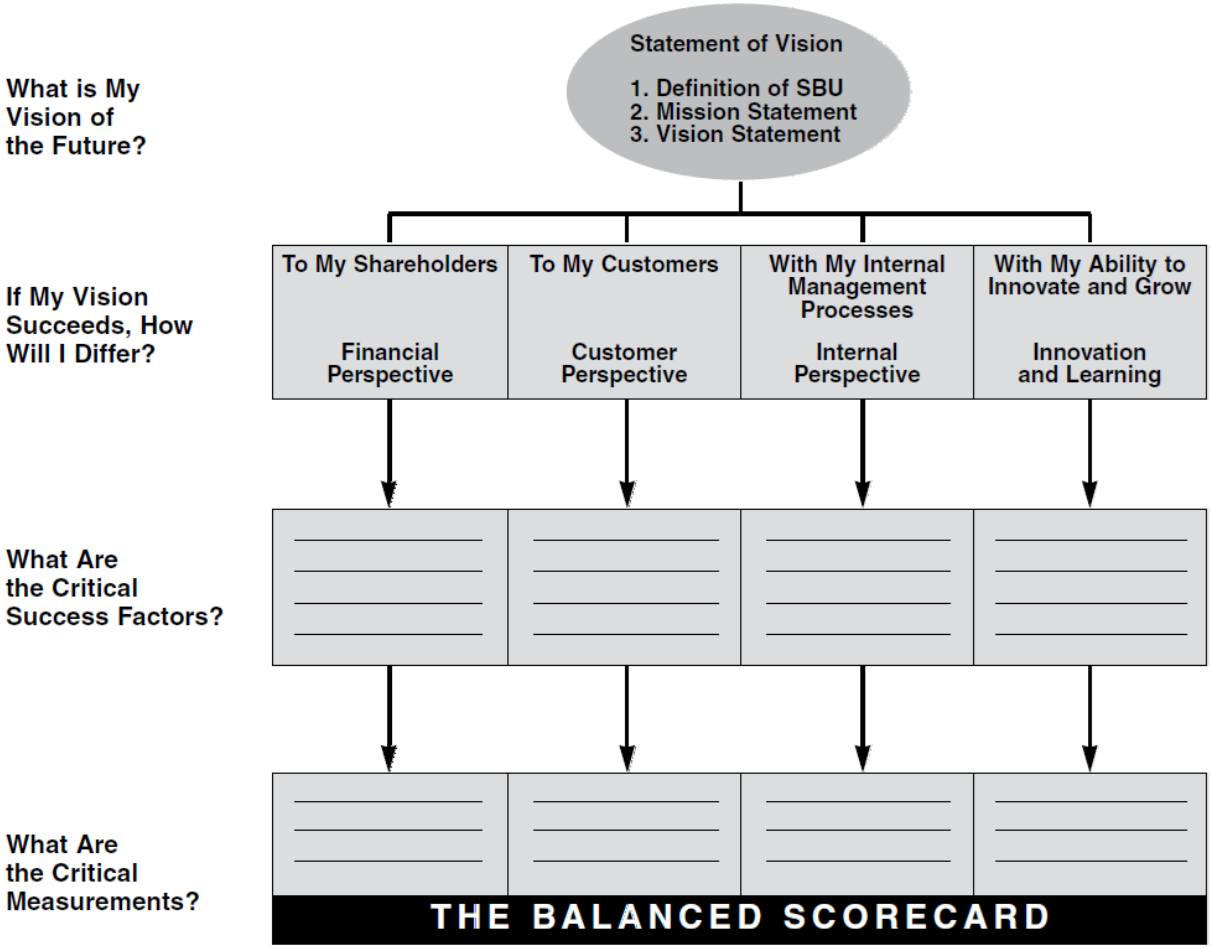


Figure 18 – An illustration of the method for applying the BSC model by Kaplan & Norton (1993)

The question of the third phase is: "what are the critical success factors?". Unfortunately, Kaplan and Norton do not explain what they mean with this and how these 'factors' are different from the strategic goals. I have interpreted the critical success factors as things that need to be the case *now*, in order to have a chance to meet the strategic goals *later*. In other words, they are the performance variables that have to be carefully monitored on a daily basis. (Here we might be reminded of System 3 of the VSM, which is tasked with daily management and control.) These factors are necessary for success but not sufficient, hence the term 'critical'. This means that while there might be other factors that determine whether or not the strategic goals will be attained, they are considered to be less influential than the critical ones.

The forth and last phase revolves around the question "What are the critical measurements?", i.e. "what are the KPIs?" According to Kaplan and Norton, there should not be more than four or five KPIs per perspective. They seem to assume that this last step is straightforward because they provide no extra guidance for this part of the process.

This concludes the discussion of the Kaplan and Norton method. The authors of the two other methods propose that some aspects of it do not match the characteristics of SME very well. I go into these methods below.

Option 2: Hudson et al.

Hudson et al. (2001, pp. 806-807) claim that a method for the implementation of the BSC model in SMES should:

a) cost very little resources;
b) provide results quickly;
c) be flexible enough to deal with the dynamics of SMEs.

The authors state that the method of Kaplan and Norton has not been made with these requirements in mind and propose their method as a suitable alternative for SMEs. See Figure 19. Like the original by Kaplan and Norton, their method is composed of four phases. During phase 1 – called 'Name' – an overview is made of the strategic goals of the company, just like phase 2 of Kaplan and Norton's method. From then on, however, the designer selects one of these strategic goals and develops suitable KPIs for it during phase 2, called 'Act'. In phase 3, 'Use', the designer ensures that data collection for the chosen KPIs is realized. Finally, in the fourth phase called 'Learn', the KPIS are evaluated. Based on the evaluations, the designer (or design team) thinks of adaptions to either the strategic goal, the KPIs or the way they are measured, after which the cycle starts anew. The next iteration serves two purposes. First and foremost, the designer chooses a new strategic goal in phase 1 to focus on. This means that new KPIs are developed for this goal, and the data collection of the KPIs is organized. The minimum number of iterations that are necessary is therefore equal to the number of strategic goals that are formulated at the start of the process. Secondly, the changes that were found to be required during the previous evaluation are implemented as well.

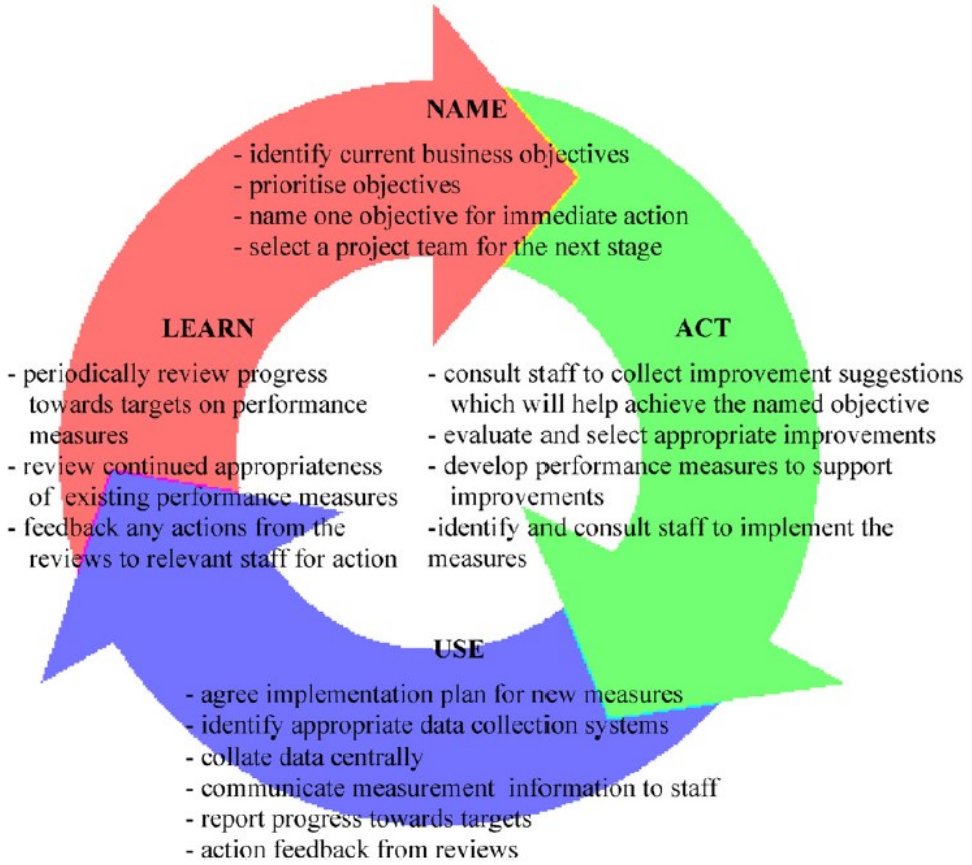


Figure 19 – The Continuous Strategic Improvement (CSI) Process for SMEs, from Hudson et al. (2001)

Hudson et al. (2001) claim that a cyclical, incremental method as theirs is much more suitable to SMEs because it produces results (i.e. actual change) very early on in the process. Fast results is one of the requirements for SMEs listed earlier. According to the authors, the reason this is important for SMEs in the first place, is that SMEs are generally very dynamic. Because of this, projects bear higher risk of being stopped prematurely, for example because key people are needed for other things or left the company. Furthermore, the authors claim that SMEs are more focused on short term results than larger organizations. Fast results are also supposed to feed the enthusiasm of the people involved and therefore establishes momentum of the project, increasing the probability that the project will actually be finished. Finally, this method supports the division of work across different teams, i.e. one team performing one iteration and another team performing the next.

The authors do not refer to a lot of scientific evidence for these supposed characteristics of SMEs. Nevertheless, from my observations at Ymor I can say that I recognize a lot of these dynamics here.

Before I make any more comparisons between this method and the other, I will first discuss the third and last alternative, namely that of Biazzo and Garengo (2012).

Option 3: Biazzo & Garengo

In their article, Biazzo and Garengo compare the two previous methods and conclude that both are based on a 'top-down' approach. By this they mean that both methods start with the identification and formulation of explicit visions, strategic goals and long term plans for the company.

According to the authors, however, people working at an SME are generally not very interested in rationalizing their often intuitive and informal strategic processes (2012). Many SMEs do not even have formal strategic plans in written form. They claim that many strategic aspects such as organizational structure are not as explicit and static in SMEs as in larger companies. That is why Biazzo and Garengo provide a method that starts by observing the current practices within the SME, and then attempts to *derive the implicit strategy of the organization from those observations.* With doing so, they write, their method is one that follows a 'bottom-up' approach instead of 'top-down'.

The method consists of two phases that each have two steps (see Figure 20). The purpose of the first phase is to unveil the current situation and the strategic goals of the company, while the second phase results in new KPIs and, if it seemed necessary, an adapted strategy.

To unveil any implicit strategic goals, the first phase starts with step 1. The purpose of this step is to model the 'individual dashboards' of the managers. 'Individual dashboards' are defined by Biazzo and Garengo as "*the aggregation of the performance measures that are utilized by single individuals to assess the activities that they are responsible for"* (Biazzo & Garengo, 2012, p. 27). When these are mapped, they can be used to construct an implicit management dashboard, which is defined as the sum of all the individual dashboards. The authors claim that this second type of dashboard is often implicit in SMEs because of two reasons. Firstly, SMEs often grow in an organic way, as opposed to tightly controlled. Secondly, the use of any performance measurements already present is often restricted to the department or team that created it. This makes it unlikely that anyone in the company has an overview of the performance measures that are being employed in the company.

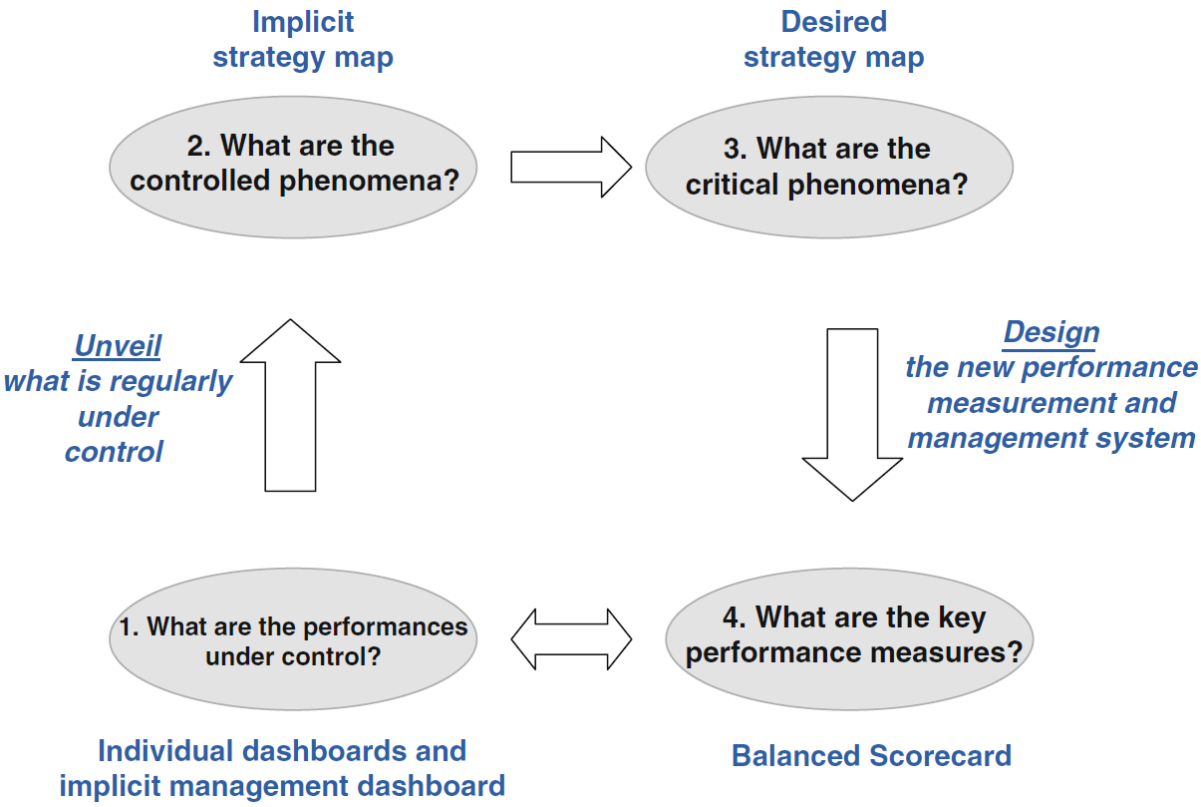


Figure 20 – An illustration of the method for applying the BSC model by Biazzo en Garengo (2012)

According to Biazzo and Garengo, the management dashboard reveals the implicit strategy of an organization. However, here the authors seem to be confused. The management dashboard actually reveals what the managers consider to be the key performance variables (or as Kaplan and Norton call them: critical success factors). As we can see in Figure 18, critical success factors and strategic goals are not the same things. The first refer to the requirements to the current state of company, while the second refer to requirements to the future state. Determining the critical success factors does not mean that we know the strategic goals, although they are probably good indicators of strategy.

This means that we need to take an additional step to find the implicit strategy and make it into an implicit Strategy Map as mentioned in Figure 20. A Strategy Map is a way of visualizing a company's strategy. It was actually Kaplan and Norton that first wrote about it. Figure 21 shows the template of a Strategy Map. As one can tell from the four layers, the Strategy Map is based on the BSC model.

The Strategy Map template in the figure shows four layers that correspond to the four perspectives of the BSC model. The idea of the Strategy Map is to fill each layer with goals associated with that perspective. As the figure implies, the goals of one layer are seen as drivers for the goals on the layer above it.

Figure 21 – A template of a Strategy Map

Phase one ends after the implicit Strategy Map has been made. Next is phase 2, the purpose of which is to design new KPIs and, if necessary, new strategy before that. This phase starts with step 3, which involves reviewing the implicit Strategy Map together with the managers. The goal is to learn from the implicit Strategy Map, and design a new, explicit Strategy Map. This reveals the true purpose of the implicit Strategy Map, which is to provide a starting point for the design of new strategy. It provides an opportunity to make new strategic choices by neatly listing the ones that were made before, which were perhaps made unconsciously. Thus the result of step 3 is a new, explicit Strategy Map that illustrates the new (or rather 'enhanced') strategic orientation.

Next is step 4, which is the final step. This step involves going back to the dashboards to identify gaps between the things that are now considered most important and the things that are being measured. Biazzo and Garengo give examples of questions that the managers might ask themselves during this step:

> *"Are the critical factors, present in the implicit map and 'confirmed' in the desired map, adequately translated by the existing indicators?* […] *Do the existing measures actually detect the phenomenon that the organisation wishes to control?"* (Biazzo & Garengo, 2012, p. 31):

The result of step 4 should be a new or enhanced set of KPIs. Because this is a method for the implementation of the BSC model, these KPIs will be divided over the four BSC perspectives.

Comparison and choice

I have described three different methods of applying the BSC model with the purpose of identifying key performance variables and developing KPIs for these variables. In all three methods, strategy plays a central role. However, the method by Biazzo and Garengo is the only one of the three that does not take strategy as a starting point but begins with the analysis of current practice. That is why the authors call it a bottom-up approach, as opposed to the top-down approach followed by the other two methods. Biazzo and Garengo describe the main benefit of their method as follows:

"In an [SME], where finding sophisticated formalisations of the strategic vision and of the entrepreneurial formula is rare and where a habit and attitude toward conceptualisations has not developed yet, [our] approach to the establishment of the future strategy map is operatively and psychologically a winning one: it enables a company to overcome the classic "blank page syndrome" and it makes the intellectual efforts to rationalize the strategic vision easier to face and overcome."

In other words, their method is supposed to be easier because it strives to redesign the current strategy, instead of starting from the beginning, for which SMEs generally lack the resources, motivation or capacity.

Compared to this, the methods of Hudson et al. (2001) and Kaplan and Norton (2000) are very alike. The most important difference is that with the former method, the strategic goals are translated to KPIs one at the time instead of in parallel like Kaplan and Norton imply. Although the method by Hudson et al. (2001) will probably produce results the fastest of the three, I estimate the situation at Ymor to be stable to such a degree that the need for the quickest results is negligible. The power of this method lies in direct implementation of measures. However, the goal of this research project is to make a MIS design, not necessarily to implement changes as fast as possible. In other words, my MIS design method is of the planned and linear type as opposed to the rapid prototyping style of Hudson et al.'s (2001) method. The reason is that a planned approach is considered to demand less of the resources at Ymor and preserves the internal coherence of the system.

For the reasons outlined above, I chose to apply the method by Biazzo and Garengo to identify the key performance variables of Delivery (and later develop matching KPIs).

3.3.2. *Applying the BSC model*

As I concluded in the previous paragraph, I have chosen the method of Biazzo and Garengo to use for the application of the BSC model. The method consists of two phases. Below I discuss each phase separately. Because the goal is to identify the key performance variables of Delivery, I stopped the process after step 3, i.e. after constructing the explicit (or 'desired') Strategy Map. Step 4 involves actually designing KPIs and falls outside the scope of this analysis. I did this at a later time and discuss this step in chapter 4.

Phase 1: an implicit Strategy Map for Ymor

I will now discuss the process of creating the implicit Strategy Map of Ymor as a whole. As Biazzo and Garengo's method prescribes, this requires that the individual dashboards of the managers are determined first. To do this I conducted semi-structured interviews with the managers of Delivery (two DMs, four SDMs), Sales (the Technical Accounts manager, the Sales department manager), and of Software Development (the Team Leader). These departments are the largest three and account for nearly all people working at Ymor. The questions in these interviews were designed to identify what these managers consider to be the critical success factors and strategic goals, both on the level of Ymor as a whole as well as for their respective departments. I also added questions that focus specifically on the success factors and strategic goals of the Delivery department. To clarify, this last set of questions were asked to every interviewee, whether they were part of Delivery or not. The idea was that multiple perspectives would enrich the view on the purpose and strategic role of the Delivery department.

After conducting these interviews I compiled the results and made the implicit Strategy Map, see Figure 22. The implicit Strategy Map is of Ymor as a whole, not just for Delivery. This ensures that we

get the 'whole picture' before we zoom in on the strategic orientation of the Delivery department with the explicit Strategy Map. Yes, the *explicit* Strategy Map, which I introduce in the next section, is focused on Delivery. After all, this project is concerned with designing a MIS for Delivery and for that I need to know the key performance variables of Delivery. The reason for making the implicit Strategy Map for Ymor as a whole is that it makes it easier to see the role that Delivery is supposed to play. This in turn helps with formulating the explicit goals and success factors for Delivery.

I will now continue with a discussion of the implicit Strategy Map in Figure 22, starting with the upper layer and going down from there.

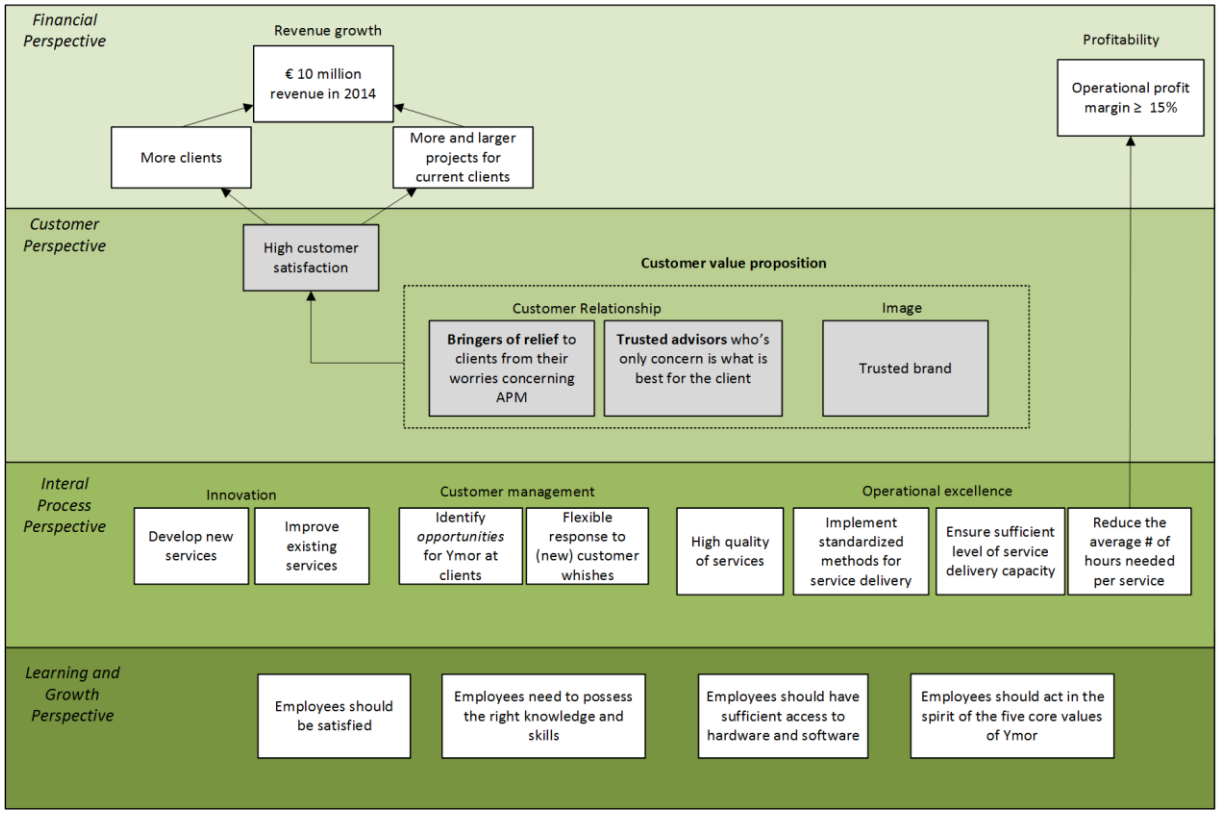


Figure 22 – The implicit Strategy Map for Ymor as a whole, based on the results of the interviews

The financial perspective

The financial goals of Ymor are known to everyone at the company, although the two Sales managers had a lot more to say about these goals than the other interviewees. There were two main goals for 2014. On the one hand, there was the goal to reach a revenue figure of at least €10 million. The other goal was to keep the operational profit margin at 15% or higher. Thus the second goal involves profitability. On the short term, the revenue goal is considered to have a higher priority.

At the time of writing, the people at Ymor were working hard to reach these goals, especially the people at the Sales department. They focused on two things: attracting more new clients on the one hand, and increasing the amount and size of the projects sold to existing clients on the other.

The customer's perspective

With the exception of the Team Leader of the Software Development department, every interviewee spoke a lot about Ymor's goals for its customers (i.e. clients). Both of the Sales managers stressed the importance of company image and high levels of trust between Ymor and its clients. When it comes

to company image, it is Ymor's goal to be known as a) a group of specialists that relief clients of their Application Performance Monitoring (APM) worries, and b) as trusted advisors that give advice that is best for the client, regardless of the consequences for Ymor.

As Figure 22 shows, when it comes to the customers the final goal is customer satisfaction. All DMs and SDMs answered that everything that they do at the Delivery deparment is directed to this goal. The relation between customer satisfaction and both acquiring new customers and sustaining existing customers is assumed to be positive. The relation between EBIT margin and customer satisfaction was not mentioned by any of the interviewees, although it seems safe to assume that high levels of trust at the customers is a competitive advantage and enables higher prices.

The 'internal processes'-perspective

What needs to happen within Ymor to reach the customer- and financial goals? That is the question that I asked the interviewees to determine the goals for Ymor's internal processes, the third layer of the implicit Strategy Map in Figure 22.

The goals in this perspective that I identified can be divided over three categories: 'innovation', 'customer management' and 'operational excellence'. The categories are also shown in Figure 22. The managers spoke about innovation in roughly two different ways. The Sales managers saw innovation as responding to (new) needs in the market in some way, e.g. by developing and providing new services. But to the DMs and SDMs, innovation had a slightly different meaning. To them, innovation referred to the ability to solve unexpected technical problems that they stumble upon during project (i.e. during service delivery). Solving that problem means that innovation took place, because from then on the people at Delivery have learned and are capable of solving similar problems more effectively. I will return to the subject of innovation in more depth later.

The second category of internal processes goals is 'customer management'. Although they are about customers, these goals are from an internal perspective and therefore belong in this layer instead of the previous one. Nevertheless, it was mostly the Sales managers who mentioned these goals. This should be no surprise as this falls within their responsibilities and contribute directly to the goals in the second layer.

The last category is 'Operational Excellence'. The four strategic goals that fall under this were mainly mentioned by the DMs and SDMs, which should not be surprising since reaching these goals is part of their responsibility as managers. (In contrast, it is still unclear who is responsible for the previously mentioned innovation goals.) The four Operational Excellence goals are: improving the quality of services, implementing the standardized methods for the delivery of these services, ensuring the right amount of capacity of the Delivery department, and reducing the average number of hours needed to deliver each service.

As Figure 22 shows, the relations between the goals in the Internal Process layer and the goals of the Customer layer were left undefined, except that the former are drivers of the latter. This ambiguity reflects the answers of the interviewees; the assumption is that they are critical for success, but the exact relation with customer goals is not specified by anyone at Ymor.

The 'Learning and growth'-perspective

The last layer in Figure 22 corresponds with the 'learning and growth' perspective of the BSC model. The elements in these layers are the goals or requirements to the organizational resources that enable Ymor to reach the goals in the third layer. Again, the exact relation between the elements of the

different layers have not been specified by the interviewees. Instead, they seem to assume that attaining the goals in the fourth layer drive or enable the goals in the third.

The first goal in this layer is that Ymor employees should be satisfied with their work at Ymor, because if the work is enjoyable then employees are assumed to be more motivated to work towards the other goals. The second goal that the managers mentioned is that the Ymor employees need to possess the right knowledge and skills. What exactly these are is often not clear, but I will return to this subject in a later paragraph. Ensuring that people have the right capabilities is not a trivial task, especially not for Delivery employees since that work is highly specialized; people with the right experience are hard to find. The third goal is that the people at Ymor should have sufficient access to the kinds of hardware and software that they need to do their job. This goal also mainly concerns the people at Delivery, because the delivery of the services often requires the availability of specialized software tools. The same goes for hardware such as computers, smartphones and other devices. Finally, the last goal in this layer is that all Ymor personnel acts in the spirit of the five core values of Ymor. In English, these are perhaps best described as 'willingness', 'trustworthiness', 'ambition', 'focused on results' and 'quality'.

Phase 2: an explicit Strategy Map for the Delivery department

I based the previous Strategy Map on the data from interviews, which I compiled and interpreted. That is why it is called the *implicit* Strategy Map. The next phase of Biazzo and Garengo involves making a new, *explicit* Strategy Map with the managers involved. In this case I involved the DMs, since they will be the users of the upcoming MIS. In accordance with the method, the implicit Strategy Map was taken as a starting point to design of the new explicit Strategy Map. The result is shown in Figure 23.

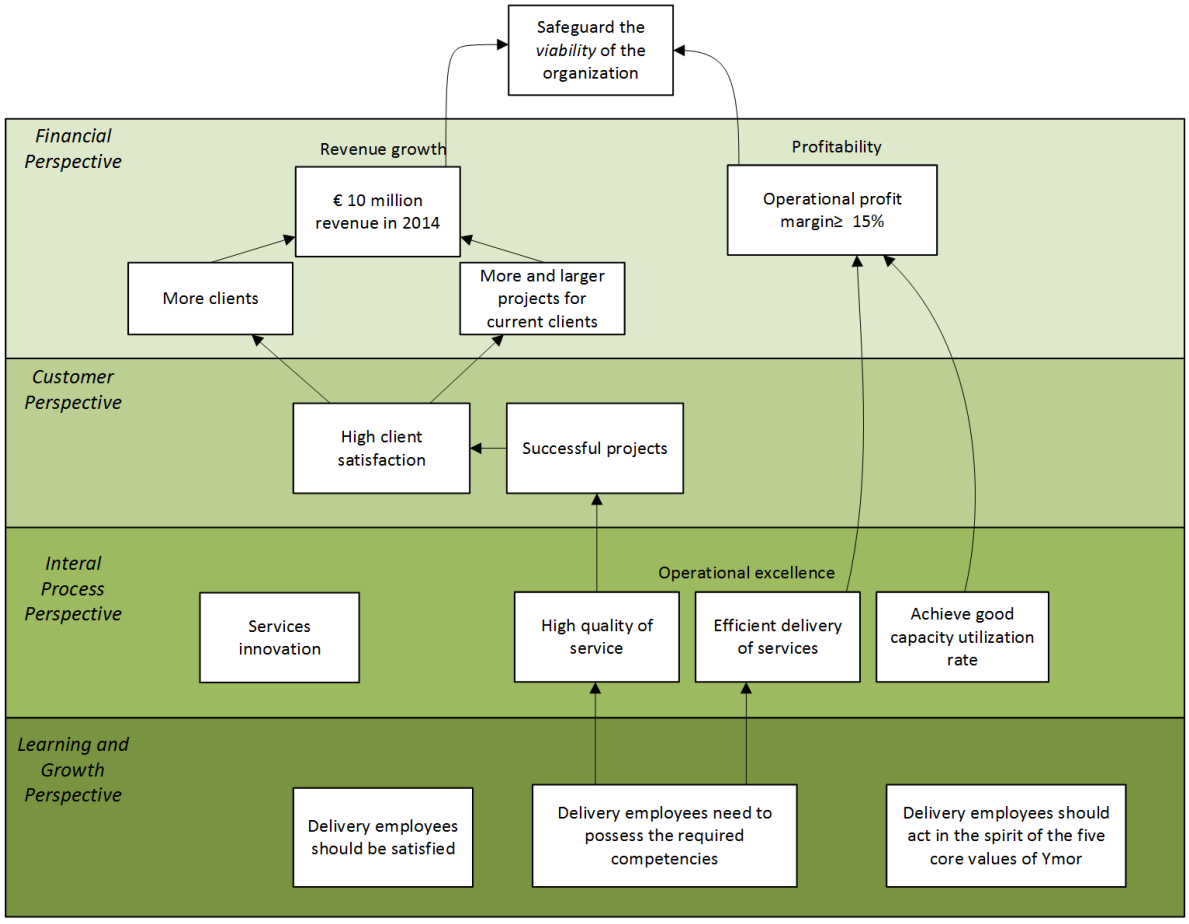


Figure 23 – The explicit Strategy Map for the Delivery department, constructed together with the DMs

The financial perspective
The two financial goals in Figure 23 are identical to those in Figure 22. After all, these are the final goals of Ymor and ultimately everything is supposed to help attain them. The apparent duality of the two goals in Figure 22 is solved by defining a higher purpose: ensuring the viability of the organization. I suggest that this approach should be used if issues arise that concern the priority between the two financial goals.

All activities of the Delivery department should contribute to the achievement of the financial goals. However, the criteria used by the Management Team to determine the performance of Delivery is customer satisfaction. In other words, the DMs are responsible for high customer satisfaction but not for its effects on the financial goals.

The customer's perspective
According to the interviewed managers, the Delivery department is not the only part of Ymor that influences customer satisfaction. Almost every department contributes to it somehow. After discussing this issue, the DMs and I concluded that the Delivery department contributes to customer satisfaction by successfully finishing projects (i.e. delivering services). The explicit Strategy Map for Delivery in Figure 23 illustrates this. Another factor that contributes to client satisfaction is the corporate identity (i.e. image) of Ymor shown in the implicit Strategy Map. However, this was left out of the explicit Strategy Map for the Delivery department because it falls under the responsibility of Sales.

Thus 'project success' was the first identified performance variable of the Delivery department. The rest of the identified performance variables are in the two bottom layers of the Strategy Map in Figure 23, which I discuss below.

The 'internal processes'-perspective
A large part of the internal processes of Ymor falls within the borders of the Delivery department. This explains why in this layer many elements of the implicit Strategy Map in Figure 22 are included in the explicit Strategy Map for Delivery in Figure 23. However, after discussions with the DMs, I chose to formulate some of these elements differently.

The first goal of Delivery from this perspective is that it should innovate its own services. Unfortunately, the design of the innovation goals and processes have received little attention yet, so it was not possible to define a norm for it. Nevertheless, the DMs consider innovativeness to be one of the key performance variables, which makes it the second to be identified thus far. The apparent gap between the importance of innovativeness and the lack of organization of innovation processes is discussed in section 3.3.3.

The second Delivery goal from this perspective is achieving a high quality of service. After discussing this goal with the DMs and SDMs it became apparent that there is much overlap between quality and project succes (the first goal in the customer's perspective layer). Quality was often defined as the extent to which a project (i.e. service) meets the client's expectations. But this was also the definition of project success. Because of this overlap I decided to not add quality as a key performance variable when project success is already one.

The third goal concerning the internal processes of Delivery is to deliver services as efficiently as possible, so that higher profit margins are achieved. The efficiency of the Delivery department is therefore also considered one of the key performance variables, and is the third to be identified as such.

The fourth and final ´internal process´ goal is achieving a good capacity utilization rate of the Delivery department. With a good capacity utilization rate, there is little unused capacity that is making unnecessary costs. But this rate should not be as high as possible either, because some unused capacity is important to deal with fluctuations in demand. I return to this subject in chapter four, where I define the KPIs for the performance variables, plus their norms. For now, it is enough to note that the capacity utilization rate is considered one of the key performance variables, which makes it the fourth thus far.

The 'Learning and growth'-perspective

The three goals in the fourth layer of Figure 23 are also part of the implicit Strategy Map in Figure 22. There are however two differences. The first is that the goals Figure 22 refer to all Ymor employees, while the explicit Strategy Map refers to Delivery employees only. This should be no surprise since the former was made for Ymor as a whole, while the latter was made specifically for the Delivery department. The second difference is that the words ´knowledge and skills´ in Figure 22 are replaced by the word ´competencies´ in Figure 23. The meaning stays the same. The reason for this change is that I later made use of a model that defines the relation between services and competencies (see section 3.3.4), and using different terms would be unnecessarily confusing. The working definition of competencies that I used is 'all personal features that influence the degree to which a person can perform a certain activity', e.g. knowledge, skills, attitude, et cetera.

Another difference between Figure 22 and Figure 23 is that one goal has not been included in the latter, namely the goal that people should have sufficient access to the hardware and software that they need for their activities. The reason this goal is excluded is because it is still unclear which department should be responsible. In the current situation, Sales, Delivery and IT Support are involved in the matter. Furthermore, hardware and software capacity is rarely the 'bottleneck' of service delivery because any shortages are solved within one or two days by simply ordering. Any surpluses are put into stock and saved for later use.

The three goals that are in fact shown in Figure 23 are: 'to have satisfied employees', 'coverage of competencies' and 'to act in accordance of the five core values'. These are all considered to be key performance variables of the Delivery department. This brings the total number of identified performance variables to seven.

Summary: the performance variables of Delivery

I will now first summarize the findings of this second analysis phase.

By applying the BSC model using the method of Biazzo and Garengo, I tried to find an answer to the following questions:

Question 1. What are the financial performance variables of the Delivery department?

Question 2. What are the customer-related performance variables of the Delivery department?

Question 3. What are the performance variables related to the internal processes of the Delivery department?

Question 4. What are the resources-related performance variables of the Delivery department?

Following the method of Biazzo and Garengo, I first constructed an implicit Strategy Map for Ymor as a whole, based on interviews with managers from the three largest departments: Sales, Delivery and

Software Development. This implicit map was the starting point for the collaborated effort to construct an *explicit* Strategy Map. In other words, the DMs and myself took the implicit map to analyze the current role of Delivery in the greater system (Ymor) and derived from that an explicit Strategy Map for the Delivery department.

The following key performance variables were identified with the explicit Strategy Map:

1) Project successfulness
2) Innovativeness
3) Project efficiency
4) Capacity utilization
5) Personnel satisfaction
6) Coverage of competencies
7) Integrity[2]

Number one and three refer to the average degree of success and efficiency of all the projects that are carried out by people of the Delivery department. Number two refers to the innovativeness of the department as a whole.

In the following section, I will review these variables to determine the usefulness of this list for our purpose, i.e. designing a Lockean MIS.

3.3.3. A review of the performance variables

According to the data from the interviews, there exists quite a lot of agreement among the DMs and SDMs about the meaning of 'project success', 'efficiency', 'capacity utilization' and 'personnel satisfaction'. Decisions about these matters can thus be regarded as *structured*, and are suitable for a Lockean MIS. However, the last of those four, 'personnel satisfaction', was not taken into account when I later designed the MIS. The reason being that Ymor has involved a third party to measure this variable by means of interviews and surveys. Thus the DMs will be able to make informed decisions about this variable without the use of the MIS under design.

Unfortunately, there does not seem to be a lot of agreement between the DMs and SDMs about the meaning of the other performance variables, i.e. 'innovativeness', 'possession of required competencies' and 'integrity'. This means that any decisions involving these variables are *unstructured*, or wicked, at least to some degree. According to Mason and Mitroff, this makes them unsuitable for Lockean MIS (see chapter 2). If the Lockean MIS is to provide useful information about these phenomena, then the ambiguity that surrounds them must first be removed somehow. For this project, I chose to do this for number six, i.e. possession of required competencies, but not for innovativeness or integrity. I will now discuss why.

The meaning of innovativeness at Delivery is unclear because there has been almost no attempt to define the innovation goals and processes of the department. Innovations are essentially regarded as semi-spontaneous and beneficial events that happen during or in between service delivery activities. There is little or nothing known about what management decisions are involved with increasing

[2] Which is defined for this purpose as 'the degree to which Delivery personel acts in accordance with the company's core values'

innovativeness, let alone what their outcomes might be and what the utility of these outcomes are. This is problematic, because innovativeness was identified as a key performance variable. It might be necessary for the DMs to focus on this subject in the coming months.

The other variable left out of the design of the MIS is integrity. The clarity issues around 'integrity' are comparable to those of 'innovativeness'. However, with something as personal and subjective as integrity, it seems that strong agreement on what integrity means for the daily activities at Delivery is unlikely, and should even be approached with suspicion. After a discussion with the DM, we agreed that decisions about integrity are inherently wicked and should therefore not be reduced to something that is measurable with a Lockean MIS. Instead, a Kantian MIS would be more appropriate, which works by comparing different perspectives on the matter. In fact, one could say such a system is already in place at Ymor; the yearly performance reviews between managers and their employees has a part where both parties present their view on matters like attitude and integrity. If the DMs ever feel the need to make the degree of integrity of Delivery employees more controllable, then my suggestion would be to enhance these performance reviews and/or increase their frequency.

Thus, innovativeness and integrity are judged to be outside the domain of the forthcoming Lockean MIS. However, the coverage of competencies is not. The coverage of competencies is the degree to which there are enough people with the competencies required to deliver the services. The DMs already make decisions involving competences on a regularly basis (as opposed to innovativeness). It serves a clear purpose, namely to ensure that Delivery personnel are capable of delivering services successfully. Furthermore, competencies are arguably not as subjective as integrity, so defining the meaning of this variable is feasible.

What makes decisions involving the coverage of competencies unstructured is that is not fully known what exactly all the relevant competencies are. In the next section I describe my method and attempt to identify them.

3.3.4. *Identification of relevant competencies*

Although the DMs obviously had some idea of what the relevant competencies are, they never made a list. The HR-department has a list of competencies but they define competencies as general skills and attitudes. These are not directly related to specific activities at Delivery. That is why I have attempted to make a list of specific competencies required for the activities of Delivery personnel. I will first explain the method that I have used for this end. This is followed by a discussion of the process with which I applied the method to the situation at the Delivery department.

Method

I have created a method for the identification of relevant competencies at service providers. The method is based on the model of services of Gallouj en Weinstein (1997). Figure 24 shows the illustration of their model.

According to Gallouj and Weinstein, services consist of a set of service characteristics, which are shown in the figure as the Y variables. For the realization of each service characteristic, one or more competencies are required. The competencies are shown as C-variables in the figure. Next to competencies, some (but not necessarily all) service characteristics require one or more technologies, i.e. the T-variables. Technologies can be divided into two categories: material technologies such as machines and other hardware, or immaterial technologies such as software or methods. Finally, sometimes it is necessary that the client possesses certain competencies in order for the services to be delivered successfully.

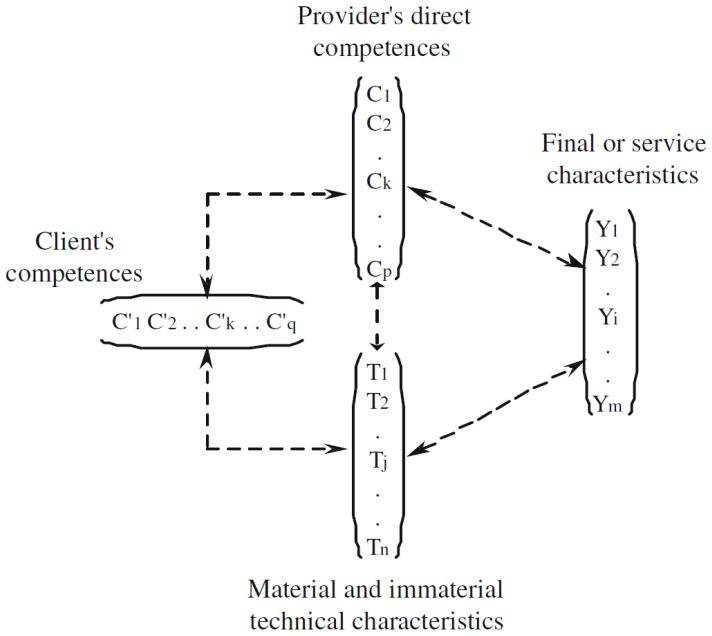


Figure 24 – Model of the characteristics of a service, copied from Gallouj and Savona (2009)

The model suggests that the relevant competencies for a service provider are determined by the service characteristics of its services, the technologies used, and the competencies of its clients. In other words, the relevant competencies for Delivery can be identified by finding all the relevant C', T and Y variables. Each unique variable can require a unique competence. Note that this also goes for each combination of these variables. This means that, theoretically, the amount of relevant competencies could be enormous because there are many combinations possible with only a small amount of C', T and Y variables. On the other hand, many of these combinations could require roughly the same competencies, so this does not necessarily have to be the case.

To make the above less abstract, I will now describe an example of the use of the model for the Delivery department. Suppose that a client requests the help of Ymor for the management of the performance of their software applications. As part of their service, Ymor agrees with the client that they will monitor the performance (i.e. response time) and availability of said application. In other words, Ymor and the client agreed on a certain service characteristic (i.e. a Y-variable). Suppose that the client has previously tried to do this themselves last year, and that they purchased monitoring

software called 'Foglight' for this purpose. Although the client has since decided to let Ymor take care of the matter, they requested that Ymor use Foglight, mainly because those licenses have been bought already and it has been partly implemented. Not using Foglight would be a waste. What this means is that the technology characteristic for this particular service has been set as well (i.e. a T-variable). Lastly, suppose that the client is a small municipality with little expertise with APM and IT technology in general. This might remind us that an understanding of APM terms and relevant IT technologies is a required client competency to realize this particular service (i.e. a C' variable). This means that Ymor might need to provide workshops at the end of the project to teach the people at the client about the proper use of the monitoring system that they have set up.

If we take the previous specification of a service (in this case one C', one T and one Y), we can derive the required competencies of the service provider (i.e. the C variables). For example, a required competency is that the service provider must be able to apply the Foglight software for monitoring application performance. Another required competency of the service provider is to be able to give workshops or otherwise explain APM terms to relative laymen.

The previous example also illustrates what makes identifying all relevant C variables so complex. The first required competency that I derived from this example was to be able to use Foglight (a T-variable) for performance monitoring (a Y-variable). But Foglight can be used for many more things. This means that this particular competency is *the result of the combination of two variables.* Having worked with Foglight is not necessarily enough; one needs to be able to apply it in a certain way. The same principle applies to experience with performance monitoring: it is not enough to know what needs to be done if you are unfamiliar with Foglight. That is why the required competency is a result of the combination of different characteristics.

Having said that, not all required competencies have to be associated with *combinations* of C', T and/or Y-variables. In fact, we can decide ourselves how complex this competency-identification process will be. To make it as simple as possible, we could choose to neglect combinations altogether. This means that every relevant competency is derived from a single characteristic (i.e. a single C', T or Y variable). If we would apply this simple approach to the previous example, we would identify the following competencies:

- Being able to use Foglight (derived from the T variable in the example);
- Being able to set up performance monitoring (derived from the Y variable in the example);
- Being able to train people (derived from the C' variable in the example);

Many services have strongly overlapping C', T and Y variables. For example, a service that uses Foglight for performance *testing* has an overlapping technology characteristic (i.e. Foglight) with a service that uses Foglight for performance monitoring. They use the same technology. Because this simple approach disregards combinations, it will result in the identification of a smaller amount of general competencies that are more general.

At the other extreme is the most complex approach to the identification of relevant competencies (i.e. C variables). With this approach, every combination of C', T and Y variables results in the identification of a unique competency. Applying this approach to the previous example would result in the identification of just one relevant competency: 'being able to apply Foglight to set up performance monitoring for a client with little APM expertise'. As we can see, this competency is very specific when compared to the three competencies that were identified using the simple approach. The complex approach will result in a very large list because every combination of technology, service characteristic and client competency (i.e. Y, T and C' variables) requires a different competency (i.e. C-variable).

Until now I only discussed the four sets of variables of the Gallouj and Weinstein's model. But according to De Vries (2006), their model needs to be enhanced by adding a fifth set, namely the client's technologies (i.e. C'T variables). See Figure 25. The reason for this extra set is that, during the last ten to fifteen years, IT technology has come to play a central role in companies and other organizations. Many contemporary services companies provide their services by letting the client use their technology to access their own. This addition to Gallouj and Weinstein's model is useful for our purpose, because it is true that the delivery of Ymor's services depends greatly on the type of IT systems that a particular client owns.

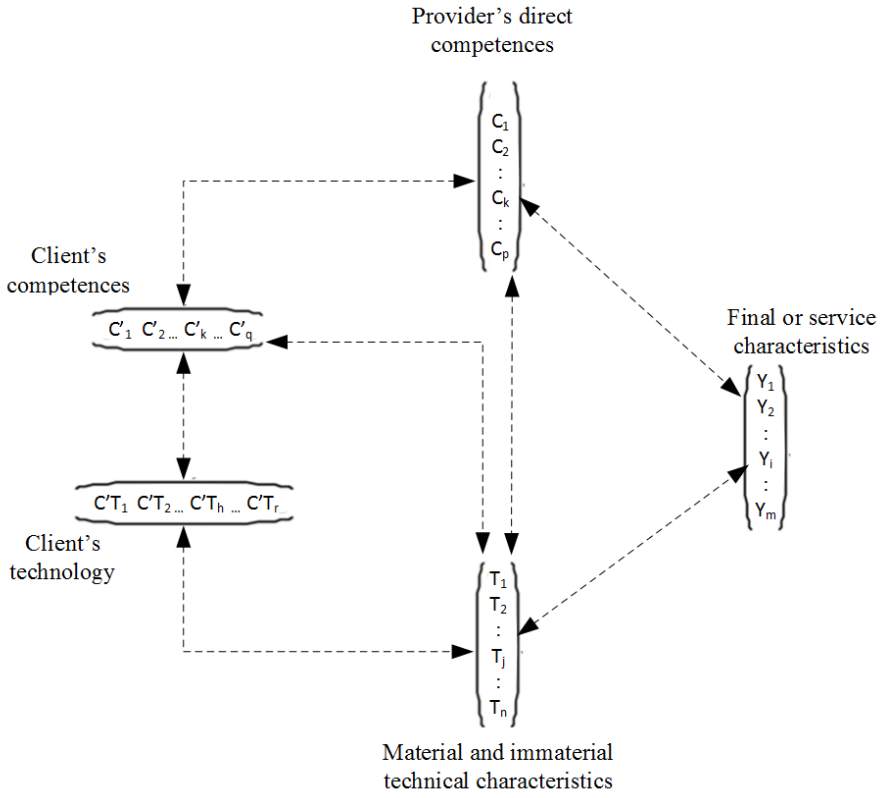


Figure 25 – A visual representation of De Vries' version of the service model, made by De Vries (2006)

The method that I made for the identification of relevant competencies is based on the reasoning that I have outline thus far. To summarize, the method consists of the following steps:

Step 1. Identify the service-characteristics of the service provider (i.e. Y-variables);
Step 2. Identify the technologies that the service provider applies (i.e. T-variables);
Step 3. Identify the necessary client competences (i.e. C'-variables);
Step 4. Identify the relevant client technologies (i.e. C'T-variables);
Step 5. Select the combinations of these variables that require the definition of a unique competency. (This step is essentially a way of controlling the complexity.)
Step 6. Determine the required competencies (i.e. C variables) for every variable identified in steps 1 to 4 and for every combination of variables identified in step 5.

Result

Because of the limited time available for the identification of relevant competencies during this project, I decided to skip step 3 and step 4. These steps would take more time than step 1 and 2 because the DMs and SDMs at Delivery have thus far spent little time thinking about what the relevant

client competences and client technologies are. They had a better idea of what the numerous service characteristics (Y-variables) of Ymor are and what technologies are used within the department. These things are considered to be more important too, which is also why step 1 and 2 were included but step 3 and 4 were not.

Below I have detailed the results of each step.

Step 1

The first step involves the identification of the service characteristics of Ymor's services. For this I sought the help of a Technical Account Manager (TAM), who works for the Sales department. TAMs are Sales people who are also very informed when it comes to the processes and tools with which the people at the Delivery department actually deliver Ymor's services. They specialize in linking customer needs to Ymor's capabilities, which is why a TAM should be the ideal person to help with the identification of the service-characteristics that together form Ymor's services.

When asked about the characteristics that define Ymor's services, the TAM showed me the image in Figure 26. It turned out that the TAMs had asked themselves a similar question some weeks before, and the 'service circle' in Figure 26 was their first attempt at an answer.

The inner most circle in Figure 26 reads 'gebruikersgeluk', which roughly translates to 'end-user happiness'. This is the goal of every service of Ymor: to ensure the happiness of their client's end-users (i.e. the people using the client's applications) by helping their clients manage the performance of their software applications. Slow responses or low availability of applications are seen as deterrents to end-user happiness. Ymor helps by preventing these kinds of performance problems.

The second circle, i.e. the one that encompasses the first, shows the three services that Ymor provides to their clients to achieve the end goal of user happiness. These are called 'Ymonitor', 'Yvalidate', and 'Troubleshoot'. These terms are rather loosely defined, and two Ymonitor services could be quite different from each other. However, each Ymonitor service revolves around the monitoring of the performance and availability of one or more applications. Usually performance is defined as the response time, while availability refers to the percentage of time the application was functional. Yvalidate services, in turn, revolve around performance tests of applications to see their behavior under a certain load, or to find out under what load the application stops working. Finally, Troubleshoot services always involve an investigation into seemingly unexplainable and/or persistent performance problems.

As I have said, these are loose definitions, and services of one kind are often expanded with elements of the other. This goes especially for the Ymonitor service, which is often a required part of Yvalidate and Troubleshoot services. Furthermore, services of the same kind still be quite different. In the words of the model of Gallouj and Weinstein: although they are called the same, they can still consist of different *service characteristics* (i.e. Y-variables). For example, some Ymonitor projects include diagnostic monitoring activities, and some do not. Some other Ymonitor projects revolve around end-user monitoring instead. The total set of possible service characteristics that together compromise Ymor's services are shown in the outer layer of Figure 26, i.e. the largest circle.

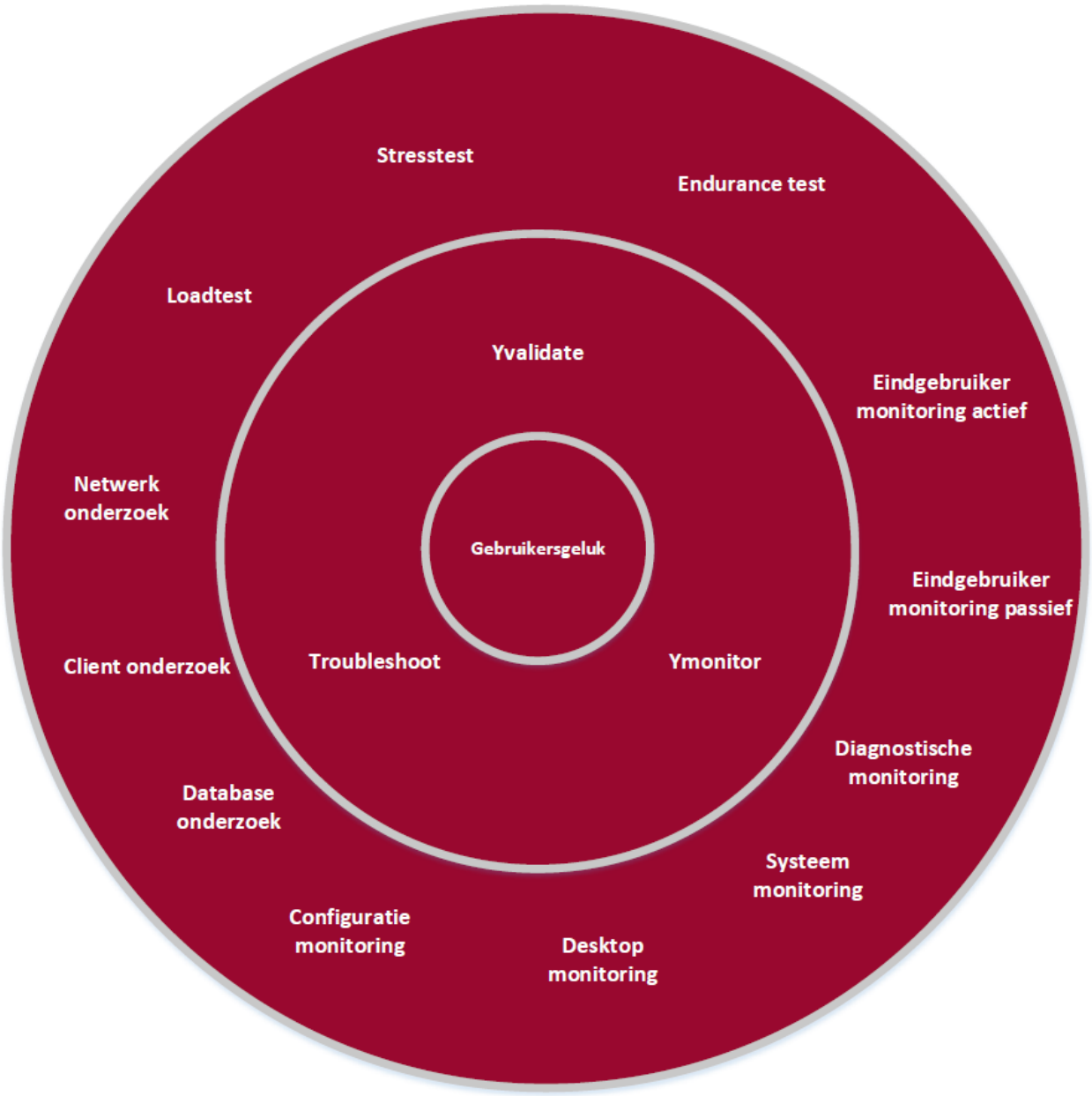


Figure 26 – The first version of Ymor's 'service circle', designed by the TAMs of Ymor (in Dutch)

Thus at this point I had a list of relevant service characteristics, i.e. Y variables. To check if this list was complete, and to see if the meaning of the service characteristics on the outer layer was unambiguous to Delivery employees, I discussed Figure 26 with the DMs, SDMs and a few senior team members. All of them largely agreed on the contents of the outer circle. However, they saw two small problems with it.

According to them, the first problem with Figure 26 had to do with the three service characteristics nearest to 'Troubleshooting', i.e. 'Network investigation, 'Client investigation' and 'Database investigation'. The problem was that these terms could be interpreted in a few different ways. They thought it likely that there might be some types of research missing, depending on your interpretation. The second problem was that distinguishing between three types of performance tests (i.e. 'Stress test', 'Load test', and 'Endurance test') was unnecessary because they require almost the exact same

competencies. For this reason I have combined these three and call them ‘Performance tests’ from now on.

The resulting list of service characteristics that I took into account are shown in Figure 27. As can be seen from the list, the problematic service characteristics (Y8, Y9 and Y10) are included. However, I suggest that the DMs, SDMs and TAMs review these in order to ensure that any ambiguity surrounding the terms is removed. This might result in a new version of the list.

Y_1: Performance tests
Y_2: End-user monitoring (active)
Y_3: End-user monitoring (passive)
Y_4: Diagnostic monitoring
Y_5: System monitoring
Y_6: Desktop monitoring
Y_7: Configuration monitoring
Y_8: Database investigation
Y_9: Network investigation
Y_{10}: Client investigation

Figure 27 – A list of service characterstics that form the Ymonitor-, Yvalidate- and Troubleshoot services

Step 2

With the service characteristics listed I continued with step 2, which involved the identification of the different technologies employed by Delivery personnel. There were two types: 1) methods and standardized approaches to the realization of certain service characteristics, and 2) software tools. The first type are sometimes called process characteristics, which is a sub-type of technology characteristics according to Gallouj and Weinstein.

I have derived the first type technology characteristics through analysis of the standardized methods of Yvalidate services (Y_1) and Ymonitor services (Y_2 to Y_7). The method for performance tests consists of the following phases:

1. Project intake
2. Technical realization
3. Testing and retesting
4. Analysis of test results

The other method is focuses on the monitoring service characteristics, i.e. Y_2 to Y_7. (At the time of this project, there were no standardized methods for the last three service characteristics, i.e. Y_8 to Y_{10}.) This method consists of the following phases, regardless of the type of monitoring involved:

1. Project intake
2. Technical realization
3. Exploitation
4. Governance

The complete list of process characteristics is shown in Figure 28.

Technologies (methods and phases)
Y_1: performance tests
P_1: Project intake
P_2: Technical realization
P_3: Tests and retests
P_4: Analysis of test results

$Y_2 – Y_7$: monitoring
P_1: Project intake
P_2: Technical realization
P_5: Tests and retests
P_6: Analysis of test results

Figure 28 – A list of phases for the delivery of Ymonitor and Yvalidate services

The second type of technology characteristics is software tools. The list in Figure 29 was made in collaboration with a senior Delivery employee who is also called the Tools Specialist.

Technologies (software)
T_1: Ymonitor
T_2: Silkperfomer
T_3: AutoHotkey
T_4: DynaTrace
T_5: Evolven
T_6: DCRum
T_7: NextThink
T_8: Splunk
T_9: WUG
T_{10}: Precise
T_{11}: Exinda
T_{12}: Foglight
T_{13}: Jmeter
T_{14}: HP Loadrunner

Figure 29 – A list of software tools used to deliver Ymonitor-, Yvalidate- and Troubleshoot services

Step 5 and step 6
As I have stated at the start of this process, I have chosen to skip steps 3 and 4 because they need more in-depth investigation, for which there was too little time.

Step 5 involves the selection of *combinations of service characteristics and technologies* that require unique competencies. After a discussion with the DMs and SDMs, I decided to only take into account the combinations of service characteristics and the phases of their respective methods. This means that a unique competency was defined for every combination of the P variables and Y variables (disregarding the last three). For example, one of the resulting competencies was the ability to perform the ´Project Intake´ phase (P_1) of a 'Diagnostic Monitoring' service (Y_4). Another is the ability to complete a 'Tests and Retests' phase (P_3) of a 'Performance Test' (Y_1). There were 28 competencies identified this way (see left column of Figure 30).

The rest of the competences (see the right column in Figure 30) that I identified are directly related to either one of the remaining service characteristics (Y_8 till Y_{10}) or the software tools (T_1 till T_{14}). This resulted in 17 other competencies, for example the competency to 'perform a database investigation' or 'to work with DynaTrace'. Of course this implies that someone who is capable of using DynaTrace can apply this tool to every type of service. That is the assumption behind the current list of competences in Figure 30. As I discuss in chapter 5, these assumptions will have to be evaluated sometime.

C_1: Intake phase for performance test(s) C_2: Technical realization of performance test(s) C_3: Execution of tests and retests C_4: Analysis of test results C_5: Intake phase for end-user monitoring (active) C_6: Technical realization of end-user monitoring (active) C_7: Exploitation of end-user monitoring (active) C_8: Governance of end-user monitoring (active) C_9: Intake phase for end-user monitoring (passive) C_{10}: Technical realization of end-user monitoring (passive) C_{11}: Exploitation of end-user monitoring (passive) C_{12}: Governance of end-user monitoring (passive) C_{13}: Intake phase for diagnostic monitoring C_{14}: Technical realization of diagnostic monitoring C_{15}: Exploitation of diagnostic monitoring C_{16}: Governance of diagnostic monitoring C_{17}: Intake phase for system monitoring C_{18}: Technical realization of system monitoring C_{19}: Exploitation of system monitoring C_{20}: Governance of system monitoring C_{21}: Intake phase for desktop monitoring C_{22}: Technical realization of desktop monitoring C_{23}: Exploitation of desktop monitoring C_{24}: Governance of desktop monitoring C_{25}: Intake phase for configuration monitoring C_{26}: Technical realization of configuration monitoring C_{27}: Exploitation of configuration monitoring C_{28}: Governance of configuration monitoring	C_{29}: database investigation C_{30}: network investigation C_{31}: client investigation C_{32}: being able to use Ymonitor C_{33}: being able to use SilkPerformer C_{34}: being able to use AutoHotkey C_{35}: being able to use DynaTrace C_{36}: being able to use Evolven C_{37}: being able to use DCRum C_{38}: being able to use NextThink C_{39}: being able to use Splunk C_{40}: being able to use WUG C_{41}: being able to use Precise C_{42}: being able to use Exinda C_{43}: being able to use Foglight C_{44}: being able to use Jmeter C_{45}: being able to use HP Loadrunner

Figure 30 – The complete list of identified competencies required by Delivery to delivery Ymor's services

3.4. Conclusion

The goal of this chapter was two answer the first research question of this project, which is:

Research question 1) *What are the functional requirements of the MIS for the Delivery department at Ymor?*

I have attempted to answer this question by analyzing the most suitable value of the five key variables of MIS, using a method I created for this purpose. The result is summarized in Figure 31.

Key variables of a MIS	Appropriate value of the variable for this situation
Psychological type of the users:	'Thinking'-'Sensing'
Mode of presentation of information:	Computer-generated reports
Evidence generation method:	Lockean
Organizational context:	System 3 of the Delivery department
Class of problems:	Structured

Figure 31 – A table with the results of the two analysis phases

As Figure 31 shows, the MIS under design will be a Lockean system to meet the need for computer-generated reports based on measurement data. This need is expressed by the DMS, who are regarded as being of the 'Thinking'-'Sensing' type. The context within the MIS will operate is defined as System 3 of the Delivery system (i.e. Delivery department). System 3 is one of the subsystems of the VSM and is involved with the daily control of the department, which in this case mainly consists of delivery teams. Finally, the decision problems that the DMs were supposed to be able to make with the support of the MIS turned out to be of the unstructured type. All that was known was that the decision problems involved the performance of the Delivery department. But it was not clear what 'performance' meant in this situation.

Because unstructured decision problems are unsuitable for a Lockean MIS, I have attempted to make them into structured decision problems. This meant that the 'performance of Delivery department' needed to be defined into multiple performance variables. This was done using the BSC model, and the resulting list of performance variables was:

1) Project successfulness
2) Innovativeness
3) Project efficiency
4) Capacity utilization
5) Personnel satisfaction
6) Coverage of competencies
7) Integrity

Unfortunately, numbers 2 ('Innovativeness') and 7 ('Integrity') turned out to be unsuitable for a Lockean MIS, which is why they were left out of the scope of the design. Furthermore, it was recently decided that number 5 ('Personnel satisfaction') was to be measured by a third party that specialized in measuring these matters through surveys and interviews. This means that this variable was also left out of the scope of the MIS design. At the end, the forthcoming MIS turned out to be suitable for only four out of seven of the performance variables. Or, to put it in terms of decision problems: the forthcoming MIS will be designed to provide information regarding:

a) decisions related to improving the average 'project succes' of the Delivery department;
b) decisions related to improving the 'efficiency' of the Delivery department;
c) decisions related to improving the 'capacity utilization rate' of the Delivery department, and;
d) decisions related to improving the 'coverage of competencies' of the Delivery department;

Finally, it is important to note that the MIS will not provide information on every facet of these decisions. As was outlined in chapter 2, a decision problem is "*to choose from among a set of acts* $A_1, \ldots, A_m$ *the one which optimizes the decision-maker's return* U_{ij} *, where* U_{ij} *is the utility or value of the outcome* O_{ij} *corresponding to the doublet* (A, S_j)*, where* $\{S_j\}$ *is the set of the 'states of nature'."* (Mason & Mitroff, 1973, p. 479). With this MIS design project, it is assumed that the set of actions {A},

outcomes {O} and utilities of those outcomes {U} are more or less known to the DMs (i.e. the decision makers). The MIS under design will be responsible for providing information on {S} – the set of the 'states of nature' – by means of measurement.

In chapter four, I describe a design of a MIS that meets the requirements set in this chapter.

4. Design

In this chapter, I describe the four steps I took to specify the MIS design. These steps are:

1. Design KPIs for the four key performance variables identified in chapter 3.
2. Design the data structure of the MIS
3. Describe (and design) the data sources of the MIS
4. Design the structure of the reports that the MIS should be able to produce

The steps are based on design steps described by Wieringa (2003). Wieringa identifies three levels in software product design: Business Solution Specification, Software Requirements Specification and Software Decomposition Specification. The result of each level is the basis of the next. Only the last level involves actual software programming.

Chapter three of this report contains a description of the Business Solution Specification. I expand on this business level solution in paragraph, where I detail the design of KPIs (step 1 in the list above). The three other steps of the list – designing the data structure, data sources and reports of the MIS – are all part of the Software Requirement Specification. My design of the MIS ends there, which means it should be ready to be used by software developers to implement the MIS.

Each of the four steps is discussed in one of the four paragraphs of this chapter. Although the previous list implies a linear approach, I have actually gone through multiple cycles of design. After every cycle I requested feedback from the envisioned users (i.e. the DMs) as well as the people that will be partly responsible for the supply of data (e.g. SDMs). It is hoped that through these feedback-design cycles, the validation of the design has been ensured. The results described in this chapter correspond to the final design iteration.

4.1. Key Performance Indicators (KPIs)

Each of the four performance variables identified in chapter 3 required one or more KPIs to make it measurable.

As opposed to the physical sciences, many of the variables in the social/management domain are hard to measure, (e.g. 'employee satisfaction' and 'project success'). In those cases, measurable indicators of these variables need to be found. An indicator is an observable sign of the value of a certain variable. An indicator for project success, for example, would be the final grade that the customer would give if you would ask him to define his satisfaction with the project on a scale from 1 to 10. Because 'project success' is one of the variables that together define the performance of the Delivery department, the grade is a **performance indicator**. Some indicators are more reliable and/or valid signs of their associated variable than others. The subset of indicators that are the most reliable and valid are called the **Key Performance Indicators**, or KPIs.

To design KPIs means the following things:

1. To have a solid understanding of the most important performance variables of the organization in question (e.g. project success)
2. To select one or more indicators per variable (e.g. project grade)
3. To define a norm for each of the selected indicators (e.g. at least an 8 on a scale of 1 to 10)

Step 1 has already been discussed in chapter 3, although I will repeat the meaning of each of the four performance variables in the following sections. The main focus here is to select one or more KPIs and to define their norm.

4.1.1. *Project success*

As the Strategy Map in the previous chapter showed, successful service delivery projects is assumed to lead to higher customer satisfaction. Customer satisfaction is the final goal of the Delivery department, but measuring it is currently the MT's responsibility. Because increasing customer satisfaction is ultimately also the goal of other departments, the DMs decided to focus on their influence on this variable, which is through successful service delivery project (or 'projects', for short).

Because of the importance of this variable, the DMs and SDMs already have a way of measuring it. They do this by sending a small survey to the client after the completion of every project. The survey consists of a few questions regarding a few aspects of the service delivery such as timeliness, quality, et cetera. The final question asks the client to give an overall grade of the project's results. The DMs and SDMs use this grade as the indicator for project success.

Every Delivery team starts and finishes projects every month. This means that quite a few (different) grades are received on a monthly basis. A suitable KPI of project success on the level of the whole Delivery department, then, is the average of all these projects. Likewise, a good KPI on the level of a single Delivery team is the average grade received from the clients by that team.

The calculation method for this KPI is summarized in Figure 32. The norm for this KPI as already been decided by the MT of Ymor. The goal is to reach an average project grade of 8 or higher, on a scale from 1 to 10. Because there is some ambiguity about rounding, an average between 7.5 and 8 is considered a risk, but not yet problematic.

The average project grade as a KPI for project success:

$$G_{\text{Delivery}} = \frac{\sum_{i=1}^{n} G_i}{n}$$

Where,

G_i	=	the grade given by the client of project *i*;
n	=	the total number of service delivery projects considered;
G_{Delivery}	=	the average project grade attained by Delivery.

The same principles apply to the calculation of this KPI on the level of single teams.

Norm table for G_{Delivery}:

***G* (grade)**	**Judgement**
≥ 8	Good
7.5 – 8	Risk
< 7.5	Problematic

Figure 32 – The calculation method of the KPI for project success: 'average project grade'

4.1.2. *Efficiency*

The second performance variable is efficiency. In abstract terms, efficiency refers to the amount of input that is required to produce a certain output. In the case of businesses, input is costly, so increasing the efficiency of a (sub-)organization saves money. This logic corresponds to the relation between efficiency and profit margin depicted in the Strategy Map of the previous chapter.

The largest part of costs at Delivery is formed by the wages of its personnel. A good indicator of efficiency would therefore focus on the 'amount' of personnel used to produce the Ymor services. In this case, the 'amount' of personnel used to produce a service is defined in terms of number of hours.

Of course a possible alternative would be to measure any changes in output instead of input. But service delivery projects are always should by Sales with a *fixed price*. In other words, the output level is fixed. It therefore makes more sense to focus on measuring the number of hours needed for each project in order to enable management decisions that are aimed at reducing this number.

Unfortunately it is not possible to set a fixed norm for the amount of hours that can be spent on a project, because each project differs in size and scope. Instead, an indicator must focus on the deviation of the actual amount of hours spent from the budgeted amount of hours. For example, suppose that 250 hours are budgeted for a certain project. If the actual amount of hours spent turns out to be 200, then only 80% of the budget has been used. This a good result in terms of efficiency.

The average percentage of budgeted hours that are actually spent as a KPI for *efficiency*:

$$A_{\text{Delivery}} = \frac{\sum_{i=1}^{n} A_i}{n}$$

where,

$A_i = \frac{W_i}{P_i} \times 100\% =$ the devation from the hours budget per project i, in %

$W_i =$ Actually amount of hours spent on project i

$P_i =$ Predetermined hours budget for project i

$n =$ the total number of service delivery projects considered;

The same principles apply to the calculation of this KPI on the level of single teams.

Norm table for $A_{Delivery}$:

Value	Judgement
≤ 100%	Good
100% – 105%	Risk
> 105%	Problematic

Figure 33 – The calculation method of the first KPI for efficiency: 'budget deviation'

The resulting KPI for efficiency is shown in Figure 33. It is the deviation from the budget, defined as a percentage of the budgeted amount of hours. A percentage lower than 100% means less hours have been used than the amount budgeted, while a percentage larger than 100% means more time was

spent than planned for. Note that this KPI assumes that the way the amount of budgeted hours is decided is reasonably accurate and reliable. (This is arguably a weakness, but one that needs to be accepted for this project.)

From the feedback of the DMs and SDMs, it became clear that an alternative KPI was desired as well, one that measures the actual hourly rate of a service delivery project. Of course this hourly rate is only a virtual one, since the projects are sold for a fixed price and not on an hourly rate basis.

The norm that the DMs have set is an hourly rate of €100. The actual hourly rate can be calculated by dividing the fixed price by the actual amount of hours spent on the project. Notice that the latter is also an important factor in the first KPI for efficiency. They are very much a like, but still tell a slightly different story.

The most important benefit of this second KPI is that it provides a more direct link to the financial goals of the company, i.e. the profit margin. After all, the largest part of the costs of service delivery at Ymor is determined by the number of hours spent on the project. If the norm of €100 an hour is indeed sufficient to reach the profit margin goals, then this KPI can be used to determine if this goal is under pressure or not.

Unfortunately, this KPI is also not without its weaknesses. Sales can have different reasons for selling a project for a fixed price that is actually too low to make a profit. One of the main reasons for doing so is to gain the trust of a potentially big client by first doing a (smaller) project for a discounted price. The small loss suffered from these projects are assumed to be covered by the large gains of future projects for this client. This is arguably a good long-term strategy, but it also means that one of the KPIs of Delivery might show bad performance when it is actually out of their control. This is why the use of the first KPI is important: it could help determine if bad values are due to low selling prices by Sales, or because of actual problems with Delivery.

Another weakness of this KPI is that a higher value is not always better. If the virtual hourly rate reaches very high numbers, this could be a sign that a client is being billed but little or no work is actually being done. Of course this must be prevented from happening, which is why I introduced a ceiling on the desired value of this KPI.

Figure 34 summarized the calculation of this second KPI for efficiency. As can be seen, the virtual hourly rate should not fall under €100, but very high values should also be viewed with suspicion. The €200-€300 and >€300 boundaries are based on experiences of the DMs. According to them, experience shows that hourly rates of between €200 and €300 indicate a serious risk of problems with adequate service delivery to clients, while values of €300 and above are almost certainly problematic.

One final note: the hours spent on a project must include the hours spent by managers and other people that are not directly part of a certain delivery team.

The average actual hourly rate as a second KPI for efficiency:

$$U_{\text{Delivery}} = \frac{\sum_{i=1}^{n} U_i}{n}$$

Where,

$U_i = \frac{K_i}{W_i} = \text{Actual hourly rate per project } i$

$K_i = \text{Selling price per project } i \text{ in } €$

$W_i = \text{Actual amount of hours spent on project } i$

$n =$ the total number of service delivey projects

The same principles apply to the calculation of this KPI on the level of single teams.

Norm table for $U_{Delivery}$:

Value	Judgement
< 90	Problematic
90 – 100	Risk
100 - 200	Good
200-300	Risk
> 300	Problematic

Figure 34 – The calculation method of the second KPI for efficiency: 'average actual hourly rate'

4.1.3. *Capacity utilization*

The capacity utilization rate is defined here as the ratio of required capacity to available capacity (Corsten & Stuhlmann, 1998, p. 163). Because of the knowledge-intensive nature of the service delivery work, capacity is defined in terms of available man-hours, or alternatively some other time-based unit like FTEs. Capacity utilization is a performance variable for Delivery because it influences both project success and profitability. When almost all of the capacity is utilized, i.e. during 'busy times', project success is put under pressure because there is little time left to be flexible and deal with problems like sudden technical obstacles. Furthermore, with demand for Delivery's services being somewhat unpredictable, high capacity utilization would mean that the department is not able to deal with sudden increases in demand. For these reasons, a low capacity utilization should be preferred. But of course the lower it goes, the more unnecessary overhead costs are made, which is how capacity utilization influences profitability.

To calculate the capacity utilization rate requires the measurement of available capacity, as can be read from the above definition. Measuring capacity can be done in two ways: by measuring output or by measuring input. Which way is chosen depends on the type of organization. As Pycraft et al. explain:

"In high-volume, repetitive, low-variety operations, output measures of capacity are often preferred, because of their predictable relationships to the required input resources, and because actual and

forecast sales are usually defined in terms of quantity of outputs (for example, cars per month). In complex operations producing a wide variety of outputs, each requiring different inputs, measures of capacity based on inputs are usually considered to be most appropriate." (2000, p. 384)

Operations at the Delivery department are of the second type mentioned by Pycraft et al. By far the most important input at Delivery operations are human resources. That is why in this case capacity is defined in terms of time available. There are of course other inputs distinguishable, such as software and hardware, but they play much smaller role and are also much more scalable.

Of course, there are different kinds of human resources at Delivery. HRM has linked every Delivery employee to a function profile. These describe the responsibilities and some required competencies for people with a particular profile. Each function profile also has a certain salary range associated with it. Disregarding SDMs and DMs, each employee at Delivery has one of these five function profiles:

1. Performance Director
2. Performance Architect
3. Performance Consultant
4. *Ketenbewaking* Specialist
5. *Ketenbewaking* Specialist Trainee

'Ketenbewaking' is Ymor's trademark and translates roughly to 'safeguarding the digital supply chain'.

Delivery employees with different function profiles each play a different role in the realization of Ymor's services. So instead of simply measuring the total amount of FTEs present at Delivery and calling that Delivery's capacity, we could use the function profiles to distinguish different inputs and thus different types of capacity. If such measurements would be implemented, we might find that Delivery's capacity regarding Performance Consultants is meeting demand with a comfortable margin, while Delivery's capacity of Specialist Trainees is too high because there is not enough for them to do.

Another way of measuring capacity of the Delivery department in more detail is by measuring the capacity of separate teams. If we assume that each team has approximately the same composition of function profiles (which is one the policies of the DMs), we could simply measure the capacity of a team by taking the sum of the available hours or FTEs in that team.

Whatever way capacity is measured, the capacity utilization rate is calculated in the same way. Figure 35 summarizes the calculation method of this KPI.

The norm table is based on capacity utilization data provided by the US Federal Reserve (2014), which suggests that for manufacturing companies the maximum value of capacity utilization rate is around 85% during strong years for the economy. Some suggest that this number is a little bit higher for service providers (Bittencourt, 2013). In any case, every company has to aim for a capacity utilization rate that results in the necessary flexibility while at the same time keeping overhead costs to a minimum. Widely used queuing models suggest as capacity utilization rates increase, queue's become exponentially longer (Bassett, 1992, pp. 41–45). A capacity utilization rate of Delivery that nears 100% is therefore not desirable. Especially when we consider that innovation is another performance variable for Delivery, because creating innovations also takes time.

Capacity utilization rate calculation (per function profile) as KPIs :

$$CUR_f \text{ in period } t = \frac{TS}{TA}$$

Where,

$\mathrm{CUR}_f =$ Capacity Utilization Rate

$\mathrm{TS} =$ total amount of time spent per function profile *f* on service delivery during period *t*

$\mathrm{TA} =$ total amount of time available per function profile *f* for service delivery during period *t*

Norm table for CUR_f:

Value	Judgement
≤ 70%	Problematic
70% – 75%	Risk
75% – 85%	Good
85% – 90%	Risk
≥ 90%	Problematic

Figure 35 – The calculation method of the KPI for capacity utilization

At the end the DMs have to decide on the norm for this KPI. Perhaps it should be lower than 80, or perhaps higher. My recommendation is to measure this KPI for a while and look for correlations between certain values of the capacity utilization rate and good values of other KPIs such as the average project grade. That way the DMs could learn about the capacity utilization rate that is optimal for their department.

If for some reason there is still doubt about what the norm should be, I would like to stress that Ymor strives to be seen as a flexible organization that provides quality services. Ymor is not competing on cost, at least not primarily. This might be useful to remember if there is any doubt between lowering costs and increasing capacity. In general, a small amount of over-capacity seems to be justified.

4.1.4. *Coverage of competencies*

What 'competencies' means was already explained in chapter three. The main challenge for this performance variable was to find a suitable way to measure the competencies coverage.

After discussing this matter with one of the DMs, who is the main client of this project, we decided on the use of small internal survey. The goal of the survey is to measure the amount of people that possess each identified competence, i.e. the coverage of competencies. The survey that I designed is added to the appendix of this document, see Appendix A. Note that this is just a design. I recommend that the final implementation will be via a web-application instead of a separate word document. The benefit of such an implementation is that the survey is easily accessible and data is fed directly to the back-end (i.e. database).

The people that are supposed to fill in the survey are mainly the SDMs. They are asked to estimate for each of their team members their level of expertise with every competency. The five levels are 'Novice', 'Advanced Beginner', 'Competent', 'Proficient' and 'Expert'. This scale is developed by Dreyfus and Dreyfus (1980). The survey contains a description of each of these levels.

The SDMs are chosen as the ones that fill in the survey because their role suits it. They are team leaders and already responsible of planning training of their team members. Their teams are small enough for them to know their team members quite well. They should therefore be more than capable of estimating their levels of expertise with different competencies. Nevertheless, the final goal of the survey is that not only the SDMs fill the survey in for each team member, but every team member as well. This is often called 360 feedback (e.g. Campion et al., 2011). The power of 360 feedback is that multiple perspectives are considered, which is important for a variable as hard to measure as 'competency'. If the survey is used in this way, then that means that a Kantian aspect is added to the Lockean MIS I have designed.

Based on the data that the SDMs will generate through this survey, the DMs will have information on the coverage of competencies in their department. Again we stumble upon difficulties when we attempt to define a norm. How many people possessing a certain competency (on a certain level) does Delivery need? To answer this question a few things need to be done.

Firstly, for every service delivery project it needs to be clear what the agreed upon service characteristics are. As it is now, every project is called either an 'Ymonitor', 'Yvalidate' or 'Troubleshoot' project, or a combination of these. These labels need to be more detailed. What kind of Ymonitor project are we talking about? What are its service characteristics? As we saw in chapter three, it could be diagnostic monitoring, (passive) end-user monitoring, et cetera. These are different things requiring different competencies.

Secondly, it needs to be clear for every service characteristics how many people with certain levels of expertise are required to realize them. For example, the 'diagnostic monitoring' service characteristic might require at least two people with a 'proficient' level of expertise when it comes to 'DynaTrace', an important diagnostic monitoring software tool.

Only when the competency requirements are known for each service characteristic, and the service characteristics are known of each project, can the DMs calculate a norm for this KPI. Accomplishing this will most likely require the help of people from Sales, who are responsible for describing the goals of each service delivery project. They could therefore help with the first step that I described here.

4.2. Data structure of the MIS

Having specified the measures that the MIS is required to process, I now turn to specifying its data structure. I have modelled this data structure using the Entity-Relationship notation. The result is called an Entity-Relationship Diagram (ERD).

An ERD shows the entities of the outside world that the MIS will store information about. There are is often a relation between these entities that is relevant to the MIS, and the ERD illustrates these relations as well. Finally, the ERD also shows the attributes of the entities that we are interested in.

The ERD for the MIS under design is shown in Figure 36. It is important to note that there are many more relations between the entities in reality than depicted in the ERD. Furthermore, entity relations that are included in the ERD are very often actually much more complex. For example, the ERD

suggests that the SDM does not possess any competencies but obviously this is not the case at all! However, the competences of the SDMs are not directly related to carrying out service delivery operations, but instead involve the management thereof. This example shows that the purpose of the ERD is not to give an accurate description of reality, but to determine what information about the world the MIS needs to serve its purpose.

I will now continue with a short explanation of the most important elements in the ERD.

The ERD in Figure 36 contains multiple entities with different kinds of relations. Each relation shows the cardinality, i.e. the number of sets of an entity that can be related to another entity. For example, the relation between 'customer' and 'project' shows a one-to-many relation, meaning that one customer can be related to zero or more projects, but each project is related to one and only one customer.

Furthermore, three entities in the ERD are of a special type called 'associative entities'. The difference between these and normal entities is that the normal ones have independent identity, while the associative entities are identified by the two entities it associates. This means that associative entities have properties of relations as well as properties of an entity. For example, the associative entity called 'Skill level' in the ERD illustrates the relation between competencies and Delivery employees. So in this case the relation has a name, which is 'Skill level'. Note that the associative entities lack a unique ID. Instead, they are identified by the combination of the two primary keys of the entities that it associates. In the case of the previous example, these are *competencyID* and *employeeID*. The associative entities are necessary because without them there would be a many-to-many-relation between the two normal entities. Many-to-many relations are traditionally avoided to prevent later problems with database design.

Another element in Figure 36 that might require explanation is the special relation between 'Delivery Employee', 'SDM' and 'Employee'. The lines and circle are meant to show that the former two are subtypes of the latter. In other words, 'Employee' is their super-type, which means that every 'Delivery Employee' and 'SDM' inherits the attributes 'Employee'. The circle containing a *'d'* means that there is no overlap between the subtypes. So an SDM cannot also be a Delivery Employee at the same time. The dotted line pattern of the circle indicates that the instances of the sub-entities can migrate, e.g. an SDM can become a Delivery Employee (in which case it ceases to be an SDM).

The last elements of Figure 36 that require clarification are the entities' attributes. Of course, every entity has in reality many more attributes, but the ones included in the ERD are the attitudes that matter to the purpose of the MIS. Together with relations, storing the value of each entity's attributes enables the calculation of KPIs by the system. For example, for the average project grade KPI, the MIS uses the values of the 'customer grade' attribute of every project in a certain time period. In contrast, for the two efficiency KPIs the relevant attributes are each Project's 'selling price', 'hour budget' and the sum of 'total time spent' by associated employees. When it comes to the calculation of the capacity utilization KPI, all that is necessary are the data from the Delivery Employee's 'FTEs' attribute, the sum of 'total time spent' by employees, and the relation between the Delivery Employees and their function profile or team (depending on the KPIs calculation method, see paragraph 4.1). Lastly, storing each Delivery Employee's skill level per competence enables count the value of the coverage of competencies KPI.

Finally, one might ask where the norms for the KPI are stored. However the norms are not part of the data structure of the MIS. I regard them as part of future settings of the MIS interface, where they affect the way data is displayed.

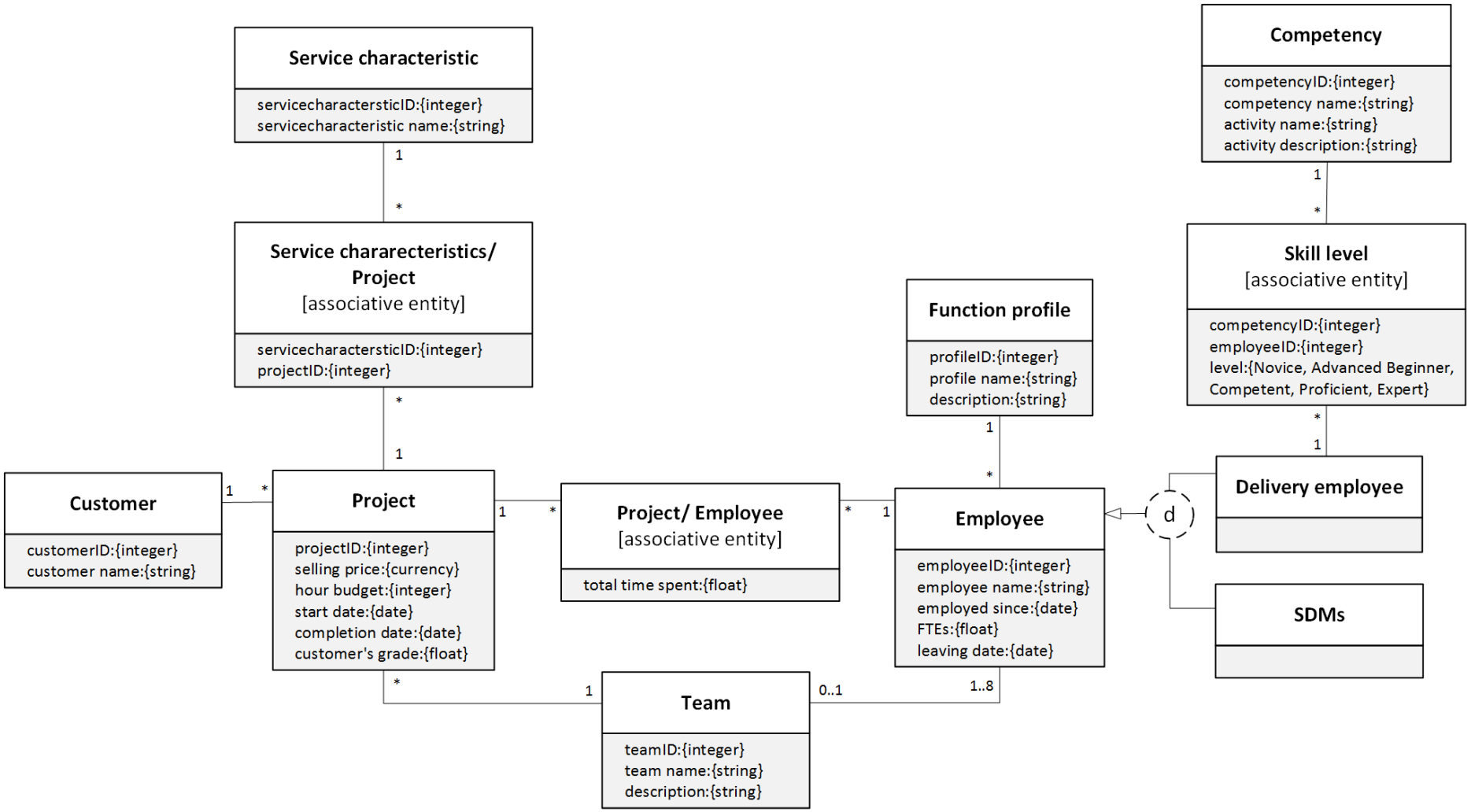


Figure 36 – An ERD for the MIS under design

Now that the data required for the functioning of the upcoming MIS have been specified, it is time to address the issue of where this data should come from. I discuss this matter in the next paragraph.

4.3. Data sources of the MIS

After the data structure of the MIS it was time to model the data sources (Wieringa, 2003). I have done this by constructing context diagrams of the MIS.

A context diagram represents the system under design within its context by showing its communications with external entities. I have made a different context diagram of the MIS for every KPI, because a single diagram showing everything would be too crowded. Thus each context diagram in this paragraph shows the necessary communication with external entities to enable the calculation of a particular KPI.

Every context diagram contains a 'connection domain' and 'subject domain'. The connection domain is perhaps best described as the interface between the MIS and the external environment. It is generally difficult to decide if the components in the connection domain are part of the system under design or not. In other words, the connection domain is a description of the boundary between the system and its environment.

The subject domain is the part of the environment that contains the actors and other entities where the MIS will receive information *from* and/or *about*. This means that some of the entities in the subject domain never communicate directly with the MIS, but the MIS will have information about them and that is why they are relevant. For other entities in the subject domain, the reverse is true: their role is to communicate information about other entities to the MIS. They are relevant not because the MIS stores information about them but because the MIS receives information from them.

According to Wieringa, context diagrams should be descriptive models. That means that they should describe how the present situation is, as opposed to prescriptive models that describe what the situation should become. They context diagrams that I present in this paragraph are mostly descriptive. However sometimes the situation at Delivery just did not enable the measurement of a certain KPI, so in those cases I chose to add a few prescriptive elements to suggest a solution. These parts of the context diagrams are colored red, so that it is clear that they are not actually descriptions of the current situation but of a desired future situation.

The next sections each contain a discussion of a single performance variable and their data sources, illustrated with context diagrams.

4.3.1. *Project success*

Before the start of this research project, the SDMs already used a survey to measure project success. The survey is sent after the completion of each service delivery project and asks the client to give the project a grade between 1 and 10. That grade is the measure of that project´s success. The KPI for project success on the level of the Delivery department is the average of these grades.

Thus the MIS requires the data contained in the filled-in surveys. Unfortunately, although it is known how the data is generated, it is currently not clear where the data is saved. Sometimes the surveys are saved on Ymor's cloud-storage environment, but sometimes the surveys are not saved anywhere and they remain in the email accounts of the SDMs. Those surveys that are in fact saved in the cloud-environment are not always easy to find because they are located in different folders and have different file-names.

This means that the context diagram needed to be added with a prescriptive element to account for these problems, or else the MIS design would not have access to the data required for this KPI. See Figure 37 for the context diagram. The figure shows a subject domain with the SDMs and the clients having communications between each other, just as I explained previously. When the SDM has received the filled-in survey (which contains the data that the MIS needs), he is ready to save the data in a way that is compatible with the MIS. Because this was not yet the case, I decided to add a suggestion to the context diagram. This is the red element in the connection domain. As can be seen from there, I suggest that the SDMs save the filled-in survey to a single destination within the cloud-storage environment, using a naming convention for the file (e.g. 'CLIENT_TEAM_GRADE.docx'). If this is done, the only thing that the MIS would be able to do is to regularly check that destination (i.e. folder) for new documents and download them.

I have suggested this option because it requires the least amount of change in the way the SDMs and everyone else work. However, another option that should be considered is to convert the survey into a simple web-application and ask the SDMs to refer their clients to it instead of sending survey documents via email. The clients would then fill in their answers via the web-application and save the data directly into the MIS. This option clearly has some benefits but it also requires more work. For one, the web-application must be built, and two the SDMs will have to stop sending surveys like they are used to and direct their clients to the web-app instead.

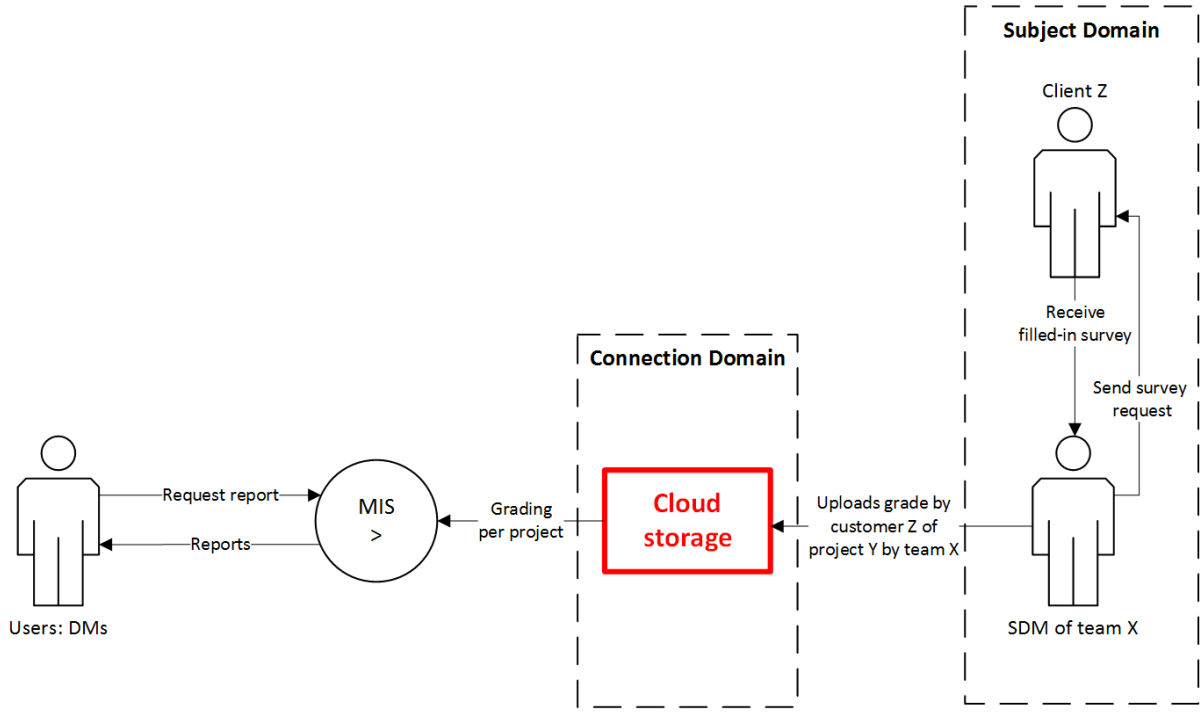


Figure 37 – A context diagram of the MIS with a focus on the KPI for Project Success

4.3.2. *Efficiency*

As discussed in paragraph 4.1, both KPIs for efficiency can be calculated per department (i.e. for all projects) or per team (i.e. for all projects done by a certain team). Either option requires approximately the same data: a) the selling price per project, b) the total amount of time spent on the project), and c) the amount of hours budgeted per project. As with the KPI for project success, these data are already available at Ymor. Fortunately, in this case they are even already stored in other MIS. See Figure 38.

'Sales MIS' stands for the MIS used by the Sales department. In the case of Ymor, the system in question is called 'Salesforce'. The Sales people store information about service delivery projects sold, but also about opportunities that are likely to become actual projects in the future.

The 'Project ERP system' is the system used by multiple departments to store different types of information about Ymor's projects. The system in question is called 'Navision' at Ymor. This system contains information about current or previous projects, including project ID codes, client names, hour budgets, hours spent, et cetera.

Finally, Ymor's employees register the amount of hours they worked via an hour registration system, which at Ymor is called 'Timesheet'. Timesheet is accessed through a simple web page where employees can fill in the amount of hours and also what they were spent those on (e.g. 'project *X*' but also 'internal training' et cetera).

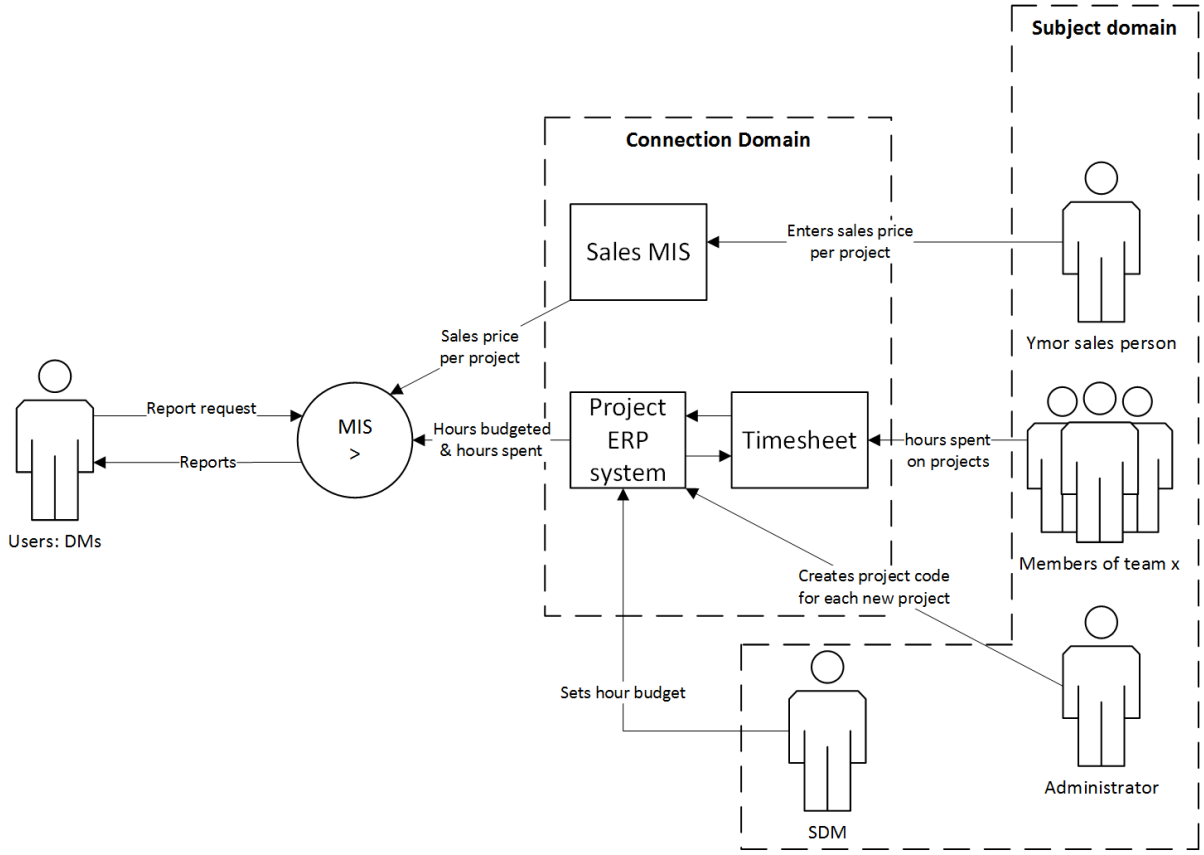


Figure 38 – A context diagram of the MIS with a focus on both KPIs for Efficiency

As can be seen from the figure there are no red elements, meaning the current situation at Ymor is ready for the forthcoming MIS. (Although of course the communication between the MIS and the two entities in the connection domain still needs to be set up when the MIS is developed and is not a reality yet.)

4.3.3. *Capacity utilization*

As is the case for the efficiency KPIs, the data needed for the capacity utilization KPI is already present at Ymor. The context diagram in Figure 39 shows this.

According to the figure, the data required by MIS for this KPI is stored in Ymor's Project ERP system as well as the 'HR MIS', which is the MIS used by the Human Resources (HR) department. The HR MIS at Ymor is called 'Cobra' and it contains information about the employees, including the amount of FTEs per employee and their function profile. Thus the data from Cobra is used to measure the total available capacity. The data from the Project ERP system (i.e. Navision) is required to measure the *required* capacity. When both factors are known, the capacity utilization rate can be calculated.

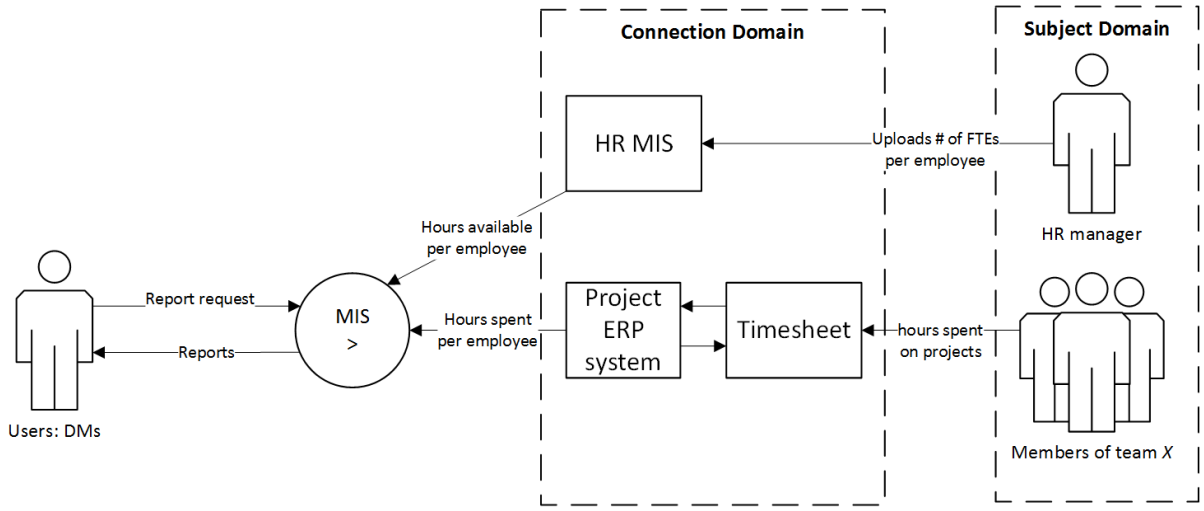


Figure 39 – A context diagram of the MIS with a focus on the KPI for Capacity Utilization

4.3.4. *Coverage of competencies*

The last context diagram involves the communications required to measure the KPI for the 'coverage of competencies'. If one looks at Figure 40, one can see that every part of the connection domain and the subject domain is colored red. This means that the required data for this KPI is not being measured by anyone or anything in Ymor yet. The context diagram in Figure 40 is therefore entirely prescriptive and does not describe the current situation.

As can be read from the figure, my advice is to take the competencies-survey in Appendix A and transform it into a web-page format. The SDMs and all other people that are required to fill it in can simply visit the web page, answer the questions and save the data directly into the MIS. The benefit of using a simple web page over a survey document sent over email is that the former can be made much more interactive and visually appealing.

Of course the downside is that such a web page and the required back-end needs to be developed and governed. Fortunately, Ymor is equipped with a Software Development department, which makes developing and governing such a simple website very inexpensive.

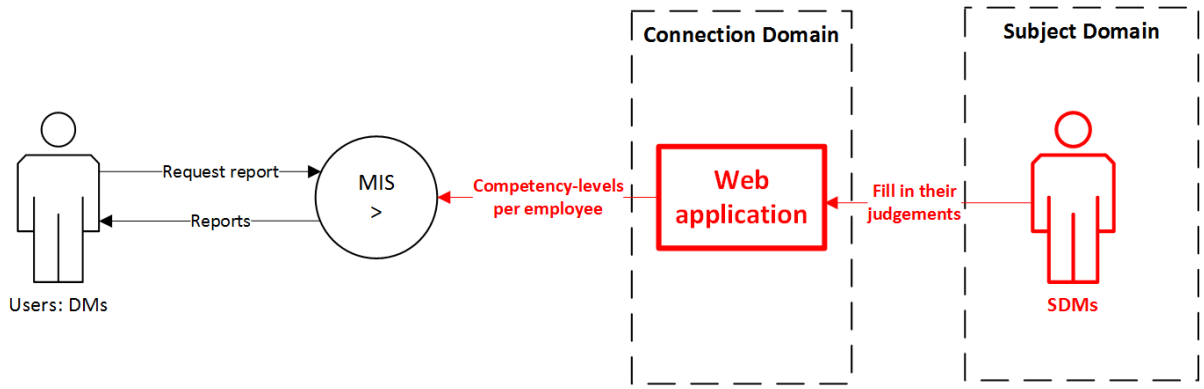


Figure 40 – A context diagram of the MIS with a focus on the KPI for Coverage of Competencies

4.4. Reports

The paragraphs 4.2 and 4.3 contain discussions concerning the data that the MIS requires, and where all this data should come from. Every context diagram in the previous paragraph shows the DMs as end-users that request and receive reports of the MIS. In this paragraph, I take a closer look at what kind of reports these will be. As with the previous paragraph, I discuss each performance variable separately and present the structure of the reporting on the associated KPI(s).

I have made an example for every KPI of how the reporting should be. These examples share the following properties:

- Every example is made with Microsoft Excel and uses ‘dummy data’, i.e. randomized sets of numbers within a certain boundary;
- The top of the columns in most examples indicates a time period, e.g. ‘t-8’ means ‘eight periods ago’ and ‘t’ means the current period. The examples do not go further back than ‘t-8’ but there is no reason why the real reports should not either.
- The examples are static images, while the MIS interface will likely be much more interactive. As such, the viewer of the examples is not able to see the data that the numbers in the tables are based on. For example, there is no way to tell from the first table in Figure 41 how many projects the average is based on. The final MIS interface however should be designed in such a way that the DMs can easily inspect the underlying data, i.e. the set of grades from which each average is calculated.
- In the final MIS interface, the norm should be adjustable and the colors of the cells should change accordingly.
- Every example is made in the table/matrix format, but the final MIS interface should have a graph option as well.

I will now move on to the description of the example reports.

Figure 41 shows the report example for the KPI for project success: the average project grade. The periods can be whatever suits the situation best, whether that is weeks, months or quarters. This will depend mostly on the frequency with which projects are completed and new grades are coming in. The bottom row shows the department average per period, while the rest of the rows show the team averages per period. As I have mentioned before, the management of Ymor has set the norm for this KPI on 8 or higher. Everything that meets this norm is colored green, while the rest is either yellow or red, depending on the amount of deviation from the norm. I have decided to use these colors to visualize the norm because they are also used this way in the Ymonitor software, which should make the reports feel more familiar to all the Delivery personnel.

Average client project grades

	t-8	*t-7*	*t-6*	*t-5*	*t-4*	*t-3*	*t-2*	*t-1*	*t*	*Moving average*
Team A	9	7,5	8	8,5	8,7	7	8,5	8,2	8	*8,2*
Team B	7,3	7,8	7,3	7,8	8,7	9	7,5	8,7	7,3	7,9
Team C	8	8,2	7,8	8,7	8,2	8,5	7	8,7	8	8,1
Team D	8,5	7	8,7	8,7	9	7,8	8	7,5	9	8,2
Team E	8,2	7,8	9	7	8,7	8,2	7	7,8	8	8,0
Team F	8,5	9	8	7,8	8,2	8,7	9	8	8,2	8,4
Team G	7,3	9	7,3	7,5	8,7	8	8,2	9	7,5	8,1
Team H	8,7	7,3	7,5	7,3	8,2	8	7,5	7	7,3	7,6
Average	8,1	8,1	7,7	8,0	8,6	7,8	8,4	8,3	7,9	8,1

Norms: ≥ 8 Good 7.5-8: Risk ≤ 7.5 Bad

Figure 41 – Example of MIS reports on the values of the KPI for 'project succes'

Figure 42 shows the example report of the first KPI for efficiency: the budget deviation. As the figure shows, values lower than 100% are good because they indicate that, on average, less hours have been spent on projects than was planned. Although we have to make a few assumptions (see paragraph 4.1), this is a sign of improved efficiency. Conversely, values above 100% are problematic because they indicate a decrease in efficiency. The rest of the report is structured in the same way as the previous one; the column to the right shows the moving average per (team), while the bottom row shows the department average per period.

Budget deviation: the average percentage of the budgeted hours used for projects

	t-8	*t-7*	*t-6*	*t-5*	*t-4*	*t-3*	*t-2*	*t-1*	*t*	*Moving average*
Team A	92%	96%	97%	97%	93%	80%	99%	98%	95%	94%
Team B	102%	109%	92%	83%	96%	105%	103%	104%	99%	99%
Team C	110%	104%	101%	110%	90%	92%	103%	101%	96%	101%
Team D	80%	102%	107%	95%	90%	97%	101%	104%	97%	97%
Team E	85%	80%	98%	96%	99%	95%	88%	109%	103%	95%
Team F	89%	97%	90%	108%	105%	102%	83%	105%	104%	98%
Team G	103%	97%	88%	98%	92%	97%	89%	100%	96%	96%
Team H	103%	99%	104%	97%	107%	98%	105%	98%	87%	100%
Average	96%	98%	97%	98%	97%	96%	96%	102%	97%	97%

Norms:	≤ 100%	Good
	100-105%	Risk
	> 105%	Bad

Figure 42 – Example of MIS reports on the 'budget deviation' KPI for 'efficiency'

The next figure, i.e. Figure 43, shows a reporting example of the second KPI for efficiency: the average actual hourly rate. The norm for this KPI is based on the target hourly rate of €100. Values under €100 are considered problematic. Values higher than €100 are good, up until the point of €200-€300. Values above €200 are met with suspicion because the DMs know form previous experiences that high values like that indicate other problems (see paragraph 4.1).

Average actual hourly rate during projects (in €)

	t-8	*t-7*	*t-6*	*t-5*	*t-4*	*t-3*	*t-2*	*t-1*	*t*	*Moving average*
Team A	118	88	102	90	82	102	94	98	102	97
Team B	99	106	104	94	106	107	90	98	110	102
Team C	88	99	104	123	93	116	112	104	108	105
Team D	100	97	95	123	85	89	105	108	100	100
Team E	93	107	99	99	104	100	133	87	105	103
Team F	129	115	138	98	112	113	88	128	111	115
Team G	106	96	116	86	100	108	99	93	100	100
Team H	107	93	114	94	81	96	72	94	96	94
Average	105	100	109	101	95	104	99	101	104	102

Norms	100-199	Good
	90-100	Risk
	≤ 90	Bad
	200-300	Risk
	> 300	Bad

Figure 43 – Example of MIS reports on the 'average actual hourly rate' KPI for 'efficiency'

The next example involves the KPI for capacity utilization, which is simply called the capacity utilization rate. The example reporting for this KPI is shown in Figure 44. The figure shows the two

different ways of calculating this KPI that I mentioned in paragraph 4.1. The first is based on the required and available number of FTEs per function profile. (Note that when I say 'required' here, I mean required for working on service delivery projects.) The second ignores function profiles and focuses on teams, and so is based on the number of required and available FTEs per team. The report shows both because together they can provide more information than apart. The two types of this KPI can be used for slightly different decisions. On the one hand, capacity based on function profiles is important when deciding on what job-experiences the DMs are looking for when hiring new people. On the other hand, the capacity KPI based on teams is important for decisions consider the compositions of the teams. Lastly, the top row shows the average department-wide capacity utilization rate, which is calculated based on all the required and available FTEs in the department.

High values of all these KPI variants indicate that there is too much project work. If this is true than the DMs basically have to options: increasing capacity or decreasing the work. However, the latter is not possible since Sales is keen on selling as much service delivery projects as possible to reach the long term growth goals of the company. In the other hand, low values of these KPIs can mean different things. The first possibility is that people have too little to do. This is however very unlikely since the mood at the department is often one of urgency and haste. The second possibility is that people are spending time on other things than the clients' projects. Examples of other activities are internal training or assisting the testing cycle of the Ymonitor software. The third possibility that I can see is that there are not enough people capable of doing the required work, because they have the wrong function profile for it. In these cases, low values of this KPI for one function profile must be accompanied with very high values for another.

Capacity utilization rate (%)

	t-8	*t-7*	*t-6*	*t-5*	*t-4*	*t-3*	*t-2*	*t-1*	*t*
Delivery department	80%	85%	80%	82%	71%	83%	82%	90%	80%
Performance Director	73%	87%	81%	76%	83%	89%	72%	80%	73%
Performance Architect	71%	75%	80%	88%	65%	82%	75%	69%	71%
Performance Consultant	92%	83%	81%	87%	79%	78%	89%	80%	92%
Ketenbewaking Specialist	70%	81%	88%	88%	82%	84%	75%	80%	70%
Ketenbewaking Specialist Trainee	80%	85%	77%	82%	71%	80%	82%	90%	80%
Team A	75%	85%	69%	88%	66%	85%	88%	76%	75%
Team B	92%	70%	83%	86%	91%	78%	81%	78%	92%
Team C	95%	80%	78%	77%	90%	90%	74%	76%	95%
Team D	83%	84%	72%	83%	94%	71%	73%	74%	83%
Team E	80%	79%	81%	73%	73%	80%	72%	81%	80%
Team F	88%	94%	70%	73%	73%	81%	90%	83%	88%
Team G	88%	87%	72%	86%	79%	82%	76%	74%	88%
Team H	84%	81%	82%	85%	87%	74%	72%	85%	84%

Norms						
75% - 85%	Good	70%-75%	Risk	≤ 70%	Bad
		85%-90%	Risk	≥ 90%	Bad

Figure 44 – Example of MIS reports on the 'capacity utilization rate' KPI

The final example report is shown in Figure 45 and is concerned with the competencies of the people at Delivery. As I have explained in the final section of paragraph 4.1, it is not possible to reasonably determine a norm for these measures at this point. However, it is hoped that the DMs will find that these numbers complement their own observations so that they can estimate whether the numbers are approximately right or not. For example, although there is no formally defined norm, it was clear to the DMs that there were not enough people with the appropriate level of competency with DynaTrace software. They were not able to give an exact number of people required with this competency, but they knew it was more than the current amount. I hope that the DMs can use the report in Figure 45 in that way until the steps described in paragraph 4.1 are taken to come up with formal norms.

Number of Delivery employees that possess a competency on a certain level					
	Novice	*Advanced beginner*	*Competent*	*Proficient*	*Expert*
C_1: Intake phase for performance test(s)	16	6	8	3	1
C_2: Technical realization of performance test(s)	16	9	2	2	0
C_3: Execution of tests and retests	13	4	1	2	1
C_4: Analysis of test results	13	8	2	2	3
C_5: Intake phase for end-user monitoring (active)	14	4	1	3	2
C_6: Technical realization of end-user monitoring (active)	12	2	7	5	1
C_7: Exploitation of end-user monitoring (active)	9	4	2	5	0
C_8: Governance of end-user monitoring (active)	19	7	4	2	0
C_9: Intake phase for end-user monitoring (passive)	14	9	2	2	1
C_{10}: Technical realization of end-user monitoring (passive)	9	7	4	3	1
C_{11}: Exploitation of end-user monitoring (passive)	16	5	7	3	3
C_{12}: Governance of end-user monitoring (passive)	14	4	3	2	1
C_{13}: Intake phase for diagnostic monitoring	10	6	4	4	3
C_{14}: Technical realization of diagnostic monitoring	12	5	1	4	1
C_{15}: Exploitation of diagnostic monitoring	11	2	8	4	0
C_{16}: Governance of diagnostic monitoring	18	5	8	2	1
C_{17}: Intake phase for system monitoring	9	9	6	5	0
C_{18}: Technical realization of system monitoring	16	8	1	3	1
C_{19}: Exploitation of system monitoring	10	4	4	4	1
C_{20}: Governance of system monitoring	19	2	8	5	2
C_{21}: Intake phase for desktop monitoring	16	9	7	3	1
C_{22}: Technical realization of desktop monitoring	15	5	2	3	1

C_{23}: Exploitation of desktop monitoring	10	7	3	5	1
C_{24}: Governance of desktop monitoring	15	8	3	3	1
C_{25}: Intake phase for configuration monitoring	16	5	3	5	0
C_{26}: Technical realization of configuration monitoring	14	6	7	2	1
C_{27}: Exploitation of configuration monitoring	10	2	3	5	1
C_{28}: Governance of configuration monitoring	16	9	8	5	1
C_{29}: database investigations	15	4	3	5	2
C_{30}: network investigations	12	7	5	4	1
C_{31}: client investigations					
	13	3	6	3	2
C_{32}: being able to use Ymonitor					
C_{33}: being able to use SilkPerformer	16	4	8	4	3
C_{34}: being able to use AutoHotkey	15	3	4	3	1
C_{35}: being able to use DynaTrace	10	4	6	5	1
C_{36}: being able to use Evolven	14	9	4	1	1
C_{37}: being able to use DCRum	12	2	5	1	1
C_{38}: being able to use NextThink	10	8	6	1	1
C_{39}: being able to use Splunk	12	7	4	3	2
C_{40}: being able to use WUG	14	2	1	3	1
C_{41}: being able to use Precise	14	5	6	1	0
C_{42}: being able to use Exinda	10	9	6	2	0
C_{43}: being able to use Foglight	11	7	4	2	1
C_{44}: being able to use Jmeter	15	4	4	1	1
C_{45}: being able to use HP Loadrunner	9	9	1	3	1
Norms	?	Good	?	Risk	?	Bad

Figure 45 – Example of MIS reports on the measures of 'Coverage of Competencies'

5. Recommendations

With the design of the MIS specified in chapter four, it is time to discuss a few recommendations. The recommendations of this chapter involve either the *implementation* of the MIS design, the *use* of the implemented MIS, or future *expansion* of the MIS' functionality.

5.1. Implementation of the MIS

When it comes to the implementation of the MIS, the DMs have two options. The first option is to approach third-party software providers to see if Ymor can purchase a MIS from them that meets the requirements specified in chapters three and four. The other option is to let Ymor's own Software Development department built the MIS using the design specification in the previous chapter.

However, before implementing the system it would be good if managers from Sales, DMs and SDMs came together to go through another iteration of the competency-identification method that I discussed in chapter 3. This is important because these people all have to agree on the completeness of the list of relevant competencies. It is possible, even likely, that the list that I produced is incomplete, as feedback from the SDMs and some other senior Delivery employees indicated. Furthermore, when I applied the method I skipped steps 3 and 4 because of limits to time for this project.

In addition to completing the list of competencies, the DMs and SDMs need to come up with norms for the level of coverage of each of these competencies in Delivery. The question that needs answering is: how many people that possess a certain level of expertise are needed per competency? Because competencies are linked to service characteristics and technologies, it is first necessary that it becomes known for each project what its service characteristics are and what technologies will be applied. This way it will be known exactly which of the competencies are needed per project. This is the *qualitative* aspect of defining the norm. The next step is *quantitative*, and involves defining the amount of people and the minimum level of expertise per competency during a project. Only when it is clear what competencies are needed per project as well as how many of them, can a norm be calculated.

5.2. Use of the MIS

The next two recommendations involve the use of the MIS after it is implemented. Both are about the measuring competencies of Delivery employees as a performance indicator.

Firstly, I would like to repeat certain advice that is often given in HRM literature (e.g. Maylett & DecisionWise, 2009). The advice is to ensure that employees know that measuring their competencies is done for development purpose only, and not related to any rewarding schemes. Otherwise the resulting data is likely to be unreliable because people generally tend to let their interest and that of others influence their answers.

The second recommendation for the use of the MIS is to make sure that the list of relevant competencies is updated regularly (e.g. once a year). The reason for this is that the list of relevant competencies is actually dynamic. Because of changing markets, technologies and even innovations by Ymor itself, some competencies can become irrelevant or new ones might appear. If the list of competencies is not updated properloy, then there is a risk that important competencies will be overlooked.

5.3. Possible expansion of the MIS

At the end of chapter 3 I concluded that the MIS will be of a Lockean type, and thus will only be responsible for providing information on 'the state of the world', i.e. {S}. In this case this means it measures the value of four performance variables. Of course, as we have seen in chapter 2, to effectively make decisions the decision makers need more information than that. They need to know the actions that they can choose from {A}, as well as their (likely) outcomes {O} and the utility of those outcomes {U}.

The Lockean MIS designed during this research project does not provide information on these last three aspects of decisions. The assumption is that these three sets are still known by the DMs. While this could be (mostly) true, it seems likely that this will change as their department will continue to grow in the coming years. Decisions will become more complex and the range of possible actions and outcomes is expected to increase. But even if this would not be the case, one could argue that having all this knowledge locked with two DMs poses a risk to the company.

In any case, I recommend to expand the MIS in the near future with a component that provides the DMs with information on those three aspects of decisions (i.e. {A}, {O} and {U}). Adding such a component would turn the MIS into a *Decision Support System* (DSS). To make such an expansion possible, some kind of model will be needed that is capable of taking the current state of the world and simulating the outcomes of different actions in order to determine the one with the most utility for the decision maker.

Designing such a model is a complex task and requires further research. The most difficult part of the model will probably defining the relations between actions and possible outcomes. After all, the Delivery department exists of knowledge workers and its activities and output are not narrowly defined.

6. Discussion, conclusion and future research

6.1. Looking back

The goal of this research project was twofold: to design a MIS for Ymor using a new design method, and by doing so to learn about the usefulness of said method. The two research questions that I formulated in chapter 1 reflect this duplicity. To clarify, the research questions were:

Research question 1) *What are the functional requirements of the MIS for the Delivery department at Ymor?*

Research question 2) *To what extent is this design method useful for the design of MIS in other contexts*?

The purpose of the first question was to enable the design of a MIS for Ymor (or more specifically, for the Delivery department) and corresponds to the first goal of this research. It was answered at the end of chapter 3. The purpose of the second question is to learn about the design method that I developed and used for this project. To find an answer to this question is the goal of this paragraph.

6.1.1. *Core problem and design method*

But first let us start with the core problem that motived this research project. The problem was that the DMs had insufficient information about the performance of their Delivery department. They desired a new MIS that would provide the information that they were missing, which is why designing such a system was this project's first goal.

The question then became how to approach such a design problem. As I describe in chapter 2, I based my approach on the work of Mason and Mitroff (1973). In their article, they explain what the five key variables are that according to them make up a MIS:

1. The **psychological type** of the users of the system;
2. The **method of evidence generation** (or: the nature of the guarantor of evidence);
3. The **organizational context** within which the MIS operates and problems occur (see below)
4. The **class of problems** that the system is supposed to provide information about
5. The **mode of presentation** of evidence (i.e. information) by the system.

To determine the requirements of the MIS means to decide on the most suitable values for these five variables. Of course there are many other possible requirements for a MIS. A common example are user-requirements (e.g. 'the MIS has to respond quickly' or 'the use of the MIS should be easy to learn'). But I consider these kinds of requirements to be details that do not involve the essence of the MIS. These should therefore be dealt with during a later stage of MIS design or implementation. What is needed first is a sketch: a general outline of the main features or parts of the system under design. Deciding on the values of the five key variables listed above is the equivalent of making such a sketch.

My method is a way to find the appropriate values of the five key variables in situations where a new MIS is considered desirable. Ideally the five variables should be considered as a coherent whole. However due to limits to our analytic abilities we are forced to look at them one at the time and decide

on their values in a certain sequence. As I have described at the start of chapter three, there are two possible sequences. One either starts with defining the psychological type of the users (1), followed by choosing the appropriate mode of presentation (5) and method of evidence generation (2), after which the organizational context (3) is mapped and finally the class of the decision problems is determined (4). Thus the first possible sequence is 1-5-2-3-4. The second possible sequence starts with the organizational context, followed by the classification of the decision problems within that context. Next the method of evidence generation that matches these class of problems best is chosen. Only then is the psychological type of the users determined, followed by the choice for a mode of presentation that suits this type. So the second sequence of analyzing the five variables is 3-4-2-1-5.

6.1.2. *Review of the design process and its usefulness*

For the situation at Ymor I choose to follow the first sequence, which means I started with analyzing the psychological type of the end-users (i.e. the DMs). The reason for this is that the DMs were also the clients of this project. From the way they formulated their wishes and expectations with regards to the MIS I concluded that they are of the 'Thinking'-'Sensing' type. The validity of this classification can be questioned since I did not use a scientifically tested method for this. As I explained in chapter three, the classification is mostly based on my observations during my months at Ymor.

However, what is more important at this point is the question if determining the psychological type is useful for the design of a MIS in general, even if scientifically tested methods are applied for this. The answer to this questions seems positive; a MIS serves one or more end-users en therefore should match the way they perceive and evaluate information. An exception to this answer might be situations where the group of end-users is very heterogeneous, meaning a lot of the end-users are of a different psychological type. Examples of such situations are MIS design projects for large corporations. In these cases, this variable is unlikely to play a defining role during the design process of the MIS, since all types are relevant and the MIS design cannot focus on a single one.

The next key variable for which I have attempted to determine the most suitable value, was the mode of presentation of information used by the MIS. This was fairly straight-forward: the DMs explicitly requested that the future MIS would be able to produce management reports. In other words, they had already set the value of this second key design variable of the MIS.

Now, is the consideration of this design variable a useful addition to a MIS design method? I believe that it is, because Mason and Mitroff remind us that more personal modes of presentation than the traditional computer generated reports are possible, and these modes are more suitable with end-users of certain other psychological types. Thus the usefulness of this considering this design variable is linked with the usefulness of defining the psychological type of the users.

The next key design variable of a MIS that I tried to determine the appropriate value for, was the evidence generation method. I found that the Lockean evidence generation method was the most suitable of the five alternatives for the forthcoming MIS at Ymor. I based this decision on the previously determined psychological type and the preferred mode of presentation, and the fact that the purpose of the MIS was to provide information on the current state of the world (i.e. the Delivery department's performance). Of course it is entirely possible to break a MIS up in sub-systems and assign a different evidence generation method to each of them. This is essentially what I did when I concluded in chapter three that some aspects of Delivery's performance are better left measured by a MIS based on the

Kantian inquiring system. Unfortunately, I had to leave those aspects out of the scope of this project due to time constraints.

Prompting an explicit choice for one of Churchman's five evidence generation methods (or 'Inquiring Systems' as he calls them) is the most important way in which my design method distinguishes itself from other methods. It is also one of the two reasons why this method deserves to be labelled as 'a systems approach', the other being its use of the Viable Systems Model (but more on that later). Although Churchman implies that the five evidence generation methods go from primitive forms to the more advanced (Churchman, 1971, p. viii), the intention of my design method is not to promote Kantian, Hegelian or Singerian inquiring systems. Instead I believe that the properties of a situation mostly decide which is more appropriate. But by making the choice of an evidence generation method explicit, my design method forces the designer to acknowledge the existence of different perspectives on information, truth and the meaning of inquiry. It reminds the designer of Churchman's warning that every MIS has built-in sources of deception, and thus stimulates the designer to look for them (or at least to be wary of them). Many other MIS design methods pretend that these sources of deception do not exist, thus promoting the design of a MIS that offers possibly deceiving information and thus a false sense of certainty for the managers that use them.

So, does this feature of my design method make it more useful than other MIS design methods? I fear not. In fact, it might even be considered less useful, if the usefulness of a MIS design method is defined as the ability to simplify the process of designing the MIS. The purpose of this third design variable is to ask more questions, and not to provide answers. Thus it makes my design method more complex without the guarantee that the resulting MIS will be better. But I hold that what we get in return is a better chance of becoming aware of sources of deception in our MIS design, after which we can address them and enter a discussion about their meaning and implications for management practice.

The next design variable that I considered was the organizational context of the MIS under design. This is where my design method deviates from the work by Mason and Mitroff. Instead of the three possible values that they list (i.e. Strategic Context, Management, and Operational Context), I used the Viable System Model (or VSM). By looking at Ymor through the lens of the VSM, I found that the organizational context of the MIS is 'System 3' of the viable system that is the Delivery department.

Again we must ask the question: is it useful to determine the organizational context of a MIS by means of the VSM? Does it improve our design process? Firstly, defining the organizational context of a MIS is closely related to defining the problems that it will provide information about. These problems are the fifth and last key variable and are therefore discussed below, but it seems safe to state at this point that determining those is quite useful – even vital – for any MIS design process. So presuming it is necessary to define the organizational context of a MIS as a means of setting the scope, the question becomes: is the VSM useful? Looking back at the design process during this project, I conclude that it is. It is particularly useful for small and medium enterprises (SMEs) such as Ymor, because the five subsystems of the VSM can be recognized in organizations in any field and of any size. The common lack of formal organizational charts at SMEs is thus not a large issue anymore when the VSM is used. The VSM provides very good focus to the MIS design project without losing touch with the whole.

The fifth and last key design variable of a MIS is the class of decision problems that it is supposed to provide information about. As I explain in chapter two, Mason and Mitroff distinguish between structured and unstructured decision problems. The relevant decision problems of the DMs all involve improving or maintaining the performance of the Delivery department, but because 'performance' in this case was ill-defined, the decision problems were of the unstructured type. This was one of the obstacles during the design process, because the rest of the values of the other four design variables are most suitable for *structured* problems, and quite unsuitable for *unstructured* ones. That is why I decided to start a second analysis phase using the Balanced Scorecard (BSC) model, to define 'performance' into a list of performance variables. The result was a list of seven performance variables. Unfortunately only four were suitable for the upcoming MIS; two were still too unstructured and analyzing them further would cost too much time, and one other performance variable (employee satisfaction) fell outside the scope of this project because a third party was already hired to measure it.

Now, I believe it is self-evident that determining the type of decision problems, i.e. whether they're structured or unstructured, is useful during the design of the MIS. After all, the purpose of the MIS is to provide users information about these decision problems. Looking at the decision problems this way is thus the first step to determining the information needs that the MIS is supposed to meet.

One might say I merely "carved off" a piece of the meaning of 'organizational performance' by creating that list of performance variables. Perhaps I should have respected the wickedness and complexity of it by changing the evidence generation method to one that is more suitable with wicked problems, such as the Kantian or Hegelian variants. That was definitely a possibility, although it probably would not have made the DMs and other people at Ymor very pleased, because they expected a Lockean MIS from the start and the type of evidence provided by for example Hegelian systems is entirely different.

Finally, we might ask if the BSC model was a useful addition to the design method as a tool for defining the performance of an organization (in this case the Delivery department). Although definitely a helpful model, there is no reason why another model would not suffice. In fact, other models will be necessary, depending on the situation. The BSC model was useful because it is a model of performance of companies and in this case 'performance' needed to be defined. If the design issue was not defining 'performance' but something entirely different, then of course some other model would be necessary. I do suspect however that the BSC model is useful in many other MIS design projects at fast-growing SMEs, since management maturity models suggest that they often lack a formal definition of their key performance variables.

6.1.3. *Conclusion*

The goal of this part of this chapter was to find an answer to the following research question:

Research question 2) *To what extent is this design method useful for the design of MIS in other contexts*?

As the previous discussion suggests, my answer to this question is rather positive. I believe that the design method that I have applied during this project is promising, as it increases our understanding of both the processes and goals of MIS design. Its usefulness is not limited to the situation at Ymor; I expect that it can be applied in many different situations where a new MIS is considered desirable.

This is mainly because of the conceptual strength of the models that it is based on, such as the VSM, the MBTI and Churchman's five inquiring systems. The VSM in particular enables the use of this design method at either large or small organizations. Compared to many other MIS design approaches, another strong point of this design method is that it considers the five key variables of a MIS as a whole, and so ensures coherence between them. What is especially valuable is that, of the five variables, the psychological type of the users and the evidence generation method are two. The first is a valuable addition because it opens up the design task to the social/psychological domain, whereas many MIS design approaches are strictly technical in nature. The explicit discussion of the evidence generation method of the MIS, on the other hand, is in some ways not very useful at all. It links the MIS design task to the philosophical domain and its main function is to raises more questions, instead of providing answers. Nevertheless, I believe it is a necessary addition to any MIS design approach because the questions point to the sources of deception that as Churchman showed us are inevitably built into every MIS that we design.

6.2. Looking forward

I would like to make two suggestions for further research. Firstly, to test this MIS design method at different organizations in entirely different situations, for example a non-profit organization that needs a new MIS but exists of users with different psychological types and who prefer other modes of presentations of information. This kind of research might inform us about the true usefulness of the design method.

The other possible direction for future research is the testing and validation of the MIS implementations that result from this design method. After all, the design of a MIS is meant to solve a certain (information) problem. These kind of research projects should make clear whether the designed systems are in fact meeting all the information needs. Through such investigations, we might learn about the sources of deception that have inevitably found their way in this MIS design method.

References

Adams, C., & Neely, A. (2000). The performance prism to boost M&A success. *Measuring Business Excellence*, *4*(3), 19–23. doi:10.1108/13683040010377818

Andersen, H. V., Lawrie, G., & Shulver, M. (2000). The Balanced Scorecard vs. the EFQM Business Excellence Model–which is the better strategic management tool. *2GC Active Management*, 1–15.

Bassett, G. A. (1992). *Operations Management for Service Industries: Competing in the Service Era*. Greenwood Publishing Group.

Bayne, R. (1995). *The Myers-Briggs Type Indicator: A Critical Review and Practical Guide*. Nelson Thornes.

Beard, A. N., & Santos-Reyes, J. (1999). Creating a fire safety management system for offshore facilities. *Facilities*, *17*(9/10), 352–362.

Beer, S. (1984). The Viable System Model: Its Provenance, Development, Methodology and Pathology. *The Journal of the Operational Research Society*, *35*(1), 7. doi:10.2307/2581927

Beer, S. (1994). *Platform for change: a message from Stafford Beer.* Chichester; New York: J. Wiley.

Biazzo, S., & Garengo, P. (2012). The Balanced Scorecard for SMEs: A Circular Approach. In *Performance Measurement with the Balanced Scorecard* (pp. 23–38). Springer.

Bittencourt, V. S. (2013, November). *Measuring the Level of Capacity Utilisation in Non-Manufacturing Sectors*. Workshop presented at the ECFIN 2013 Business and Consumer Surveys Workshop, Brussels. Retrieved from http://ec.europa.eu/economy_finance/db_indicators/surveys/documents/workshops/2013/workshop_nov_15_brussels_viviane_seda_bittencourt.pdf

Campion, M. A., Fink, A. A., Ruggeberg, B. J., Carr, L., Phillips, G. M., & Odman, R. B. (2011). Doing competencies well: Best practices in competency modeling. *Personnel Psychology*, *64*(1), 225–262.

Churchman, C. W. (1971). *The Design of Inquiring Systems: Basic Concepts of Systems and Organization*. Basic Books.

Corsten, H., & Stuhlmann, S. (1998). Capacity management in service organisations. *Technovation*, *18*(3), 163–178. doi:10.1016/S0166-4972(97)90121-8

Courtney, J. F. (2001). Decision making and knowledge management in inquiring organizations: toward a new decision-making paradigm for DSS. *Decision Support Systems*, *31*(1), 17–38.

Courtney, J. F., Haynes, J. D., & Paradice, D. B. (2005). *Inquiring Organizations: Moving from Knowledge Management to Wisdom*. IGI Global.

De Toni, A. F., Fornasier, A., Montagner, M., & Nonino, F. (2007). A performance measurement system for facility management: the case study of a medical service authority. *International Journal of Productivity and Performance Management*, *56*(5/6), 417–435.

De Vries, E. J. (2006). Innovation in services in networks of organizations and in the distribution of services. *Research Policy*, *35*(7), 1037–1051. doi:10.1016/j.respol.2006.05.006

Dreyfus, S. E., & Dreyfus, H. L. (1980). *A five-stage model of the mental activities involved in directed skill acquisition*. DTIC Document.

EFQM. (2012a). EFQM Excellence Model. Retrieved July 4, 2014, from http://www.efqm.org/efqm-model/model-criteria

EFQM. (2012b). Model Criteria. Retrieved July 22, 2014, from http://www.efqm.org/efqm-model/model-criteria

Espejo, R., & Gill, A. (1997). The viable system model as a framework for understanding organizations. *Phrontis Limited & SYNCHO Limited*. Retrieved from http://www.phrontis.com/syncho/V1.pdf

Espejo, R., & Reyes, A. (2011). *Organizational Systems*. Berlin, Heidelberg: Springer Berlin Heidelberg. Retrieved from http://link.springer.com/10.1007/978-3-642-19109-1

Espinosa, A., & Walker, J. (2013). Complexity management in practice: A Viable System Model intervention in an Irish eco-community. *European Journal of Operational Research*, *225*(1), 118–129. doi:10.1016/j.ejor.2012.09.015

Gallouj, F., & Savona, M. (2009). Innovation in services: a review of the debate and a research agenda. *Journal of Evolutionary Economics*, *19*(2), 149–172. doi:10.1007/s00191-008-0126-4

Gallouj, F., & Weinstein, O. (1997). Innovation in services. *Research Policy*, *26*(4–5), 537–556. doi:10.1016/S0048-7333(97)00030-9

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly: Management Information Systems*, *28*(1), 75–105.

Hidalgo, A., & Albors, J. (2008). Innovation management techniques and tools: a review from theory and practice. *R&d Management*, *38*(2), 113–127.

Hudson, M., Lean, J., & Smart, P. A. (2001). Improving control through effective performance measurement in SMEs. *Production Planning & Control*, *12*(8), 804–813. doi:10.1080/09537280110061557

Kaplan, R. S. (2008a). Conceptual Foundations of the Balanced Scorecard. In A. G. H. and M. D. S. Christopher S. Chapman (Ed.), *Handbooks of Management Accounting Research* (Vol. Volume 3, pp. 1253–1269). Elsevier.

Kaplan, R. S. (2008b). Conceptual foundations of the balanced scorecard. *Handbooks of Management Accounting Research*, *3*, 1253–1269.

Kaplan, R. S., & Norton, D. P. (1992). The Balanced Scorecard--Measures That Drive Performance. *Harvard Business Review*, *70*(1), 71–79.

Kaplan, R. S., & Norton, D. P. (1996). Using the balanced scorecard as a strategic management system. *Harvard Business Review*, *74*(1), 75–85.

Kaplan, R. S., & Norton, D. P. (1998). Putting the balanced scorecard to work. *The Economic Impact of Knowledge*, 315–24.

Kaplan, R. S., & Norton, D. P. (2000). Having Trouble with Your Strategy? Then Map It. *Harvard Business Review*, *78*(5), 167–176.

Kaplan, R. S., & Norton, D. P. (2001). Transforming the balanced scorecard from performance measurement to strategic management: Part I. *Accounting Horizons*, *15*(1), 87–104.

Leonard, A. (2009). The Viable System Model and Its Application to Complex Organizations. *Systemic Practice and Action Research*, *22*(4), 223–233. doi:10.1007/s11213-009-9126-z

Marr, B., & Schiuma, G. (2003). Business performance measurement–past, present and future. *Management Decision*, *41*(8), 680–687.

Mason, R. O., & Mitroff, I. I. (1973). A program for research on management information systems. *Management Science*, *19*(5), 475–487.

Maylett, T., & DecisionWise, C. E. O. (2009). 360-Degree Feedback Revisited: The Transition From Development to Appraisal.

Myers, I. B. (1962). The Myers-Briggs Type Indicator: Manual (1962).

Neely, A., Adams, C., & Crowe, P. (2001). The performance prism in practice. *Measuring Business Excellence*, *5*(2), 6–13.

Nørreklit, H. (2003). The balanced scorecard: what is the score? A rhetorical analysis of the balanced scorecard. *Accounting, Organizations and Society*, *28*(6), 591–619.

Ortt, J. R., & van der Duin, P. A. (2008). The evolution of innovation management towards contextual innovation. *European Journal of Innovation Management*, *11*(4), 522–538. doi:10.1108/14601060810911147

Pérez Ríos, J. (2012). *Design and Diagnosis for Sustainable Organizations*. Berlin, Heidelberg: Springer Berlin Heidelberg.

Proudfoot, M., & Lacey, A. R. (2009). *The Routledge dictionary of philosophy*. Routledge.

Pycraft, M., Singh, H., Phihlela, K., Slack, N., Chambers, S., Chambers, C., … Johnston, R. (2000). *Operations Management*. Pearson South Africa.

Ramage, M., & Shipp, K. (2009). *Systems Thinkers*. Springer.

Rigby, D., & Bilodeau, B. (2013). *Management Tools & Trends 2013* (p. 14). Bain & Company. Retrieved from http://bain.com/Images/BAIN_BRIEF_Management_Tools_%26_Trends_2013.pdf

Rosenhead, J. (2006). IFORS' Operational Research Hall of Fame Stafford Beer. *International Transactions in Operational Research*, *13*(6), 577–581. doi:10.1111/j.1475-3995.2006.00565.x

Rusjan, B. (2005). Usefulness of the EFQM excellence model: Theoretical explanation of some conceptual and methodological issues. *Total Quality Management & Business Excellence*, *16*(3), 363–380. doi:10.1080/14783360500053972

Schwaninger, M. (2006). Design for viable organizations: the diagnostic power of the viable system model. *Kybernetes: The International Journal of Systems & Cybernetics*, *35*(7-8), 955–966.

Taticchi, P., Tonelli, F., & Cagnazzo, L. (2010). Performance measurement and management: A literature review and a research agenda. *Measuring Business Excellence*, *14*(1), 4–18. doi:10.1108/13683041011027418

US Federal Reserve. (2014). *Industrial Production and Capacity Utilization* (No. G.17). US Federal Reserve. Retrieved from http://www.federalreserve.gov/releases/g17/Current/default.htm

Vargo, S. L., & Lusch, R. F. (2004). Evolving to a new dominant logic for marketing. *Journal of Marketing*, *68*(1), 1–17.

Wieringa, R. J. (2003). *Design methods for reactive systems: Yourdon, statemate, and the UML*. Elsevier.

Wijnhoven, F. (2012). *Information services design: a design science approach for sustainable knowledge*. New York: Routledge.

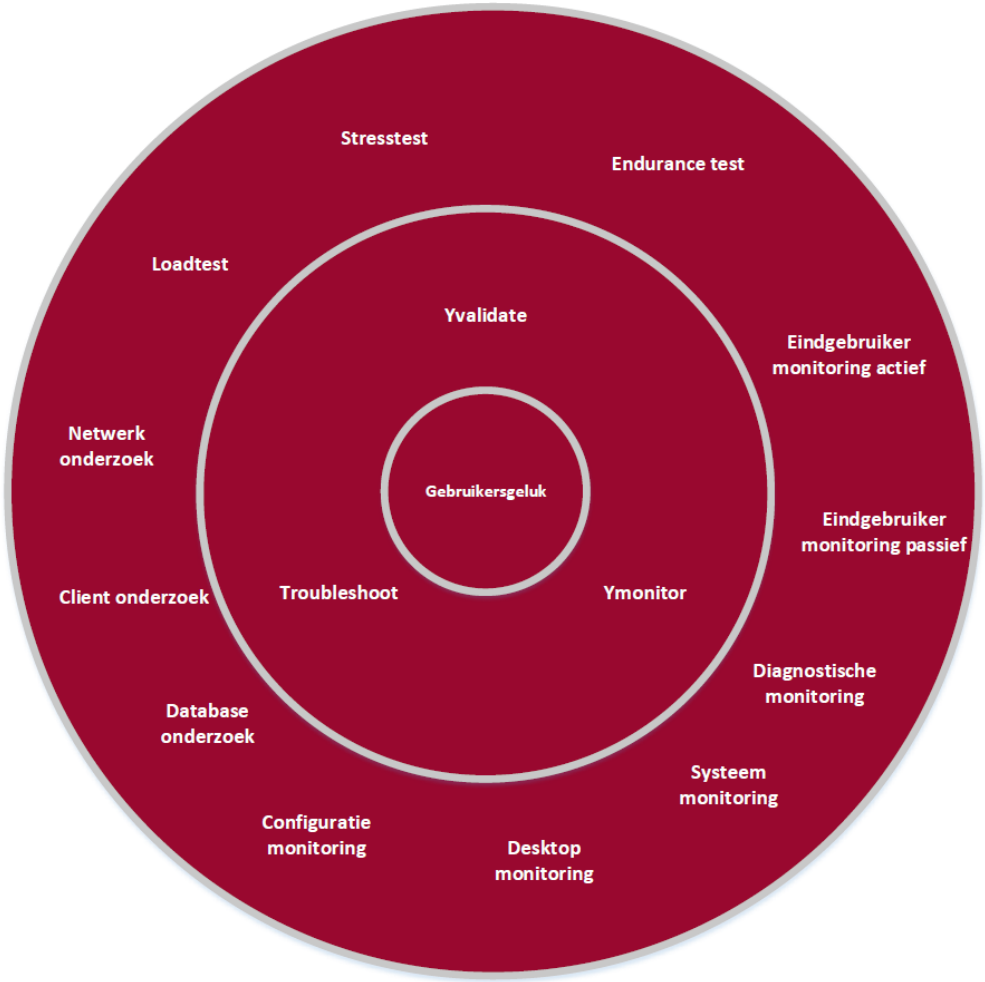
Appendix A – Competentie beoordelingssurvey

Introductie

Er zijn een aantal activiteiten geïdentificeerd die ten grondslag liggen aan de diensten van Ymor. Voorbeelden van deze activiteiten zijn *performance testen* en *diagnostische monitoring*. Deze en andere activiteiten zijn terug te vinden in figuur 1. Dit is de dienstencirkel die door TAM is gemaakt.

Voor het uitvoeren van deze activiteiten worden daarnaast een aantal technologieën bij Ymor toegepast. Op de volgende pagina staat een lijst van deze technologieën.

Met deze survey hopen we in kaart te kunnen brengen hoe goed alle teamleden ongeveer zijn in het uitvoeren van de activiteiten en het gebruik van de technologieën. Het is niet de bedoeling van deze survey om te bepalen of een persoon *goed genoeg* is of niet. Het gaat puur om het meten van het huidige vaardigheidsniveau van de teamleden zodat er een overzicht kan worden gemaakt van de afdeling als geheel. Dit kan nuttig zijn voor bijvoorbeeld recruitment of het plannen van trainingen.



Figuur 1 – dienstencirkel van Ymor, gemaakt door TAM

Technologieën (software)
T_1: Ymonitor
T_2: Silkperfomer
T_3: AutoHotkey
T_4: DynaTrace
T_5: Evolven
T_6: DCRum
T_7: NextThink
T_8: Splunk
T_9: WUG
T_{10}: Precise
T_{11}: Exinda
T_{12}: Foglight
T_{13}: Jmeter
T_{14}: HP loadrunner

Figuur 2 – Een lijst van software instrumenten in Delivery worden gebruikt

Instructies

Ik wil je vragen om voor elk van je teamleden deze survey apart in te vullen. Let hierbij alsjeblieft op de volgende punten:

- Bij het invullen van de survey dien je voor elke activiteit in te schatten wat het vaardigheidsniveau is van het betreffende teamlid. Ik maak onderscheid tussen vijf vaardigheidsniveaus, te weten:

  1. Beginner
  2. Gevorderde beginner
  3. Bekwaam
  4. Deskundig
  5. Expert

  De laatste twee pagina's van dit document bevatten elk een tabel waarin de niveaus worden beschreven aan de hand van verschillende karakteristieken, zoals hun kennisniveau en de mate van autonomie. Ik hoop dat deze informatie de verschillen tussen de vaardigheidsniveaus duidelijk genoeg maken. Bekijk deze dus alsjeblieft goed voor dat je aan de vragen begint.

- Het is mogelijk (en zelfs waarschijnlijk) dat iemand op verschillende vaardigheidsniveaus zit met verschillende activiteiten. Iemand is bijvoorbeeld een beginner met activiteit *x* maar van deskundig niveau met activiteit *y*.

- Ingevulde surveys ontvang ik graag via mijn email: **[EMAIL]**

- Ten slotte wil ik je vragen om deze survey zelf in te vullen en niet samen met de betreffende teamleden. Niemand weet nog van deze survey en dat houden we graag nog even zo omdat

er het plan is om de teamleden ook elkaar en zichzelf te laten scoren. We vermoeden dat dit beter werkt als men geen tijd krijgt om met elkaar over de survey te praten.

START SURVEY

Deze beoordelingssurvey is ingevuld door:

De medewerker die met deze survey wordt beoordeeld heet:

Van welk team is deze medewerker lid?

Ruimte voor eventuele feedback op deze survey:

Survey items

Op welk vaardigheidsniveau zou jij het betreffende teamlid inschatten als het gaat om de volgende activiteiten?

C_1: intake fase voor performance testen

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_2: technische realisatie van performance testen

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_3: testen en hertesten

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_4: analyse van testresultaten

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

Toelichting:

Op welk vaardigheidsniveau zou jij het betreffende teamlid inschatten als het gaat om de volgende activiteiten?

C_5: intake fase voor eindgebruikers monitoring (actief)

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_6: technische realisatie van eindgebruikers monitoring (actief)

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_7: exploitatie van eindgebruikers monitoring (actief)

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_8: de nazorg voor eindgebruikers monitoring (actief)

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

Toelichting:

Op welk vaardigheidsniveau zou jij het betreffende teamlid inschatten als het gaat om de volgende activiteiten?

C_9: intake fase voor eindgebruikers monitoring (passief)

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{10}: technische realisatie van eindgebruikers monitoring (passief)

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{11}: exploitatie van eindgebruikers monitoring (passief)

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{12}: de nazorg voor eindgebruikers monitoring (passief)

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

Toelichting:

Op welk vaardigheidsniveau zou jij het betreffende teamlid inschatten als het gaat om de volgende activiteiten?

C_{13}: intake fase voor diagnostische monitoring

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{14}: technische realisatie van diagnostische monitoring

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{15}: exploitatie van diagnostische monitoring

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{16}: de nazorg voor diagnostische monitoring

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

Toelichting:

Op welk vaardigheidsniveau zou jij het betreffende teamlid inschatten als het gaat om de volgende activiteiten?

C_{17}: intake fase voor systeem monitoring

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{18}: technische realisatie van systeem monitoring

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{19}: exploitatie van systeem monitoring

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{20}: de nazorg voor systeem monitoring

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

Toelichting:

Op welk vaardigheidsniveau zou jij het betreffende teamlid inschatten als het gaat om de volgende activiteiten?

C_{21}: intake fase voor desktop monitoring

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{22}: technische realisatie van desktop monitoring

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{23}: exploitatie van desktop monitoring

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{24}: de nazorg voor desktop monitoring

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

Toelichting:

Op welk vaardigheidsniveau zou jij het betreffende teamlid inschatten als het gaat om de volgende activiteiten?

C_{25}: intake fase voor configuratie monitoring

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{26}: technische realisatie van configuratie monitoring

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{27}: exploitatie van configuratie monitoring

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{28}: de nazorg voor configuratie monitoring

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

Toelichting:

Op welk vaardigheidsniveau zou jij het betreffende teamlid inschatten als het gaat om de volgende activiteiten?

C_{29}: database onderzoek

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{30}: netwerk onderzoek

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{31}: client onderzoek

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

Toelichting:

Op welk vaardigheidsniveau zou jij het betreffende teamlid inschatten als het gaat om de volgende technologieën?

C_{32}: Ymonitor

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{33}: SilkPerformer

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{34}: AutoHotkey

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{35}: DynaTrace

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{36}: Evolven

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{37}: DCRum

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{38}: NextThink

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{39}: Splunk

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{40}: WUG

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{41}: Precise

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{42}: Exinda

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{43}: Foglight

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{44}: Jmeter

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

C_{45}: HP Loadrunner

○ Beginner ○ Gevorderde Beginner ○ Bekwaam ○ Deskundig ○ Expert

Toelichting:

TABEL 1 – Beschrijving van de vijf vaardigheidsniveaus

<table>
<tr><th>Niveau</th><th>Graad</th><th>Karakteristieken</th><th>Hoe met kennis wordt omgegaan</th><th>Herkenning van relevantie</th><th>Hoe situaties worden beoordeeld</th><th>Besluitvorming</th></tr>
<tr><td>1</td><td>Beginner</td><td>• Houdt zich op rigide wijze aan regels en stappenplannen
• Nemen weinig waar
• Maken weinig onderscheidingen</td><td>Zonder referentie naar de context</td><td rowspan="2">Niet aanwezig</td><td rowspan="3">Analytisch</td><td rowspan="4">Rationeel</td></tr>
<tr><td>2</td><td>Gevorderde beginner</td><td>• Iets beter waarnemend vermogen
• Gebruikt richtlijnen voor het reageren op waargenomen variaties in de context
• Aan alle variaties in de context wordt hetzelfde belang gehecht (geen prioriteiten)</td><td rowspan="4">Met referentie naar context (i.e. context-gevoelig)</td></tr>
<tr><td>3</td><td>Bekwaam</td><td>• Kan werken ondanks drukte
• Ziet relatie tussen zijn of haar acties en de langere termijn doelen en/of plannen
• Bewust, opzettelijk plannen
• Hanteert standaard routines en procedures</td><td rowspan="3">Aanwezig</td></tr>
<tr><td>4</td><td>Deskundig</td><td>• Heeft overzicht over de context en verliest zich niet in details of grote hoeveelheden informatie
• Ziet wat belangrijk is in de situatie
• Herkent afwijkingen van normale patronen
• Beslissingen maken kost minder moeite
• Gebruikt basis principes als richtlijnen, waarvan de betekenis varieert per situatie</td><td rowspan="2">Holistisch</td></tr>
<tr><td>5</td><td>Expert</td><td>• Leunt niet langer op regels of richtlijnen
• Heeft snel een diep, intuïtief begrip van de situatie
• Hoeft alleen te analyseren bij nieuwe situaties of wanneer er problemen zijn
• Heeft een visie van wat mogelijk is</td><td>Intuïtief</td></tr>
</table>

TABEL 2 – Alternatieve beschrijving van de vijf vaardigheidsniveaus

Niveau	Graad	Kennis	Werkstandaard	Autonomie	Omgang met complexiteit	Perceptie
1	**Beginner**	Minimaal, enkel boekenkennis; kennis zonder de link met de praktijk	Kwaliteit van resultaten voldoen waarschijnlijk niet aan de norm mits onder nauw toezicht	Behoeft intensieve begeleiding en/of uitgebreide instructies	Is niet in staat om met enige complexiteit omtrent de taak om te gaan	Ziet gebeurtenissen en eigen acties grotendeels
2	**Gevorderde beginner**	Heeft kennis van de belangrijkste aspecten van de praktijk	Eenvoudige taken worden meestal met voldoende kwaliteit uitgevoerd	Is in staat om een deel van de stappen op eigen inzicht te zetten, maar vereist nog steeds begeleiding	Herkent complexe situaties als zodanig maar komt niet verder dan deeloplossingen	Ziet gebeurtenissen en eigen acties als een onderdeel van een causale keten
3	**Bekwaam**	Gedegen kennis van zowel de praktijk als het kennisgebied	Kwaliteit is voldoende al kan het soms verfijning missen	Is in staat om de meeste taken op eigen inzicht uit te voeren	Kan met complexe situaties omgaan door middel van opzettelijke, welbewuste analyse en planning	Begint te zien wat gebeurtenissen en eigen acties betekenen voor de einddoelen van het werk
4	**Deskundig**	Diep begrip van de praktijk en zeer bekend met het kennisgebied	Behaalt routinematig een volledig aan de norm voldoende kwaliteit	Is in staat om de volledige verantwoordelijkheid te nemen voor het eigen werk (en het werk van anderen, indien toepasselijk)	Kan met complexe situaties omgaan op een holistischere manier en heeft meer vertrouwen in eigen beslissingen, planning en analyse kost minder moeite	Ziet het 'hele plaatje' en hoe individuele acties daarin passen, houdt het doel in zicht
5	**Expert**	Autoriteit op het kennisgebied en diep begrip van de praktijk.	Excelleert met relatief weinig moeite	Neemt naast het bovenstaande de verantwoordelijkheid voor het overstijgen van bestaande normen en verwachtingen en creëert eigen interpretaties	Heeft een holistische kijk op de situatie en wisselt met gemak tussen sterk ontwikkelde intuïtie enerzijds en een analytische aanpak anderzijds	Kan naast 'hele plaatje' ook andere perspectieven waarderen. Heeft een visie van wat mogelijk is.

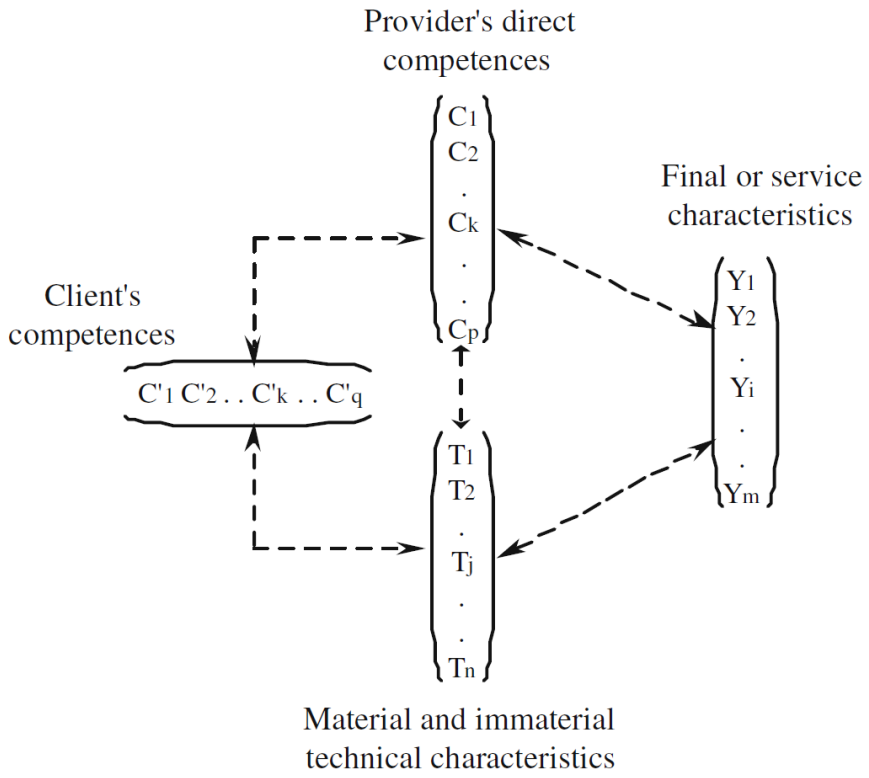
EINDE SURVEY

Appendix B – Een handvat voor de inrichting van innovatie-processen bij Ymor

De bedoeling van deze appendix is om als startpunt te dienen voor the DMers bij het nadenken over hoe innovatieprocessen bij Ymor ingericht zou kunnen worden. Dit startpunt is gebaseerd op het dienstenmodel van Gallouj en Weinstein.

Het model

Het dienstenmodel van Gallouj en Weinstein is afgebeeld in figuur 1. Volgens dit model bestaan diensten uit een set dienst-karakteristieken (i.e. de Y-variabelen in het model). Voor elk van de dienst-karakteristieken zijn een of meer competenties vereist (i.e. de C-variabelen). Daarnaast zijn in veel gevallen bepaalde technologieën vereist (i.e. de T-variabelen) voor het teweeg brengen van de dienst-karakteristieken, waar vaak ook weer bepaalde competenties bij horen. Onder 'technologieën' vallen zowel materiële als immateriële technologieën. Een voorbeeld van materiële technologieën zijn machines. Een voorbeeld van immateriële technologieën zijn software. Ook gestandaardiseerde processen, methoden en technieken vallen onder immateriële technieken. Ten slotte kunnen de competenties van de klant (i.e. de C') ook nog bepalen welke competenties bij een dienstverlener nodig zijn voor het leveren van een dienst-karakteristiek (Y) bij die klant.



Figuur 1 – Model van de karakteristieken van een dienst, overgenomen uit Gallouj en Weinstein (1997)

Innovatievormen in diensten

Op basis van hun model herkennen Gallouj en Weinstein de volgende vormen van innovatie bij diensten:

1. *Radical Innovation*
2. *Improvement innovation/Incremental innovation*
3. *Ad hoc innovation*
4. *Recombinative innovation*
5. *Formalisation innovation*

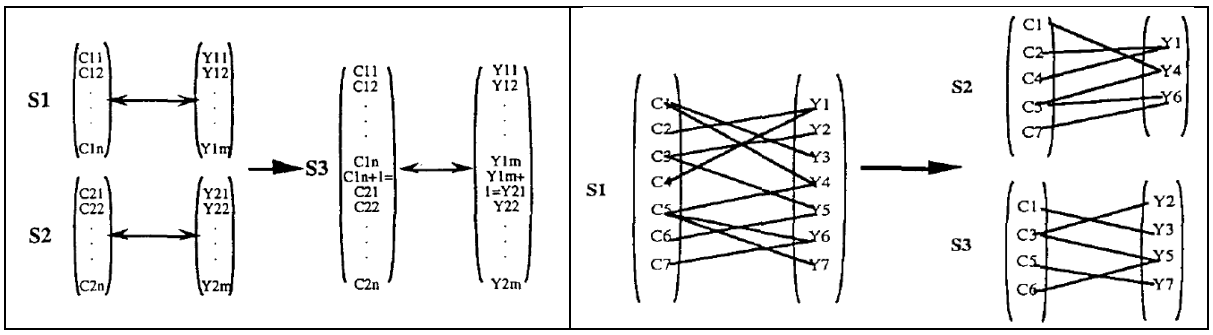
Radicale innovatie vindt plaats wanneer er een volledig nieuwe set competenties en technologische- en dienst-karakteristieken is gecreëerd (Gallouj & Savona, 2009). Een andere vorm van radicale innovatie die wordt erkend is wanneer de set competenties en technologische karakteristieken veranderen maar de dienst-karakteristieken niet. Het voorbeeld dat hierbij gegeven wordt is die van een auto die het paard en wagen vervangt. Beide leveren dezelfde soort dienst-karakteristieken (e.g. transport, snelheid, comfort) maar elk op een volledig andere manier.

Van verbeteringsinnovatie (EN: *improvement innovation*) is sprake wanneer een van de dienst-karakteristieken van hogere kwaliteit wordt zonder dat de rest van de dienst-karakteristieken verandert. Deze verbetering van de dienst wordt veroorzaakt door verbeteringen in competenties of technologische karakteristieken van de dienst. Verbeteringsinnovatie is meestal het gevolg van leren dat van nature voorkomt bij activiteiten, en niet van opzettelijke innovatiepogingen (Gallouj & Weinstein, 1997).

Incrementele innovatie lijkt erg op verbeteringsinnovatie en de twee typen zijn in de praktijk niet altijd goed te onderscheiden (de Vries, 2006; Gallouj & Weinstein, 1997). Bij incrementele innovatie wordt er een dienst-karakteristiek vervangen of toegevoegd, in tegenstelling tot verbeterd. Het betreft dus meestal een uitbreiding van de dienst.

Van zogenaamde *ad hoc* innovatie wordt gezegd dat het typerend is voor kennisintensieve organisaties. *Ad hoc* innovaties worden co-geproduceerd door de dienstverlener de klant en betreffen vaak oplossingen voor een klantprobleem. De vector van dienst-karakteristieken kan worden gezien als die oplossing. Wat typisch is voor deze vorm van innovatie is de *a posteriori* identificatie en formalisatie van de competenties en technologieën die de oplossing mogelijk gemaakt hebben.

Bij hercombinerende innovatie (EN: *recombinative innovation*) worden competenties en/of technologieën op een andere manier gecombineerd of juist gesplitst waardoor er andere diensten ontstaan. Het idee is dat de waarde van deze diensten meer is in deze nieuwe combinatie dan in de vorige combinatie, i.e. dat er synergie plaatsvindt.



Figuur 2 – De twee vormen van hercombinerende innovatie. links: d.m.v. combinatie, rechts: d.m.v. splitsing

De laatste vorm van innovatie die Gallouj & Weinstein (1997) beschrijven is formalisatie innovatie. Deze vorm van innovatie is anders dan de rest want beschrijft niet wat er vernieuwd wordt maar in welke mate de vernieuwing formeel wordt, i.e. gestandaardiseerd wordt. Er is dus sprake van formalisatie innovatie wanneer vernieuwingen die begonnen bij verbeterde competenties worden doorgevoerd naar vernieuwingen in de technologieën. Een voorbeeld is het creëren van een nieuwe methode op basis van eerder opgedane ervaringen (i.e. 'best practices'). Een belangrijk aspect van deze innovatievorm is volgens de auteurs dat impliciete kennis expliciet gemaakt wordt en dat gedurende dat proces vaak kennisgaten worden gevuld.

Inrichting van innovatieprocessen

Er zijn dus verschillende innovatievormen te onderscheiden, maar wat betekent dat voor het inrichten van innovatieprocessen bij Ymor? Wat beheersbaarheid en sturing betreft staat innovatie bij Ymor nog enigszins in de kinderschoenen. Hoewel innovativiteit als een van de prestatievariabelen is geïdentificeerd, kon daarom niet in mee worden genomen in het ontwerp van het informatiesysteem. De stap die eerst gezet dient te worden door de DMers (samenwerking met managers van andere afdelingen) is het definiëren en inrichten van innovatieprocessen.

Bij veel aspecten van management zou nu kunnen worden gerefereerd naar moderne best practices, maar volgens (Ortt & van der Duin, 2008) ligt dat bij innovatie anders. Ten eerste stellen zij dat niet alle bedrijven een formeel innovatie proces hebben, en dat sommige van hen ondanks dat toch succesvolle innovators zijn. Daarnaast blijkt dat bedrijven zeer kunnen verschillen in hun benadering tot innovatie, zelfs in gevallen waar ze tot vergelijkbare innovaties waren gekomen. Ook blijkt uit onderzoek dat er soms verschillende innovatie-management methoden worden gebruikt binnen één bedrijf, al geldt dat meestal alleen voor grotere bedrijven. Waar het volgens hen op neer komt is dat verschillende situaties om verschillende aanpakken vragen. Kortom, het inrichten van innovatie management lijkt helaas niet zo simpel te zijn als het volgen van algemene best practices. De auteurs beschrijven in de rest van het artikel hoe twee grote bedrijven, Shell en Philips, context-specifiek innovatie management hebben ingericht.

Wat we tot zover hieruit kunnen opmaken is dat er verschillende innovatieprocessen nodig zijn voor verschillende vormen van innovatie. Dus voor radicale innovatie is een ander proces nodig, bijvoorbeeld met een incubator, voor vernieuwing innovatie weer iets anders, voor ad hoc innovatie ook, enzovoorts.

De innovatievormen die voor de Delivery afdeling waarschijnlijk het belangrijkst zijn, zijn nummers twee, drie en vijf van de vorige lijst: verbeteringsinnovatie, ad hoc innovatie en formalisatie-innovatie. Radicale innovatie kost namelijk in de regel erg veel ontwikkelingstijd en de voornaamste taak van Delivery is om de beschikbare tijd te gebruiken voor het verlenen van de bestaande diensten aan de klant. Hercombinerende innovatie lijkt daarnaast belangrijker te zijn voor Sales, aangezien zij dagelijks voor klanten combinaties maken van dienst-karakteristieken die de meeste waarde hebben.

Wat overblijft zijn dus verbeteringsinnovatie, ad hoc innovatie en formalisatie-innovatie. Deze drie innovatievormen verdienen elk een eigen procesinrichting en voor elk van de drie is de specificatie van de randvoorwaarden voor succes nodig.

Bij Ymor zijn al sporen van verbeteringsinnovatie-processen waar te nemen. Er zijn bijvoorbeeld een aantal medewerkers bij Delivery die met een nieuwe technologie in aanraking zijn gekomen en hebben ontdekt dat deze interessante mogelijkheden bieden. Zij hebben zich vervolgens in deze tool verdiept

en introduceren het aan anderen. Als deze vormen van verbeteringsinnovatie wenselijk is dan is het waarschijnlijk een goed idee om extra tijd voor deze mensen beschikbaar te maken aan zulk soort innovaties te werken. (Misschien echter alleen voor mensen die hun *traineeship* hebben afgerond, aangezien zij veel tijd nodig hebben om de bestaande dienstverlening te leren uitvoeren.)

Ook is men al veel bezig met het verbeteren van de eigen software in samenwerking met Development. Dit uit zich in sporadische feature-brainstorm sessies en test-trajecten van Ymonitor software. Het proces waarmee de in-house gebouwde software wordt verbeterd is wellicht het eerste dat verder dient te worden ingericht, want hiervan hebben meerdere mensen van zowel Development als van Delivery de noodzaak al onderstreept.

Voor ad hoc innovatie is het belangrijk dat er veel met de klant samengewerkt wordt. Deze vorm van innovatie is bij Ymor ook al waar te nemen want sommige teams zijn zeer ingebed bij een grote klant en werken met hen samen aan grote problemen. Volgens mij is dit een belangrijk deel van wat het management van Ymor bedoelt wanneer zij spreken van het ontzorgen van de klant door middel van *embedding*. Het betekent waarschijnlijk wel dat teams die dit doen minder tijd hebben voor andere klanten. Voor ad hoc innovatie is het dus belangrijk dat er genoeg capaciteit is om een (deel van een) team veel tijd te laten besteden bij een enkele klant.

Van formalisatie-innovatie zijn ten slotte ook tekenen terug te zien bij Delivery. Formalisatie innovatie refereert naar het transformeren van onbewuste en/of ontastbare kennis naar expliciete, tastbare kennis. De zogenaamde expert avonden die door Ymor eens in de zoveel weken worden georganiseerd lijken hier mee te maken hebben. Tijdens zulke avonden wordt er namelijk klant-specifieke kennis en de bijbehorende *lessons learned* van een team aan de rest van de afdeling gedeeld. Ook lijkt de wens naar de 'professionalisering van de organisatie' waar het management over spreekt te wijzen naar de intentie om de kennis en werkwijzen binnen de organisatie vast te leggen in methoden en modellen. Ook dit project heeft waarschijnlijk bijgedragen aan deze vorm van innovatie, namelijk door het identificeren van de prestatievariabelen van de Delivery afdeling en het bepalen van de relevante competenties. Om deze vorm van innovatie verder te stimuleren raad ik aan om de ontwikkeling en/of aanschaf van E-learning te overwegen. E-learning wordt steeds meer toegepast in bedrijven (Hidalgo & Albors, 2008) en het medium lijkt me goed aan te sluiten bij een afdeling van IT ingenieurs.

Dit waren de richtlijnen die ik tijdens dit project ben tegengekomen. Het belangrijkste is om te bedenken dat verschillende vormen van innovatie ook verschillende processen nodig hebben. Alle vormen van innovatie die ik heb genoemd zijn belangrijk voor Ymor, maar voor Delivery gaat het waarschijnlijk vooral om vernieuwingsinnovatie, ad hoc innovatie en formalisatie-innovatie.