Establishing a Measurement System for IT Service Management Processes: A Case Study

Marko Jäntti
University of Eastern Finland
School of Computing
Software Engineering
P.O.B 1627, 70211
Kuopio, Finland
marko.jantti@uef.fi

Antti Lahtela
Regional State Administrative Agency for Eastern Finland
Development and Steering Unit for the Local Register Offices
P.O.B 1348, 70101
Kuopio, Finland
antti.lahtela@avi.fi

Jukka Kaukola
Tieto Finland Oy, Energy
P.O.B 1199, 70211
Kuopio, Finland
jukka.kaukola@tieto.com

***Abstract*—IT service providers need effective and efficient methods how to design, manage, support and measure IT services. IT Infrastructure Library (ITIL) is the most widely used IT service management framework. It consists of best practices that can be used in implementing, for example service support processes, such as incident management and problem management. Although IT service management frameworks and standards provide some guidelines for measuring IT services, many IT organizations consider service measurement as a difficult task. The research problem in this paper is how the measurement of the IT service support processes can be improved? The main contribution of this study is to 1) describe the implementation process of the ITIL-based IT service management measurement system (ITSM-MS), 2) describe the system architecture and the main functions of the ITSM-MS, 3) propose a framework for measuring IT services and 4) present the lessons learnt from the ITSM-MS implementation process. The ITSM-MS can be used to measure the performance of IT service support processes. The measurement system was developed in cooperation between a software engineering research project and a large IT service provider company in Finland.**

***Keywords*-IT service management; metrics; measurement; system**

I. Introduction

IT service providers are continuously looking for more efficient methods to improve the performance of customer support processes and to reduce the support and maintenance costs. Process improvement actions should be based on the reliable and up-to-date measurement data. Service measurement plays an important role in IT service management. This paper is based on the published conference paper [1].

The business objective of this study is to decrease the manual work related to measurement and reporting of IT service performance. Currently, thousands of IT organizations are implementing IT service management processes. Effective service management requires that there is a measurement system that enables measuring each service management process.

This paper provides a deeper literature review on the measurement of IT service support processes, includes a more detailed description how we implemented an IT service management measurement system (ITSM-MS) for IT service support processes in a large IT service provider organization and proposes a systematic framework for measuring IT services. The study is valuable because it describes a unique case that combines both IT service management and dynamic, real-time measurement tool for IT service processes.

The focus in this paper is on process metrics. According to IEEE Standard for a Software Quality Metrics Methodology a process metric is “a metric used to measure characteristics of the methods, techniques, and tools employed in developing, implementing, and maintaining the software system.” [2]. We extend this definition to involve also IT services.

Many IT service organizations consider the measurement of IT service management processes, especially service support processes, as a difficult task. Difficulties are mainly due to the following four reasons: 1) IT organizations do not have a structured approach for measuring IT services and service management processes, 2) tools used by service support teams do not enable effective measurement, 3) IT service management standards and frameworks do not provide practical examples how to measure support processes, and 4) there are too many options what to measure in service management.

First, in many IT organizations the measurement activities regarding customer support are still carried out as ad hoc activities without specified business gseoals for measurements, a measurement manager that is responsible for improving the measurement process and tools or clear description of the used metrics. Instead of an ad hoc approach, organizations need a structured approach for measuring IT service support processes. The IT organization should consider measurement and reporting as a systematic process that is managed and improved by a process manager and frequently reviewed and where each metric is linked to business objectives.

Second, service support processes are very tool-oriented processes. Unfortunately often, the service reporting and measurement functions are the weakest parts of service desk tools. The tool should enable effective measurement of both resolution times and volumes of support requests. In the ideal situation, the service measurement system provides real-time measurement data about the process performance.

Third, there is a wide selection of IT service management frameworks and standards that IT organizations can use to improve and manage their processes, such as IT Infrastructure Library (ITIL) [3], ISO/IEC 20 000 [4], Control Objectives for Information and related Technology (COBIT) framework [5], Capability Maturity Model Integration (CMMI) for Services [6], and Microsoft Operation Framework (MOF) [7]. These frameworks and standards should provide IT organizations with more practical examples how to measure IT service support processes.

Fourth, IT organizations can measure their IT operations from many different perspectives. Different stakeholders require different metrics and reports. A typical IT service provider organization deals with software products, software projects, and IT services. Products, projects and services are produced by following organization's processes. IT customers, service managers, customer service managers, product managers, process managers, and business managers have all their own requirements regarding measurement and reporting. It is challenging to create a measurement system that satisfies everybody's requirements.

The remainder of the paper is organized as follows. In Section III the research problem and research methods of this study are described. Section II describes the background and related work for our study. In Section IV, we describe the implementation process of the measurement system, the system architecture, the main functions of the system, and the proposed IT service measurement framework. Section V presents the lessons learnt from the ITSM-MS project. Finally, the conclusions are given in Section VI.

II. Background and Related Work

Surprisingly few academic studies have dealt with measurement of service support processes (incident management and problem management) from the IT service management perspective. There are studies that have focused on predicting incident volumes through statistical methods [8] and discussing service level agreements [9]. Service monitoring and measurement should begin immediately after the service level agreement is agreed and accepted and start producing service achievement reports [10].

The ITIL version 2 completely ignores the establishment of a measurement process. The Continual Service Improvement (CSI) of the ITIL version 3 is a step to a better direction. It proposes a 7-step improvement process to support IT service management measurement activities [11]. In the CSI, the measurement is based on three basic concepts: critical success factors (for example, reducing IT costs), key performance indicators (for example, 10 percent reduction in the costs of handling printer-related incidents) and metrics (for example, cost of the improvement effort). Unfortunately, the 7-step improvement program seems to be very abstract and difficult to adopt in practice. A potential canditate for a measurement process is Six Sigma but a lot of work is needed to convert it into IT service management purposes.

Six Sigma is a process improvement model that enables organisations to streamline processes by reducing the number of defects [12]. The Six Sigma approach has two key methodologies: 1) DMAIC (Define, Measure, Analyze, Improve or optimize, Control) that can be used to improve an existing business process and 2) DMADV (Define, Measure, Analyze, Design, Verify) that is used to create a new product or a process.

The measurement of the service support processes will be painful if the service desk tools do not include effective measurement functions. Advanced service desk tools enable monitoring whether service level requirements are met in resolving service desk cases. The tool should inform users if the service level agreements are close to breach (yellow warning code) or have already breaches (red code). The service desk tool must be capable of producing both time-based and volume-based performance reports.

If the service reporting and measurement function does not work properly, it will remarkably increase the manual work in producing process performance reports. In many IT organizations, process managers still have to use Microsoft Excel to produce monthly reports regarding customer support processes. For example, Jäntti, Miettinen and Vähäkainu report in their study that time-based performance metrics were difficult to implement due to tool difficulties [13].

It is surprising that although there are various IT service management frameworks and standards available, IT organizations still have problems in creating metrics and measuring the service management processes. The IT Infrastructure Library (ITIL) is the most widely used framework for IT service management. The support processes of ITIL include incident management, problem management, change management, configuration management, and release management [14]. In this paper, we focus on the first two processes that are often called front-end support processes [15].

The main objective of incident management process is to deal with all incidents including failures, questions and queries reported by the users [16]. This process is related to corrective software maintenance [17], [18]. Problem management in turn focuses on preventing problems and incidents, eliminating recurring incidents and minimizing the impact of incidents. Many organizations have difficulties to implement ITIL-based problem management activities [19]. Jäntti has reported that difficult ITIL terminology causes IT organizations challenges especially when the improvement target is the problem management process [20]. Although ITIL has introduced a selection of IT service support metrics both in version 2 [14] and version 3 [16], it does not provide sufficient information how IT service management process measurements should be done in practice.

The incident management process of the ITIL version 3 contains 15 potential metrics for incident management (e.g., total numbers of incidents, number and percentage of major incidents, mean elapsed time to achieve incident resolution) and 10 metrics for problem management (e.g., the total number of problems/period, the percentage of problems resolved within service level agreement targets, the average cost of handling a problem, the number of major problems). Instead of long lists of metrics, IT service management frameworks should provide more practical examples how to use metrics. Additionally, metrics could be divided by priority into primary metrics and secondary metrics.

Besides ITIL, there are several other IT service management standards and frameworks that address the need for monitoring and measuring service management and provide their own set of metrics. Control Objectives for Information and related Technology (COBIT) framework is designed for IT governance purposes [5]. COBIT provides both process metrics and maturity level metrics for each delivery and support (DS) process, such as DS8 Manage Service Desk and Incidents and Manage Problems. Examples of metrics include first-line resolution rate, % of incidents reopened, % of problems recorded and tracked, and % of problems that recur (within a time period) by severity.

ISO 20 000 is ITIL-compliant auditable standard for IT service management that consists of two parts: specification for service management [4] and code of practice for service management [21]. One of its requirements is that the organization shall apply suitable methods for monitoring and measurement of the service management processes. ISO 20 000 also requires that the organizations produces reactive reports, proactive reports, and scheduled reports regarding IT service management activities.

However, ISO 20 000 does not tell which metrics should be used to measure the processes. It defines very generic requirements, for example, "the organization shall apply suitable methods for monitoring and, where applicable, measurement of the service management processes". According to ISO 20 000, service reporting should focus on performance against service level targets, non-compliance and issues (SLA and security breaches), workload (volume and resource utilisation), performance reporting on major events (major incidents and changes), trend information and satisfaction analysis [4].

IT organizations have difficulties in deciding what to measure. Even in a small IT organization there are hundreds of measurement targets to choose. Naturally, each measurement target requires a different type of measurement approach. Software quality metrics can be divided, for example, into efficiency metrics (transaction time), correctness metrics (complexity, MTBF), reliability metrics (down times), and maintainability metrics (number of modules, number of errors per unit) [22]. IEEE Standard Dictionary of Measures to Produce Reliable Software [23] divides metrics simply into process metrics and product metrics where product measures are applied to software objects and process measures are applied to the activities of software development, test and maintenance. Marik, Kral and Marik [24] have examined software validation and verification metrics from the testing viewpoint. Testing-related metrics can be used in the release testing activity of the IT service release management process [25].

We propose that there are three key issues that IT service support providers should measure. First, they should measure the performance of any IT service management process, such as resolution times and volumes for service incidents, problems and change requests categorized by customers, business priority [26], request type etc. A very basic software engineering metric that measures the quality of the software or service is the number of errors that relate to a configuration item.

Second, the IT service provider organization should measure the maturity of IT service management processes or an IT organization, for example, using the process maturity model of the Control Objectives for Information and related Technology (COBIT) [5], Capability Maturity Model Integration (CMMI) for Services [6] that is an extension to CMM model [27] or other maturity assessment models, such as self-assessment model of IT Service Management Forum [28], software maintenance maturity model [29] or corrective maintenance maturity model [30]. Third, it is very important to know the customer satisfaction rate on IT services and processes. Other metrics can be implemented after the three first metrics have been introduced, for example, service business performance metrics including the costs of service unavailability.

There seems to be a clear need for better IT service management measurement frameworks that would define the goals, roles, activities and would be easy to adopt in practice. Additionally, support process managers could use the practical measurement examples of some existing software quality measurement frameworks, such as the Defect Management Framework of Quality Assurance Institute [31], the Software Quality Measurement a Framework for Counting Problems and Defects [32] and Personal Software Process [33].

This case study is a part of the results of KISMET (Keys to IT Service Management and Effective Transition of Services) project [34] and MaISSI (Managing IT Services and Service Implementation) project [35] at the University of Eastern Finland, School of Computing, Software Engineering Research Group, Finland. The KISMET project focuses on improving IT service transition processes (change, configuration and release management) while the emphasis of MaISSI project work was in the service support (service desk, incident and problem management).

The work in our research project has been divided into eight subprojects (MaISSI pilot projects). Improving the IT service management measurement was one of the pilot projects and it was carried out during years 2008 - 2009. The measurement framework was created later in 2010 during KISMET project.

The main contribution of this paper is to:

- describe the implementation process of the ITIL-based IT service management measurement system (ITSM-MS),
- describe the system architecture and the main functions of the ITSM-MS,
- propose a framework for measuring IT services and
- present the lessons learnt from the implementation process.

III. Research Questions & Methodology

The main research problem of this study is: how the measurement of the IT service support processes can be improved? Measurement and continous improvement are hot topics in the IT service provider companies at the moment. There is an urgent need both for a systematic IT service measurement process and for easy-to-use, dynamic measurement tools that enables effective and efficient performance reporting. This study focused on tool improvement providing valuable information on a unique case where a dynamic, real-time measurement tool was implemented for measuring IT service processes.

The measurement challenge was addressed by the application service manager of the case organization. Both constructive methods and case study methods [36] with action research features were used in this study. Constructive methods were used to build the measurement system and the measurement framework. A case study method is "an empirical inquiry that investigates a contemporary phenomenon within its real-life context, especially when the boundaries between the phenomenon and the context are not clearly evident" [36]. Eisenhardt reports that a case study is "a research strategy which focuses on understanding the dynamics present with single settings" [37]. The main goal of the case study method was to analyze the current state of the customer support in the case organization. Action research methods [38] were used in the design and implementation meetings. Researchers were active participants in the implementation process.

A. Case Organization and Data Collection Methods

Our case organization is a medium size business unit (around 120 employees) of a bigger organization that is one of the leading IT service companies in Northern Europe with over 16 000 employees. The company provides IT, R&D and consulting services for various industries, such as banking and insurance, energy, telecom and media, and healthcare. Our pilot was implemented together with the case organization and the MaISSI research team. Improvement of measurement activities was considered as a very important improvement target in the case organization and was selected to the main topic for the pilot project.

IT service support processes, such as incident management, problem management are part of the case organization's business framework WayToExcellence (W2E). Service desk acts as a single point of contact for customers and users. The service desk is an extended version of the help desk. While the help desk focuses mainly on resolving incidents (software and hardware failures), the service desk provides a wider range of services. Besides resolving incidents by using various knowledge repositories, the service desk can handle service requests, license issues, change requests etc. The service desk acts as a Single Point of Contact (SPOC) for customers and users and records each contact to the incident database.

The service desk assigns the incident to the back office (second-line support) which in turn assigns the case to the product support team, if necessary. The back office is responsible for managing and resolving the service requests (for example, handling requests for database queries). The first level support and back office can escalate incidents to the 3rd-level teams (product development) if program code fixes are needed.

The organization uses a java-based tool for handling all the incidents. The incident management tool supports the ITIL-based service management processes. The organization has been quite satisfied with the tool functions and its configuration options. However, they stated that producing measurement reports regarding the performance of service support processes is not effective and a lot of manual work is related to producing those reports to customers. Every month 1-2 days were spent by service managers to generate service performance reports with MS Excel.

The pilot project between the MaISSI research team and the case organization was carried out between years 2008 and 2009. The main goal of the pilot project was to establish a system that enables better measurement of IT service support processes. The process included the following steps:

- 8 August 2008: The kickoff meeting of the pilot project.
- 1 September 2008: The 1st requirement specification meeting.
- 6 October 2008: The 2nd requirement specification meeting.
- 23 October 2008: The 3rd requirement specification meeting.
- October - November 2008: The design phase of the ITSM-MS.
- January - April 2009: The implementation phase of the system.

- 9 April 2009: The final review meeting of the system.
- April - May 2009: Trainings in the case organization.
- May 2009 - July 2009: Initialization of the system.

The case organization was selected from the pool of MaISSI research project's industrial partners. The research team had had cooperation with the case organization in earlier projects. The role of the MaISSI research team (a project manager, a research assistant) was to help the case organization in the design and the implementation of the measurement system. Figure 1 describes the context of this study.

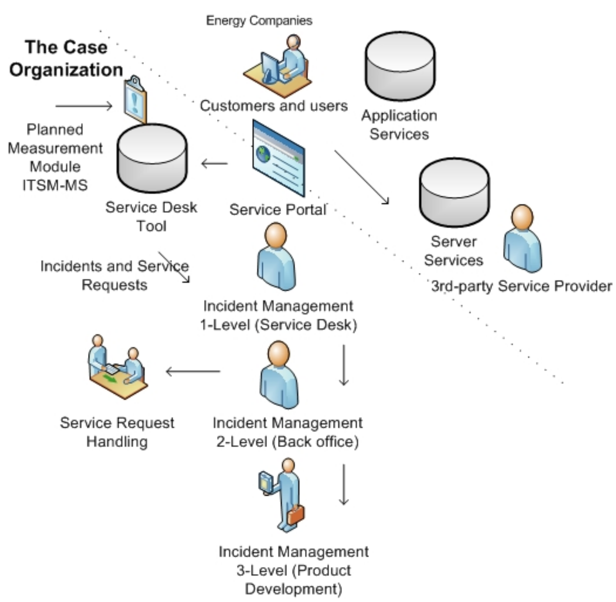

Figure 1. The case study context

The following data collection methods were used in the study:

- Participative observation (field visits and ITSM-MS work meetings)
- Interviews and discussions (application service manager, IT service manager, technical specialists during the implementation phase)
- Internal documentation (a service desk tool user guide, data on existing metrics)
- Access to the case organization, intranet and the service desk system

The MaISSI research team had access to the case organization's facilities as well as to the service desk system. The organization provided workstations to the research team.

B. Data Analysis Method

In data analysis, we used a within-case analysis method that examines a case carefully as a stand-alone entity [37]. The case study database (a Windows folder with access to MaISSI team) was created to ensure the traceability between data sources, meetings and findings. The case study database included memos from meetings with a case organization, internal documents received from the case organization regarding the measurement of IT service support processes, and design and implementation documents created by the MaISSI research team.

IV. Implementing an IT Service Measurement System

In this section, we will introduce how the ITSM-MS was established between the case organization and the MaISSI research team. The work was divided into six phases: kickoff, requirements specification, design, implementation, training and introduction and learning.

A. Kickoff Phase

The kickoff meeting of the pilot project was arranged in 8th August 2008. In that meeting, the representatives (the application service manager and the IT service manager) of the case organization reported that they have problems in measurement and reporting activities regarding IT service support processes: Creating process performance reports with the current tools is mainly manual work and takes too much time from managers. They showed the research team one of the existing process performance reports: the relation of open incidents to closed incidents per day and stated that the organization needs more that type of reports that are easy to understand and provide important information for the business decision makers.

Additionally, the goals for the pilot project were specified. The main goal of the pilot project was to help the case organization in the development and introduction of IT service support metrics. The application service manager and the IT service manager stated that it would be nice to implement a couple of simple process metrics with modern web technologies. The task would mean in practice implementing a dynamic SQL query that combines many search parametres together such as customers, products, priority, and case type.

B. Requirements Specification Phase

Few weeks after the kickoff meeting, the first requirement specification meeting for the ITSM-MS was arranged and the first requirements for the system were defined. Requirements were gathered together with the application service manager and the IT service manager of the case organization. The following general requirements were identified:

- The system must provide real-time measurement information about IT service support processes
- The system must be easy to use (produced measurement reports must be clear and easy to understand).
- The system must support ITIL-based IT service support processes, especially incident management and problem management.

- The system must provide a dynamic user interface (for example the user can select different values for producing graphs).
- The system must be implemented with the technology provided by the case organization.

Additionally, the MaISSI research team introduced some examples of IT service support metrics based on ITIL version 2, ITIL version 3, and an ITIL-based IT service management metrics book [39]. In the 2nd and the 3rd requirement specification meetings, the requirements for the ITSM-MS were clarified. The concepts in the requirements specification were based on the case organization's service management framework and the ITIL framework.

C. Design Phase

The design phase of the system was carried out during October - November 2008 and it was divided into two stages. The first stage (October 2008) was performed as an action research where one researcher from the MaISSI project team was working intensively in the case organization. The researcher took part in the case organization's IT service management team and collected data from the incident management tool.

The second stage (November 2008) aimed at preparation of the system specification based on the data that was collected earlier. For example, the following measurement targets were identified:

- Throughput times
  - by products
  - by customers
  - by request types
  - by urgency
- Request volumes
  - by products
  - by customers
  - by priority level

Additionally, the MaISSI project team defined the process metrics that could be created with the new measurement system. Metrics were defined based on ITIL service support metrics. Draft versions of metrics were designed with MS Excel.

- Number of new and closed incidents by priority level
- Number of new and closed problems by products
- Number of new change requests by a customer and a product
- Number of all service desk cases
- Incident throughput time by priority level
- Average incident resolution time

During the second stage, a drop-down menus for the user interface of the ITSM-MS were designed by identifying search parametres and their values (see Figure 2).

In this study, the ITSM-MS was constructed for the internal use of the case organization and its employees. The system was targeted to process managers, product managers and project managers to work as a real-time measurement tool that is connected to every IT service support process.

Incident Creator	Priority	Customer
All	All	All
Customer	1 - Urgent	Specified
Case Organization	2 - High	
	3 - Medium	
	4 - Low	

Figure 2. Design view of the parametres

Figure 3 shows the general system architecture of the ITSM-MS. The service desk (SD) of the case organization receives an incident from the customer, a ticket is entered into the incident management tool and a new case is opened.

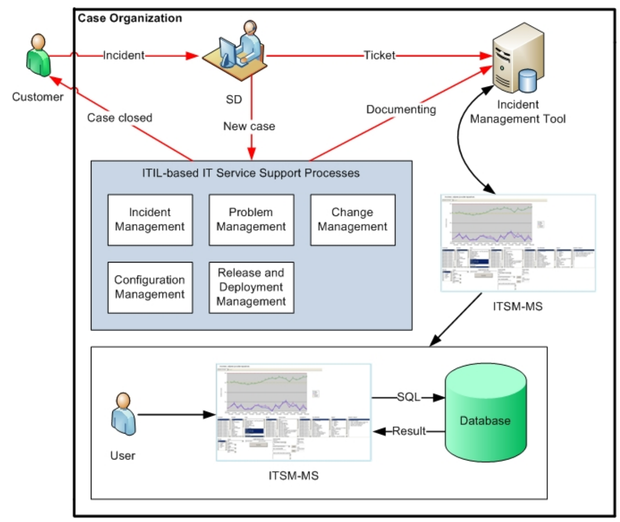

Figure 3. The system architecture

The case organization uses ITIL-based IT service support processes for resolving the case and all the data that is used during these processes is documented into the incident management tool. When the case is resolved, it is sent back to the customer and the case is closed.

As a result of the design phase, the system specification was documented and reviewed at 24th November 2008.

D. Implementation Phase

The implementation phase of the ITSM-MS was executed in January - April 2009. This phase was also performed as an action research where a member of the MaISSI research project team worked in the case organization's facilities. Based on the system specification, the ITSM-MS was implemented with the help of the case organization and its employees.

From the technical point of view, the ITSM-MS was implemented using Microsoft .NET environment, Microsoft Visual Studio, C# programming language and Oracle database. The ITSM-MS uses the database of the incident management tool. Microsoft Visual Studio was used to implement the user interface and the system functions while Oracle database stores all the data from the IT service support processes.

The ITSM-MS uses the data that is documented in the incident management tool. The user can use the system from the case organization's Intranet and make real-time graphs about different cases. The user's input is transformed into an SQL-query which returns a result from the database. Based on this result, the ITSM-MS draws a graph of the metric into the user's computer screen in the Intranet (see Figure 4).

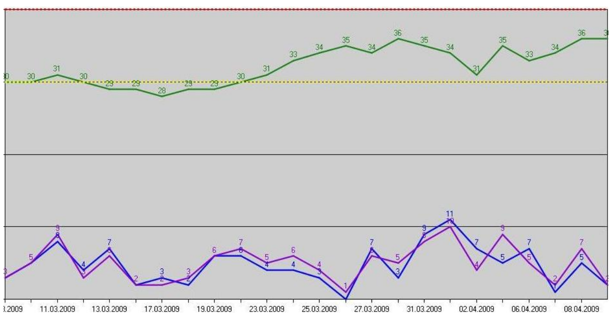

Figure 4. The process performance graph.

The user interface of the ITSM-MS contains different parametres and functions that the user must select for generating the graph and saving the user profile (see Figure 5).

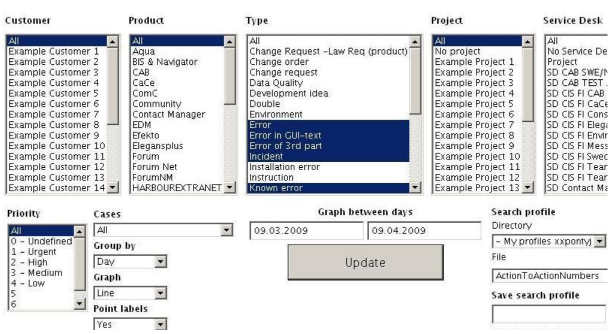

Figure 5. Setting parametres in the ITSM-MS.

The user profile is later used for generating the graph based on saved values and functions. In that way, the user does not have to enter all the values again for the graph. The user can select a specified value or multiple values depending on what type of graph is needed to represent.

- Customer: All customers of the case organization.
- Product: All products of the case organization.
- Type: Different types of cases (for example incident, problem, change request or known error).
- Project: All projects of the case organization
- Service Desk: SD of the case organization.
- Assigned to: The person who is responsible of the case.
- Classification: All cases are classified (for example error in program, hardware problem or error in documentation).
- Impact: The impact of the case (for example standard impact, major impact or note).
- Business impact: How case affects the service or the business (for example no business impact and minor error or esthetic).
- Priority: Priorities of cases: 0 - undefined, 1 - urgent, 2 - high, 3 - medium and 4 - low.
- Cases: Some particular cases.
- Group by: Days, weeks, years.
- Graph: The graph of the metric can be presented as a line graph or a bar graph.
- Point labels:Yes or no.
- Graph between days: The starting point and the ending point for the graph.
- Search profile: Search for a particular saved user profile.
- Save search profile: Save the current user profile.
- Target limits: This is used for the traffic lights (explained later in this paper).

The final review meeting of the system was arranged in 9th April 2009. Participants at that meeting were the application service manager and the IT service manager of the case organization and the MaISSI research project team. After this, the system was deployed to the operational environment.

E. Training and Introduction Phase

Trainings for the employees were carried out during April - May 2009 by the application service manager of the case organization. The user of the ITSM-MS can select target limits for the metric and save them to a user profile. Limits are used to show the current situation of different cases by traffic lights that are showed in Figure 6.

The pointer on the green section means that the situation is fine and there are not so many unsolved cases. The pointer on the yellow section means that the situation needs more attention and there are few unsolved or non-closed cases to resolve.

Finally, the pointer on the red section means that there are too many unsolved cases in the incident database. This means that the case organization must take action for resolving those cases and turn the pointer from the red section back to the green section. Numbers below the traffic lights describe cases that are currently open and the change on previous measuring period, for example last month. Numbers above the traffic lights are the target limits.

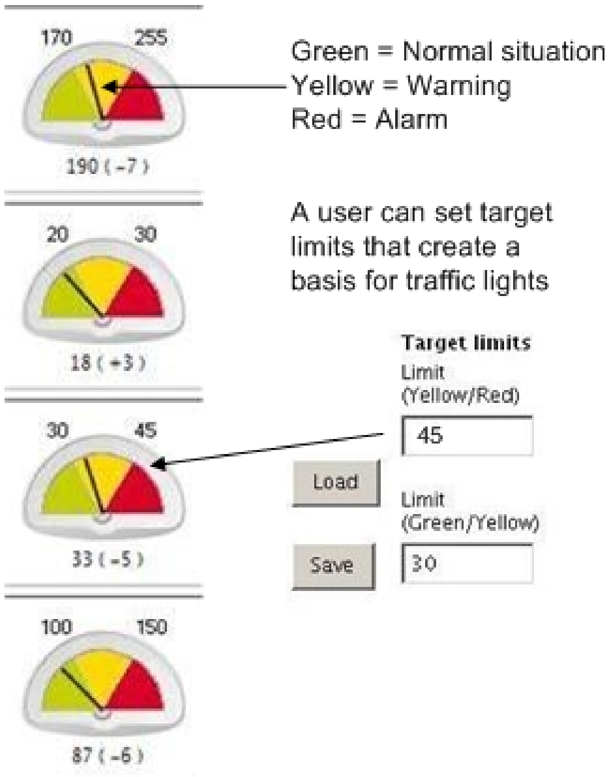

Figure 6. The traffic lights report

The ITSM-MS can also be used as a miniature version besides of the full version that was showed earlier in this paper. Figure 7 presents the mini window of the ITSM-MS system where a saved user profile is showed on the screen. The miniature version enables users to create measurement reports from a company's intranet page.

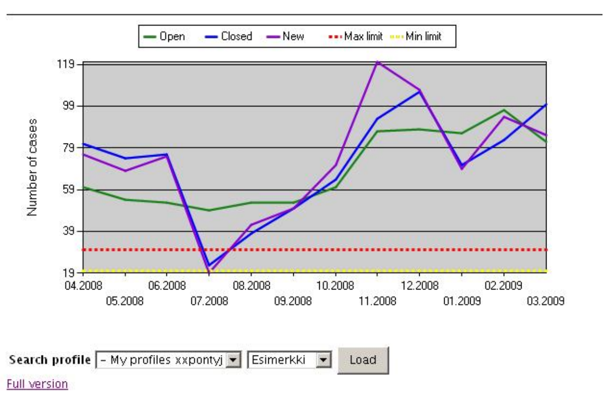

Figure 7. The mini window of the ITSM-MS.

The mini window can show all the saved user profiles that the user has created. Thus, the user does not have to open the full version of the ITSM-MS, if the required graph is saved on a profile. The mini window enables the faster use of the ITSM-MS. After trainings, the initialization phase of the system was performed and now the ITSM-MS is currently in use in the case organization.

F. Learning Phase

During the design and implementation of the IT service management measurement system, many things were learned. First, besides the well-designed and easy-to-use measurement tool, an IT service provider organization must have a systematic measurement process that defines why, how, when, to whom metrics and reports are generated.

Second, metrics should be based on the business objectives. The linkage between metrics and business objectives can be built by using critical success factors, key performance indicators and metrics. Thus, we can measure the things that are related to business strategies. Third, IT service providers should invest in how to use the collected measurement data. The data can be used to identify trends and deviations/exceptions in service quality. Information on trends can be used as an input for other service management processes, such as problem management (proactive problem management) and continuous service improvement.

It is hard to calculate what was the exact cost of building the ITSM -MS system. The work effort consisted of a design phase (1 month) carried out by a university researcher, an implementation phase (two months) by a computer science student, and supervision hours given by a project manager.

As a part of the learning phase, we created a process framework for IT service management measurement (see Figure 8) based on the 7-step improvement model of the ITIL Continual Service Improvement book.

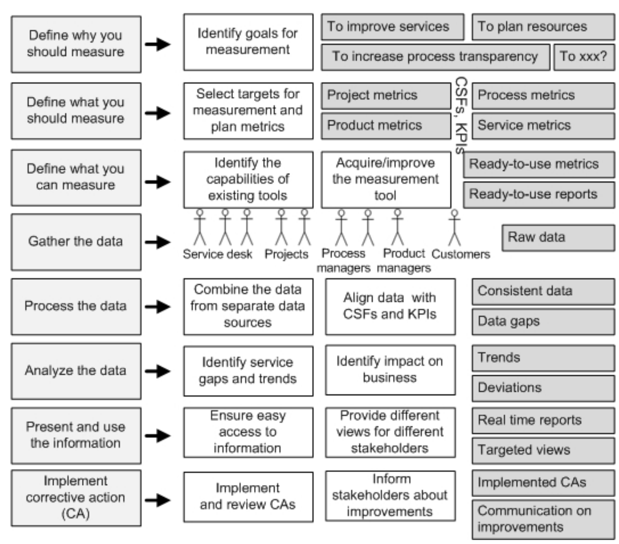

Figure 8. The measurement framework for IT service management.

Our measurement model is based on the following principles:

- Metrics should be linked to business objectives by using three elements (example given in parentheses):
  - Critical Success Factor (Quickly resolve incidents)
  - Key Performance Indicator (Percentage reduction in average time to respond a service desk call)
  - Metric (Call response time in seconds)
- Each phase of the measurement process should have clearly defined outputs. Outputs are marked with dark grey colour in our model.
- The measurement system should enable rapid decision making for business managers. Instead of complex reports one should use "traffic light" reporting or set maximum/minimum limits to charts.
- The measurement system should enable real time reporting instead of pdf reports. There is always a gap between time when a report is created and time when a report is read.
- Targeted measurement reports should be provided to stakeholders. Because stakeholders' needs for reporting vary a lot, one should provide them a possibility to create reports by themselves.

This study was focused on the implementation of the measurement system. The above presented measurement framework has not yet been validated with the case organization but it is presented here as an outcome of the learning process.

V. Analysis

The IT service management measurement system created in the MaISSI project provides several benefits for the case organization's IT service support. We analyzed the benefits by comparing the state of measurement in the case organization after the ITSM-MS vs. situation before ITSM-MS.

- The work effort required to produce performance reports: 15-20 minutes vs. 1-2 days per person
- The format of reports: dynamic real-time reports vs. fixed pdf reports
- Usability of the report tool: good, in most cases does not require additional training vs. difficult to use in many points.
- Sharing and use of measurement reports: dynamic reports accessible from websites vs. reports must be created from a service desk tool.

First, it remarkably decreased the amount of manual work in measurement. Instead of 1-2 days, the service performance reporting takes now 15-20 minutes. No Excel-based reporting is needed anymore. Second, it provides real-time reports about the process performance enabling faster business decision making regarding service support issues. Third, the usability of report/chart generator is better in the new system than in the existing service desk system. For example, the system can show the id numbers of service desk cases that have been open for a long time. Thus, process manager is able to click the id number and the system shall show the details of the service desk case that requires more attention.

Finally, the system enables an effective knowledge sharing of measurement data to appropriate stakeholders. Measurement queries can be installed into organization's intranet pages and charts can be displayed even as a personal screen-saver. The case organization is also planning to provide the reports from the ITSM-MS to customers in near future. Regarding the limitations of our system, the ISM-MS is an add-on module and cannot work without a database of a service desk tool. Additionally, there is no installation software for ITSM-MS available. Thus, the system might be difficult to transfer to new environment.

The following lessons learnt were identified during the implementation of the IT service management measurement system. The implementation phase that the lesson is based on is coded as follows: K=Kick off; R=Requirements specification; D=Design; I=Implementation; T=Training and introduction. The coding of lessons learnt helped us to maintain a chain of evidence between data sources and results.

- The incident management process is a good area to start the IT service management measurement activities (K). If the organization has a help desk or a service desk, it should likely have already several metrics that provide easy start for the improvement of measurement activities. In case of other support processes it might be that there are no existing metrics.
- Create a sense of urgency for the implementation of the measurement system (K) if people need motivating for measurement . In our case, the level of urgency was quite low. However, the improvement of reporting and measurement methods were considered actual internal development target. Perhaps more urgency for the measurement system would have given if the customers had been unsatisfied with the process performance reports.
- Time-based metrics are more difficult to implement than volume-based metrics (R,D). Calculating the resolution time was harder than expected. It should take account in the holidays, weekends and other times when support engineers are not resolving cases.
- Keep measurement reports as simple as possible (T). Although we started from really simple metrics, we found that it was difficult to interpret graphs that somebody else had done with the ITSM-MS.
- Even the top-quality measurement system is dependent on the quality of the collected data (D,I,T). We noticed that our system showed in the drop-down menu all the request types (around fifteen) that were created in the service desk tool. If teams do not have unified classification rules, measurement reports do not give reliable results.

- Selecting appropriate metrics for IT service support is difficult (R). IT service management frameworks have introduced dozens of metrics for one process. However, the frameworks do not tell which metrics are obligatory and which are more like nice-to-have requirements. We recommend Top3 approach for defining metrics. In the Top3 approach, IT organization selects, describes and implements three most important metrics for each IT service support process. Other metrics are considered as nice-to-have metrics.
- The IT organization needs a systematic measurement approach in addition to the measurement system (K,R). However, we observed in this study that the IT service management frameworks seem to lack a clear measurement and reporting process. Our initial goal was to define the measurement process for the case organization before the measurement tool but we failed to achieve this goal.
- Managers love traffic lights (I,T). Traffic lights were perhaps one of the best functions of the system. Simple colour-coded function provides business managers or process managers a rapid overview what is happening in the support and maintenance and where are the pain areas.
- Focus in the early phase of the measurement project how to deploy and use the collected measurement data instead of solely thinking how to create reports (T). In our system, the user does not have to open the full version of the ITSM-MS, if the required graph is saved on a user profile. This enables easy access to the measurement data and likely increases the system usage.
- Implementation of a measurement system does not require a large development team (I). In our project, the implementation of the ITSM-MS was carried out by one person that was a very good programmer. The system design document was created by a MaISSI researcher.

Both the research team and the case organization considered the pilot project and its main result, IT service management measurement system, as a success. After the implementation of the measurement system the cooperation between MaISSI and the case organization continued as another pilot project the goals of which was to increase the transparency of the organization's service support processes to customers and to improve the release management process. The ITSM-MS can be used to provide measurement information regarding all the support processes: incident management, problem management, change management, release management and configuration management.

VI. Conclusion

This study aimed to answer the following research problem: How the measurement of the IT service support processes can be improved? The main contribution of this study was to 1) describe the implementation process of the ITIL-based IT service management measurement system (ITSM-MS), 2) describe the system architecture and the main functions of the ITSM-MS, 3) propose a framework for measuring IT services and 4) present the lessons learnt from the implementation process.

In this paper, we described the four phases of the implementation process: requirements specification, design, implementation, training and introduction. During these phases we identified the functional and data requirements for the measurement system, described the high-level system architecture, and explained how the system works in the practice. Additionally, we presented ten most important lessons learnt from the implementation process.

Data for this study were collected using case study methods and action research methods. Additionally, constructive methods were used in designing and implementing the prototype of the ITSM-MS. IT service management measurement system was implemented together with a business unit of the large IT service provider organization in Finland. We have received very positive feedback from the case organization regarding the system. The system is considered very useful in the case organization and it has remarkably decreased the amount of manual work in creating process performance reports.

There are several limitations to this study. First, data were collected from one IT service provider company within a relatively short time period. The customers of the case organization did not participate in this study because the improvement focus was on the internal perspective. We also used three important principles of data collection [36] to increase the quality of our study 1) use multiple sources of evidence, 2) create a case study database, and 3) maintain a chain of evidence between data sources and results.

Second, we did not focus much on other non-functional requirements than usability and the fact that the system must work real-time. Third, because the case organization was selected from the partner pool of the MaISSI project, the selection method was the convenience sampling method. Finally, the case study does not allow us to generalize our research results. However, we can use our results to expand the theory of IT service management measurement.

Further research is needed to examine establishment of service support measurement systems in IT service companies and implementation of a systematic measurement process.

Acknowledgment

This paper is based on research in KISMET (Keys to IT Service Management and Effective Transition of Services) project and MaISSI (Managing IT Services and Service Implementation) project funded by the National Technology

Agency TEKES, European Regional Development Fund (ERDF), and industrial partners.

REFERENCES

[1] A. Lahtela, M. Jäntti, and J. Kaukola, "Implementing an itil-based it service management measurement system," in *Proceedings of the 4th International Conference on Digital Society*. St. Maarten, Netherlands Antilles: IEEE Computer Society, February 2010, pp. 249–254.

[2] IEEE Standard 1061-1998, *IEEE Standard for a Software Quality Metrics Methodology*. IEEE, 1998.

[3] Office of Government Commerce, *ITIL Service Lifecycle - Introduction*. The Stationary Office, UK, 2007.

[4] ISOIEC20000, *IT Service Management, Part 1: Specification for service management*. ISO/IEC JTC1/SC7 Secretariat, 2005.

[5] COBIT 4.1, *Control Objectives for Information and related Technology: COBIT 4.1*. IT Governance Institute, 2007.

[6] F. Niessinka, V. Clerca, T. Tijdinka, and H. van Vliet, "The it service capability maturity model version 1.0," CIBIT Consultants&Vrije Universiteit, 2005.

[7] Microsoft, "Microsoft operations framework," http://technet.microsoft.com/en-us/library/cc506049.aspx, July 2010.

[8] J. Caldeira and F. B. e Abreu, "Influential factors on incident management: Lessons learned from a large sample of products in operation," in *Product-Focused Software Process Improvement*, A. Jedlitschka and O. Salo, Eds., vol. 5089. Springer Verlag, 6 2008, pp. 330–344.

[9] M. Kajko-Mattsson, C. Ahnlund, and E. Lundberg, "Cm3: Service level agreement," in *ICSM '04: Proceedings of the 20th IEEE International Conference on Software Maintenance*. Washington, DC, USA: IEEE Computer Society, 2004, pp. 432–436.

[10] OGC, *ITIL Service Design*. The Stationary Office, UK, 2007.

[11] Office of Government Commerce, *ITIL Continual Service Improvement*. The Stationary Office, UK, 2007.

[12] R. Morgan and L. Ho, "Six sigma for it service management," Six Sigma Zone Report, 2004.

[13] M. Jäntti, A. Miettinen, and K. Vähäkainu, "A checklist for evaluating the software problem management model: a case study," in *SE'07: Proceedings of the 25th conference on IASTED International Multi-Conference*. Anaheim, CA, USA: ACTA Press, 2007, pp. 7–12.

[14] OGC, *ITIL Service Support*. The Stationary Office, UK, 2002.

[15] M. Kajko-Mattsson, "Maturity status within front-end support organisations," in *ICSE '07: Proceedings of the 29th international conference on Software Engineering*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 652–663.

[16] Office of Government Commerce, *ITIL Service Operation*. The Stationary Office, UK, 2007.

[17] K. H. Bennett and V. T. Rajlich, "Software maintenance and evolution: a roadmap," in *ICSE '00: Proceedings of the Conference on The Future of Software Engineering*. New York, NY, USA: ACM Press, 2000, pp. 73–87.

[18] B. P. Lientz and E. B. Swanson, *Software Maintenance Management*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1980.

[19] F. Niessink and H. van Vliet, "Software maintenance from a service perspective," *Journal of Software Maintenance*, vol. 12, no. 2, pp. 103–120, March/April 2000.

[20] M. Jäntti, "Difficulties in managing software problems and defects," Ph.D. dissertation, University of Kuopio, 2008.

[21] ISOIEC20000b, *IT Service Management, Part 2: Code of practice for service management*. ISO/IEC JTC1/SC7 Secretariat, 2005.

[22] E. Wallmueller, *Software quality assurance: A practical approach*. Prentice Hall International, 1994.

[23] IEEE Standard 982.1-1988, *IEEE Standard Dictionary of Measures to Produce Reliable Software*. IEEE, 1988.

[24] V. Marik, L. Kral, and R. Marik, "Software testing & diagnostics: Theory & practice," in *SOFSEM '00: Proceedings of the 27th Conference on Current Trends in Theory and Practice of Informatics*. London, UK: Springer-Verlag, 2000, pp. 88–114.

[25] Office of Government Commerce, *ITIL Service Transition*. The Stationary Office, UK, 2007.

[26] C. Bartolini and M. Salle, "Business driven prioritization of service incidents," in *Utility Computing*. Springer Berlin / Heidelberg, 2004, vol. 3278, pp. 64–75.

[27] P. Jalote, *CMM in Practice, Processes for Executing Software Projects at Infosys*. Addison-Wesley, 2000.

[28] IT Service Management Forum, "Itil service management self assessment," http://www.itsmfi.org/content/self-assessment-itil-v2, 7 2010.

[29] A. April, J. H. Hayes, A. Abran, and R. Dumke, "Software maintenance maturity model (smmm): the software maintenance process model: Research articles," *J. Softw. Maint. Evol.*, vol. 17, no. 3, pp. 197–223, 2005.

[30] M. Kajko-Mattsson, "Corrective maintenance maturity model: Problem management," in *ICSM '02: Proceedings of the International Conference on Software Maintenance (ICSM'02)*. Washington, DC, USA: IEEE Computer Society, 2002, p. 486.

[31] Quality Assurance Institute, "A software defect management process," Research Report number 8, 1995.

[32] W. Florac, "Software quality measurement a framework for counting problems and defects," Technical Report CMU/SEI-92-TR-22, 1992.

[33] I. Hirmanpour and J. Schofield, "Defect management through the personal software process," *Crosstalk, The Journal of Defense Software Engineering*, 2003.

[34] KISMET, "Kismet website," http://www.uef.fi/KISMET/, access: 18.1.2011.

[35] MaISSI, "Maissi website," http://www.uef.fi/maissi/, access: 18.1.2011.

[36] R. Yin, *Case Study Research: Design and Methods*. Beverly Hills, CA: Sage Publishing, 1994.

[37] K. Eisenhardt, "Building theories from case study research," *Academy of Management Review*, vol. 14, pp. 532–550, 1989.

[38] I. Benbasat, D. K. Goldstein, and M. Mead, "The case research strategy in studies of information systems," *MIS Q.*, vol. 11, no. 3, pp. 369–386, 1987.

[39] IT Service Management Forum, *Metrics for IT Service Management*. Van Haren Publishing, 2006.