

Financial impact of downtime decrease and performance increase of IT services

Joey Oostenbrink
University of Twente
P.O. Box 217, 7500AE Enschede
The Netherlands

ABSTRACT: Many companies have an IT infrastructure that suffers from instability or performs badly. These problems may entail serious costs for companies depending on these IT infrastructures. Application performance tools exist that continuously monitor IT chains and analyze the causes of these problems. The aim of this study is to investigate how much cost can be saved by applying these tools. This paper describes three small impact studies. The research has been conducted at Ymor, a company specialized in providing application performance management services. The impact studies are based on real cases at three of Ymor's customers. The findings from the impact studies are that companies can save a substantial amount of money every year by using application performance tools. We recommend to use a number of proposed variables to assess the amounts of money future customers can save when applying application performance tools.

Supervisors: prof. dr. ir. Bart Nieuwenhuis, University of Twente
Lucas Meertens MSc, University of Twente
ir. Herco Reinders, Ymor

Keywords

Ymor, application performance management, downtime, performance, response time

1. INTRODUCTION

The importance of IT in organizations is increasing rapidly. With this, the need to ensure a stable IT infrastructure is growing. Unstable and slow applications can increase costs and decrease revenue, but can also have indirect effects. Examples include a fine for violating service level agreements or even reputational damage, in case the affected application is used by a company's customers. Organizations are aware of these types of costs, but often the exact amounts are not known. Also, it is often problematic to determine the origin of IT failure, especially if a number of applications form a chain in which all parts interact with each other. Several companies close this market gap by offering organizations services to provide insights in their IT issues.

Many companies have an IT infrastructure that suffers from instability or performs badly. These problems may entail serious costs for companies depending on these IT infrastructures. Application Performance tools exist that continuously monitor IT chains and analyze the causes of these problems. Operational examples are services provided by Ymor, a Dutch company specialized in application performance management: Yvalidate is a service that tests applications so you can check whether its performance is adequate before launching the (updated version of an) application. Ymonitor is a service that enables you to assess the performance of an application and check the duration and impact of downtime. The third service Ymor offers is called Troubleshoot and involves checking an entire chain to discover where a problem originates. Very often, Ymonitor is used as a part of the Troubleshoot. A brief description of Ymor can be found in Appendix 1.

The aim of this study is to investigate how much cost can be saved applying application performance tools. In this study cost savings are studied by looking at two aspects application management tools are aiming at decreasing downtime and improving performance of a customer's information system. Since quantitative (preferably financial) output is requested, this is also incorporated in the research question stated below.

What is the financial impact of downtime decrease and performance increase of IT services by using application performance tools?

This question can be broken down into two sub-questions:

1. How can downtime decrease and performance increase be quantified to financial impact?
2. What financial impact has been realized at current customers?

The first question is answered through a scientific literature study that provides clear definitions of the concepts of downtime and performance. The goal of a literature study is to find calculations, concepts and recommendations that can prove useful for the interviews in the empirical study. These findings

will be combined with input from empirical research into a list of indicators for calculating downtime and performance. A part of this data can be collected through Ymor, so interviews will be conducted to obtain data from company representatives for the other indicators. The results will consist of the compiled list of indicators, which is the answer to question 1. Three short impact studies, including the financial impact of application performance services for those three companies, will provide an answer to question 2.

2. LITERATURE STUDY

2.1 Methodology

An academic foundation is required for the empirical research, this will be done by carrying out a literature review about topics as value, downtime and performance in terms of IT systems. Existing literature reviews on these subjects have been studied as a starting point to determine what academic literature might be useful for definitions of the key topics and to come up with variables to include in the calculations. This led to common practices in these areas and the various changes made to them in later years. Apart from scientific literature from journals, this study also contains whitepapers. These are more oriented towards practice and make it easier to shift from literature research to empirical research.

2.2 Impact and value

In order to answer the research question, it is essential to find out how certain tools can impact an IT application through improvements. This will be described in this study as the value that application performance tools create. Barney (1991) argues that a product or service is valuable when it increases the firm's performance. According to Porter (2008), value is created by a set of companies that form a vertical chain. Barney's definition appeals more to this particular research, since value is presented here as improvement at one company, rather than improvement in a network of firms. Bowman and Ambrosini (2000) distinguish between use value and exchange value. The former is estimated by customers, the latter is not realized before the product or service is actually sold. Since companies often do not know how much they will benefit from application performance services, they cannot accurately estimate the use value; in order words, the customers' perceived value has to be increased by identifying how these tools can improve IT availability (by decreasing downtime) and IT performance (by optimizing the system).

Customers estimate value as the perceived benefits divided by the perceived sacrifice, which includes all costs concerning a purchase (Ravald & Grönroos, 1996; Ulaga & Chacour, 2001; Zeithaml, 1988). Zeithaml (1988) argues that customers perceive value on the basis of personal factors: some might regard low price as value, while others compare the price with product or service quality. Some researchers, for example Hamel and Prahalad (2013), disprove the inclusion of sacrifices by stating that perceived value only represents benefits.

Both views agree on the fact that a customer's perception includes the benefits of a product or service. Sweeney and Soutar (2001) emphasize the difference between a customer's perceived value and satisfaction, with an important difference being that satisfaction occurs after the customer used the product or service, while this is not necessary for a value perception. Combining these arguments, perceived value is defined in this study as *the customer's perception of service benefits before actually using the service*.

Lapierre (2000) identified several drivers for value, which are related to products, services or relationships. Since this research is about services, the service related benefits are listed here: responsiveness, flexibility, reliability, technical competence (Lapierre, 2000, p. 125). These benefits can be framed in a context which is applicable to the stated business problem.

- One of the goals for measurement is the decrease of information system downtime, which is clearly connected to reliability: "the ability to perform the promised service dependably and accurately" (Parasuraman, Zeithaml, & Berry, 1988, p. 23), and also continuously (Avižienis, Laprie, Randell, & Landwehr, 2004). In the context of information systems, this would be availability to end users.
- Another goal, increasing the performance of information systems, can be related to responsiveness and flexibility: responsiveness means providing answers and solutions to customers and their problems quickly (Lapierre, 2000; Parasuraman et al., 1988), response times for example; and when ICT systems are flexible, they are able to adapt to various situations (Aerts, Goossenaerts, Hammer, & Wortmann, 2004; Lapierre, 2000; Li & Zhao, 2006).

To summarize: less downtime means increased availability, while an increase in performance might result in lower response times and better adaptability. The next sections describe how these two goals can be measured in a quantitative way.

2.3 Decreasing downtime

Downtime is the time an information system is 'down', i.e. unavailable for the customer to use and can be calculated by measuring the time between an interruption and the reboot that follows (Murphy & Gent, 1995). Availability is the total amount of uptime divided by the sum of uptime and downtime. Apart from this straightforward calculation, Murphy and Gent also provide the following formula (1995, p. 7):

$$Availability = \frac{MTBSI}{MTBSI + MTTR + MTTRc} \quad (1)$$

MTBSI means Mean Time Between System Interruptions and equals the uptime of a system. Consequently, the other two factors in this formula represent the system's downtime: Mean Time To Repair, and Mean Time To Recover, respectively.

When we look at the definition of downtime mentioned earlier, we observe that it equals the sum of MTTR and MTTRc.

Now that downtime has been defined, it is possible to measure it. The next step is to measure what it brings the company. The most obvious approach in quantifying the benefits of downtime decrease is a measure that provides quantitative output such as time or money. Patterson (2002) came up with a formula to estimate the cost of one hour of downtime for IT systems, which involves hourly employee costs, hourly revenue, the percentage of employees affected by an outage, and the percentage of revenues affected by an outage (Formula 2). Patterson argues that employee costs and revenues can be determined easily and emphasizes that they do not have to be perfectly precise for an estimation. The fraction of employees and revenue that are affected by an outage needs to be guessed or ranged.

$$\begin{aligned} \text{Estimated average cost of 1} \quad (2) \\ \text{hour of downtime} = & \text{Empl.} \\ & \text{costs/hour} * \% \text{ Empl's} \\ & \text{affected by outage} + \text{Avg.} \\ & \text{Rev./hour} * \% \text{ Rev. affected} \\ & \text{by outage} \end{aligned}$$

While this is, in many ways, a measure that greatly fits the problem description, it might not be a feasible one. Of course, customers can be asked to provide hourly employee costs and hourly revenue. The problem lies in the other two factors, which have to be estimated. An estimate is "a prediction based on a probabilistic assessment (...) and should be the most likely value accompanied by upper and lower bounds" (DeMarco, 1982; as cited in Grimstad, Jørgensen, & Moløkken-Østvold, 2006, p. 304). Although many estimation methods exist, the most common way is expert judgement (Jørgensen, 2007). In the case of customers, an 'expert' would be a person with extensive knowledge about the application involved in the downtime estimations. This can be a developer of the system, or a business owner, for example. Expert estimations are often intuition-based and biased. Still, Jørgensen (2007) did not find support for the claim to replace expert estimations by estimation models. In many cases a combination of both seems the best option. Some ideas to improve the accuracy of expert estimates are: combining estimates of various experts, checking the estimation background of experts and letting experts criticize their own estimates (Jørgensen, 2004).

Thus, a suitable approach to determine the benefits of downtime is by using Patterson's formula, combined with recommendations for expert judgement in order to assess the factors that have to be estimated. The empirical study therefore consists of interviews in which we will try to get answers from company representatives that help filling in the formula.

2.4 Improving performance

The previous section concerned downtime, a phenomenon that can be objectively measured and of which the costs can be calculated as well, given a particular set of factors. Performance, however, is hard to measure. The first step is to determine what actually defines the performance of an information system. This can be availability or response times, among other factors (Ludwig, Keller, Dan, King, & Franck, 2003). Regardless of what is perceived as 'performance', the corresponding factors are included in a service level agreement, commonly abbreviated as SLA (Keller & Ludwig, 2003). Hence, violations of an SLA are indicators of performance issues (Khanna, Beaty, Kar, & Kochut, 2006). An SLA is a contract that contains certain service levels a supplier has to provide the customer with; service levels are numbers that show when a certain service parameter is acceptable (Lewis, 2001). So, to find out what 'performance' actually means in terms of IT applications, we should look into companies' service level agreements. Those SLAs might contain certain levels of acceptability related to the concepts of responsiveness and flexibility, as well as downtime, which directly influences overall system performance.

A frequently cited model in information systems literature is the I/S Success Model (DeLone & McLean, 1992). This model views information system success as a process, which is shown in Figure 1.

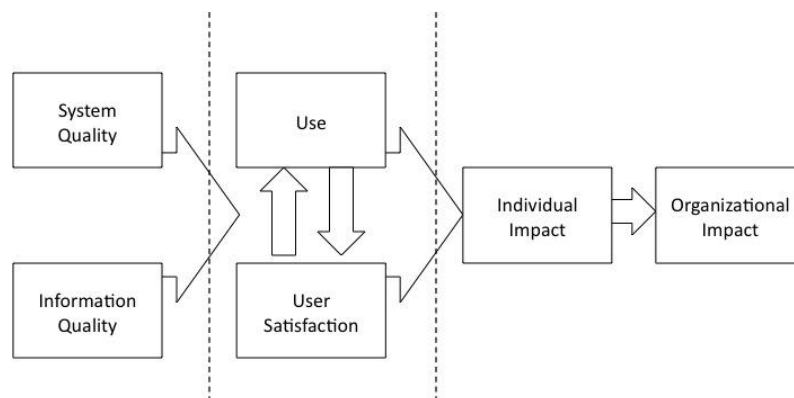


Figure 1. I/S Success Model. Adopted from DeLone and McLean (1992, p. 87).

Of this process, one part in particular is relevant to the concept of performance. System quality is the only one out of six categories that has been placed on the technical level of information, which is about an information system's accuracy and efficiency. Several measures of system quality have appeared in literature, including reliability, response time, data accuracy and flexibility (DeLone & McLean, 1992). A decade later, DeLone and McLean updated their model, although the system quality component remained the same. Five success metrics of system quality were mentioned: adaptability, usability, response time, availability, and reliability (DeLone & McLean, 2003, p. 26). Now that system performance has been broken down into several metrics, these metrics have to be quantified. The first three correspond to the information drawn from

Lapierre (2000) earlier in this paper, while the final two represent the other topic for the empirical study: downtime decrease. So, concerning the analysis of performance, three metrics will be emphasized: adaptability, usability and response time, which will be discussed in the following sections.

2.4.1 Adaptability

Stakeholders often require certain changes to be made in the environment, and adaptability is the extent to which an information system can adapt to these changes without sufficiently impacting its elements (Liu & Wang, 2005). A broader definition can be provided by ignoring the focus on stakeholders, and include every possible change in the system's environment (like the concept of flexibility defined earlier in this paper). Liu and Wang (2005) provide two metrics to quantify adaptability: IOSA (Impact On the Software Architecture) and ADSA (Adaptability Degree of Software Architecture). Depending on the amount of change scenarios, a certain amount of impact calculations have to be made. The sum of these impact amounts equals the IOSA. The ADSA has an inverse relation to the IOSA and is a value between one and zero. The downside to applying this approach in the empirical study is its difficulty. Calculating the IOSA involves factors such as the amount of impacted architecture components and connectors, and the probability of certain changes to occur. It seems unlikely to obtain all of this through interviews. Maybe even more important is the fact

that application performance services do not affect a system's adaptability in a direct way. An optimized system may be quicker to adapt to certain conditions, but adaptability takes place in the design process that precedes the moment in which external parties get involved.

2.4.2 Usability

Objectively calculating the usability of an information system is impossible, since its usability depends on the context in which it is used (Brooke, 1996). Consequently, usability cannot be measured absolutely. Several usability surveys have been designed. An example is the System Usability Scale (SUS) by Brooke (1996). Bangor, Kortum, and Miller (2008) compiled an overview of various surveys, but their primary purpose was to provide guidelines for the use of SUS in practice. Because of this, it makes it easy to conduct this survey in practice, especially since it just comprises ten questions on a Likert scale. However, usability is still a subjective metric since it is based on user input. Like adaptability, usability is a feature of an application that is not directly affected by application performance services. A system might get easier to use, but this is then accomplished by making changes to other aspects of the system.

2.4.3 Response time

Compared to the previous two metrics, response time seems the one that is the easiest to measure. The response time of a computer system is the time it takes to process an input (and respond to it) and is frequently expressed in seconds (Hoxmeier & DiCesare, 2000; Joseph & Pandya, 1986). The topic of response time has been present in academic literature for several decades, primarily in the area of psychology. Since the sixties, research on response times of computers has been published as well. An example is a paper by Miller (1968), which contains a list of various computer responses and estimations of their response times. In order to calculate response times for a particular information system, the system's relevant computer responses should be analyzed first. Very often this is known by managers since their employees tend to complain when response times are too long, so it is likely that this information can be collected during a set of interviews.

The brief review of possible performance measures shows that it can prove to be quite challenging to accurately estimate the adaptability and usability of information systems. Response times are somewhat easier to measure and will serve as the main metric for assessing performance improvements. Since system quality and user satisfaction are strongly correlated (Petter, DeLone, & McLean, 2008), an increase in system performance would be highly desired by management. Therefore, the empirical study will look further into the measurement of performance by assessing what kind of metrics are currently in place at Ymor or its customers. A decrease in response time can be combined with the wage sum of employees that experience this response time, in order to calculate the amount of money that can be saved after realizing the decrease.

3. EMPIRICAL STUDY

This section of the paper comprises the real-world studies at Ymor, which is constructed as follows: firstly, key performance indicators are identified which can be used to calculate the quantitative benefits mentioned in the literature study. Secondly, each impact study will be described briefly, including the KPIs identified in that particular situation. Finally, the results of this study will be assessed by checking for similarities between the three cases and constructing a basic overview of important KPIs for the sales department to convince new customers.

3.1 Key performance indicators

The first step is to determine which factors are needed to calculate the benefits of decreased downtime and increased performance. For this, the literature study will be used, combined with information provided by Ymor. The key performance indicators (KPIs) used by Ymor are categorized in the following three areas:

- *Health*: The health of an information system. This can be determined by various factors, including the amount of incidents, memory usage, the amount of service calls.
- *Efficiency*: This is a straightforward characteristic for an information system. Efficiency is achieved by decreasing the time-to-market, decreasing the length of downtime and lowering the cost of ownership, among other things.
- *Risk*: Information systems should be equipped with various means of protection to minimize risk. This can be a high bandwidth capacity for web servers to counter DDoS attacks, but also the availability of a disaster recovery plan so issues can be resolved quicker when something goes wrong.

This list is similar to ITIL (Information Technology Infrastructure Library), a list of best practices in IT service management, as well as ISO 20000. The latter is a standard for IT service management that offers a set of best practices and is based on ITIL (Buchsein & Dettmer, 2008). It includes topics as incident management, service reporting and availability management. Some of the main benefits of ITIL are: negotiated achievable service levels, efficiency in service delivery, and measurable, improvable services and processes (Arraj, 2010, p. 5). The measurability of services is especially important to this research. The list of health, efficiency and risk indicators contains KPIs that aim for this. The categories have some indicators in common as well. Combining the literature study and the KPI list provided by Marketing and Communications resulted in a list of indicators for assessing the value of application performance services (see Table 1). Indicators 1, 2, 4 and 5 are found at Ymor. The others are added as a result of the literature study.

3.2 Downtime

The literature study mentioned that downtime consists of the Mean Time To Repair and Mean Time To Recover, together representing the

Indicator	Category	Where to find?	Type of measure
1 Mean Time To Identify (MTTI)	Downtime	Ymonitor	Time (minutes/hours/days)
2 Mean Time To Repair (MTTR)	Downtime	Ymonitor	Time (minutes/hours/days)
3 Service level	Downtime	Service level agreement	Percentage
4 Response times	Performance	Ymonitor	Time (seconds)
5 Customer satisfaction	Both	Ymonitor	'Apdex' (custom indicator)
6 Amount of users	Both	Ask customer	Number
7 Total wages of users	Both	Ask customer	Financial

Table 1. KPIs for assessing the value of Ymor's services.

complete time an application is unavailable. Within Ymor this time period is broken down to the time to identify the problem (MTTI) and the time to repair the problem after it is identified (MTTR). So the sum of indicators 1 and 2 represents the average duration of downtime. Indicator 3, the service level, can be found in service level agreements (SLAs) as a minimum amount of availability or a maximum amount of downtime for an application. This is very often a percentage close to 100%, such as 98%. Next to the agreed service levels, it is also relevant to check the actual service levels that are achieved in practice. Perhaps a certain business-critical application is characterized by an availability of 97.6%. Its average occurrence of downtime might last for twenty hours, for example, or the amount of occurrences per year is provided. With this information the total annual downtime can be calculated.

Of course this information is a lot more valuable when management can observe the cost of these amounts of downtime, hence the addition of indicators 6 and 7. When we know how many employees actually use the application and what they earn per hour, it is possible to calculate the cost of downtime concerning total employee wages. Downtime generates more costs than just employee salary, depending on the type of application (Vision Solutions, 2008). Unavailability can impact customers when a service desk cannot operate normally, or when an online system that is used by customers is not accessible. In case downtime affects the customer, indicator 5 becomes relevant. For many customer-oriented applications, customer satisfaction is the main goal. In case the application is an internal one and is only used within the company, customer satisfaction metrics can be used to determine how satisfied the company is with the improvements of their application. For both internal and external applications, users can call a service desk when they experience problems. Downtime represents a moment during which no one can properly use the application, so the amount of service calls is probably quite high then. This is represented by indicator 6.

3.3 Performance

The main indicator of performance in this paper is response time (indicator 4), as concluded in the literature study. Response times are relevant to everyone using the system and can be quite costly when they are high. Many companies require the user to log into an application before they can actually use it. Thus, the response time of logging on is a quite general one and is known by many employees among various companies. However, most response times are very application-specific, since they depend on the kind of functionality the application offers and the type of actions the user has to carry out. Logging in is something that happens only once per session. A focus on more frequently executed processes seems wiser. For most companies, key response times are known and these can be included in the analysis.

Of course, response time is not the only factor that should be taken into account. It should be linked to monetary values, by including indicator 7 and 8. A decrease in response time, combined with the wage of employees working with the system, provide an amount of money that can be saved when you achieve this particular decrease. Also, concerning the amount of employees, you could calculate how many people are actually needed carry out all the tasks within the application after the decrease, possibly resulting in layoffs. It is quite obvious that a decrease in response times will increase user satisfaction and will decrease the amount of service calls (indicators 5 and 6). Overall, a decrease of response time affects the company in many ways and can greatly increase organizational performance, especially in the case of business-critical applications.

3.4 Impact studies

In order to show the value of application performance services, three short impact studies have been written to document the financial impact of application performance services. Each is about a company where Ymor successfully implemented their solutions, and thus is suitable as convincing evidence for potential customers. Interviews were conducted with Ymor employees who work closely with the three companies. A wide range of documents and other data within Ymor was collected as well. A limited amount of financial data was obtained through representatives at the customer companies. Because the available information is different in each impact study, these do not strictly follow the list of KPIs that was constructed in section 3.1. Instead, the cases use indicators and data that was available in practice. Section 4 summarizes the construction of calculations in the impact studies and emphasizes common variables that can be valuable to future calculations at Ymor.

Due to confidentiality reasons, this section has been removed from the public paper.

4. CONCLUSION

4.1 Results

The goal of this paper is to assess the financial impact of downtime decrease and performance increase, by using application performance tools. One should multiply the decrease of downtime in hours by the cost of one hour of downtime, which can consist of different factors. Three common factors are the number of active users, the cost of an unproductive hour, and the productivity decrease when downtime occurs. There is no universal way to calculate the value of a decrease in response times, since these can be different for each IT application. A suggested approach is to determine the response times of the most important user actions and measure the value of their decreases. An alternative is to calculate the performance increase of the entire application, but for that one should find out the frequency of each response (since not all take place as often as the others). In either case, the

decrease in response time should be multiplied by the cost of an unproductive hour and the amount of active users. The productivity decrease is assumed to be 100% since response times are mostly too short to perform another task while you wait. These proposed calculations for value are summarized below and put in an overview for the sales kit at Ymor (Appendix 2).

$$V_{downtime} = \Delta D * U * C * P$$

$$V_{response\ time} = \Delta R * U * C$$

V = value of decrease (in downtime or response time) in €

Δ = change (in downtime or response time)

U = amount of active users

C = cost per unproductive hour

P = productivity decrease in %

4.2 Limitations

Limitations of this research fall into three categories: the short timeframe, the gap between theory and practice, and the unavailability of certain information. Due to the short timeframe of this research, only a limited set of cases could be handled. Concerning the gap between theory and practice, the list in section 3.1 is a good example. This list, which was the final product of the literature study, could not be applied directly in the impact studies. The previous section showed that the value of application performance services cannot be calculated easily with a standardized list of indicators. Each impact study features its own set of factors that form the total annual savings together. The formulas for value mentioned above are the empirical result of the impact studies and are partly based on the table of KPIs. This has to do with the third type of limitation: unavailable information. For example, there was no clear data available for MTTI and MTTR in the researched cases. This was resolved by replacing them with total downtime, which is the sum of these two. Another example are the estimations of lost productivity per year were available for the municipality of The Hague. At NS this was not the case and therefore my own estimations had to be used. This can affect the validity of this paper to a certain extent, although there were no better options available to perform the necessary calculations.

4.3 Future research

Future researchers might consider using this paper for assistance in coming up with a standardized way

to measure the cost of downtime and productivity loss due to long response times. We assume that many large firms developed their own way to calculate this. However, there is no unified method available in academic literature, to the best of my knowledge. A substantial number of firms might be interested in such developments, since they would like to know how they can cut costs by optimizing their IT chains. A larger number of companies can be analyzed, enabling the researchers to generalize their findings. A study of three cases is too small for that.

4.4 Recommendations

The results of this study lead to a number of recommendations, of which most are directed towards Ymor. Parts of the impact studies in section 3.4 can be used by the marketing department at Ymor to showcase the value of application performance services on the company website. The sales department can use the impact studies, the theory-based list of indicators, and especially the value formulas to construct their future sales kit. The impact studies can also be used to convince potential customers of the financial benefits that a collaboration with Ymor could have. Since some indicators in Table 1 have to be collected from the customer, the advice is to do this in an early stage so calculations can be made as soon as possible.

Though most recommendations are targeted towards Ymor, many companies might find these calculations useful. Organizations that are interested in application monitoring might use the formulas to calculate possible benefits before contacting a company for application performance management. Apart from showing whether such an investment pays off, the calculations can also increase the awareness of IT issues and their costs within the organization. This might even increase the chances of organizations hiring application performance companies for preventive monitoring.

4.5 Acknowledgements

I would like to thank all employees at Ymor that were willing to assist me during my research. I especially appreciate the help of my external supervisor Herco Reinders, who supported me heavily during the empirical research. Also, I want to mention Iman Alipour and Inge Vollebregt for their contribution to this project. I am grateful for all the support I have been given.

5. REFERENCES

- Aerts, A., Goossenaerts, J. B., Hammer, D. K., & Wortmann, J. C. (2004). Architectures in context: on the evolution of business, application software, and ICT platform architectures. *Information & Management*, 41(6), 781-794.
- Arraj, V. (2010). ITIL®: the basics. *Buckinghamshire, UK*.
- Avizienis, A., Laprie, J.-C., Randell, B., & Landwehr, C. (2004). Basic concepts and taxonomy of dependable and secure computing. *Dependable and Secure Computing, IEEE Transactions on*, 1(1), 11-33.
- Bangor, A., Kortum, P. T., & Miller, J. T. (2008). An empirical evaluation of the system usability scale. *Intl. Journal of Human-Computer Interaction*, 24(6), 574-594.
- Barney, J. (1991). Firm resources and sustained competitive advantage. *Journal of management*, 17(1), 99-120.

- Bowman, C., & Ambrosini, V. (2000). Value creation versus value capture: towards a coherent definition of value in strategy. *British Journal of Management*, 11(1), 1-15.
- Brooke, J. (1996). SUS-A quick and dirty usability scale. *Usability evaluation in industry*, 189(194), 4-7.
- Buchsein, R., & Dettmer, K. (2008). ISO/IEC 20000 – IT Service Management – Benefits and requirements for service providers and customers.
- DeLone, W. H., & McLean, E. R. (1992). Information systems success: The quest for the dependent variable. *Information systems research*, 3(1), 60-95.
- DeLone, W. H., & McLean, E. R. (2003). The DeLone and McLean model of information systems success: a ten-year update. *Journal of management information systems*, 19(4), 9-30.
- Grimstad, S., Jørgensen, M., & Moløkken-Østvold, K. (2006). Software effort estimation terminology: The tower of Babel. *Information and Software Technology*, 48(4), 302-310.
- Hamel, G., & Prahalad, C. K. (2013). *Competing for the Future*: Harvard Business Press.
- Hoxmeier, J. A., & DiCesare, C. (2000). System response time and user satisfaction: An experimental study of browser-based applications. *AMCIS 2000 Proceedings*, 347.
- Jørgensen, M. (2004). A review of studies on expert estimation of software development effort. *Journal of Systems and Software*, 70(1), 37-60.
- Jørgensen, M. (2007). Forecasting of software development work effort: Evidence on expert judgement and formal models. *International Journal of Forecasting*, 23(3), 449-462.
- Joseph, M., & Pandya, P. (1986). Finding response times in a real-time system. *The Computer Journal*, 29(5), 390-395.
- Keller, A., & Ludwig, H. (2003). The WSLA framework: Specifying and monitoring service level agreements for web services. *Journal of Network and Systems Management*, 11(1), 57-81.
- Khanna, G., Beaty, K., Kar, G., & Kochut, A. (2006). *Application performance management in virtualized server environments*. Paper presented at the Network Operations and Management Symposium, 2006. NOMS 2006. 10th IEEE/IFIP.
- Lapierre, J. (2000). Customer-perceived value in industrial contexts. *Journal of Business & Industrial Marketing*, 15(2/3), 122-145.
- Lewis, L. (2001). *Managing business and service networks*: Springer Science & Business Media.
- Li, L., & Zhao, X. (2006). Enhancing competitive edge through knowledge management in implementing ERP systems. *Systems Research and Behavioral Science*, 23(2), 129-140.
- Liu, X., & Wang, Q. (2005). *Study on application of a quantitative evaluation approach for software architecture adaptability*. Paper presented at the Quality Software, 2005.(QSIC 2005). Fifth International Conference on.
- Ludwig, H., Keller, A., Dan, A., King, R. P., & Franck, R. (2003). Web service level agreement (WSLA) language specification. *IBM Corporation*, 815-824.
- Miller, R. B. (1968). *Response time in man-computer conversational transactions*. Paper presented at the Proceedings of the December 9-11, 1968, fall joint computer conference, part I.
- Murphy, B., & Gent, T. (1995). Measuring system and software reliability using an automated data collection process. *Quality and reliability engineering international*, 11(5), 341-353.
- Parasuraman, A., Zeithaml, V. A., & Berry, L. L. (1988). Servqual. *Journal of retailing*, 64(1), 12-40.
- Patterson, D. A. (2002). *A Simple Way to Estimate the Cost of Downtime*. Paper presented at the LISA.
- Petter, S., DeLone, W., & McLean, E. (2008). Measuring information systems success: models, dimensions, measures, and interrelationships. *European journal of information systems*, 17(3), 236-263.
- Porter, M. E. (2008). *Competitive strategy: Techniques for analyzing industries and competitors*: Simon and Schuster.
- Ravald, A., & Grönroos, C. (1996). The value concept and relationship marketing. *European journal of marketing*, 30(2), 19-30.
- Sweeney, J. C., & Soutar, G. N. (2001). Consumer perceived value: The development of a multiple item scale. *Journal of retailing*, 77(2), 203-220.
- Ullaga, W., & Chacour, S. (2001). Measuring customer-perceived value in business markets: a prerequisite for marketing strategy development and implementation. *Industrial marketing management*, 30(6), 525-540.
- Vision Solutions. (2008). Assessing the financial impact of downtime. *Irvine, CA, USA, White Paper*.
- Zeithaml, V. A. (1988). Consumer perceptions of price, quality, and value: a means-end model and synthesis of evidence. *The Journal of marketing*, 2-22.

Appendix 1: Information about Ymor

Company description

Ymor is a company specialized in application performance management with the goal of relieving the end user. Examples include performance issues and monitoring of IT chains (www.ymor.nl). Two services developed by Ymor are Yvalidate and Ymonitor. Yvalidate can test applications so you can check whether its performance is adequate even before launching the (updated version of an) application. Its benefits include avoiding investments for fixing performance problems and avoiding reputational damage. Ymonitor enables you to assess the performance of an application and check the duration and impact of downtime. This results in rapid problem-solving so exposure of failures to end users can be minimized. The third service Ymor offers is called Troubleshoot and involves checking an entire chain to discover where a problem originates. Very often, Ymonitor is used as a part of the Troubleshoot.

Appendix 2: Starting point for the Ymor sales kit

List of variables needed to calculate value:

Symbol	Variable
ΔD	Decrease in downtime
ΔR	Decrease in response time
U	Amount of active users
C	Cost per unproductive hour
P	Productivity decrease in %

The value of downtime decrease and response time decrease:

$$V_{downtime} = \Delta D * U * C * P$$

$$V_{response\ time} = \Delta R * U * C$$

Example setup of a calculation sheet*:

downtime	ΔD	x	U	x	C	x	P	=		
response time	ΔR	x	U	x	C			=		
+										
total value										

* U , C and P should be requested from the customer