Towards Mature IT Services*

Frank Niessink and Hans van Vliet

Faculty of Mathematics and Computer Science, Vrije Universiteit Amsterdam
De Boelelaan 1081, 1081 HV, Amsterdam, The Netherlands
Tel: +31 20 444 7781, Fax: +31 20 444 7653
E-mail: {F.Niessink, J.C.van.Vliet}@cs.vu.nl

Abstract

Many organizations provide information technology services, either to external or internal customers. They maintain software, operate information systems, manage and maintain workstations, networks or mainframes or provide contingency services. A widely used framework on IT service quality (at least in the Netherlands) is the Information Technology Infrastructure Library (ITIL), which seeks to publish a standard of service quality and best practices. However, it does not provide organizations with the methodology needed to assess and improve their service processes based on assessments. We propose an Information Technology Service Capability Maturity Model (IT Service CMM) that can be used to assess the maturity of IT service processes and identify directions for improvement.

This IT Service CMM originates from our efforts to develop a quality improvement framework that was targeted at helping service organizations to improve service quality. Case studies which introduced parts of our framework into different organizations had mixed results. A service process capability maturity scale, similar to the Software CMM, allowed us to explain many of the differences observed. At the same time, the IT Service CMM suggests ways to improve the service process capability.

Keywords

Process Improvement, Capability Maturity Model, IT Services.

1 Introduction

Many organizations provide information technology services, either to external or internal customers. They maintain software, operate information systems, manage and maintain workstations, networks or mainframes, or provide contingency services. An important question is how these services should be defined and managed. The complexity of IT applications makes it difficult to properly tune customer requirements and service provider capabilities. Customers often cannot express their real service requirements and do not know the corresponding performance needs. Likewise, service providers often do not know how to differentiate between IT services and how to attune them to a specific customer.

*This is a preprint of an article accepted for publication in *Software Process – Improvement and Practice*, volume 4, issue 2, June 1998, pp. 55-71.

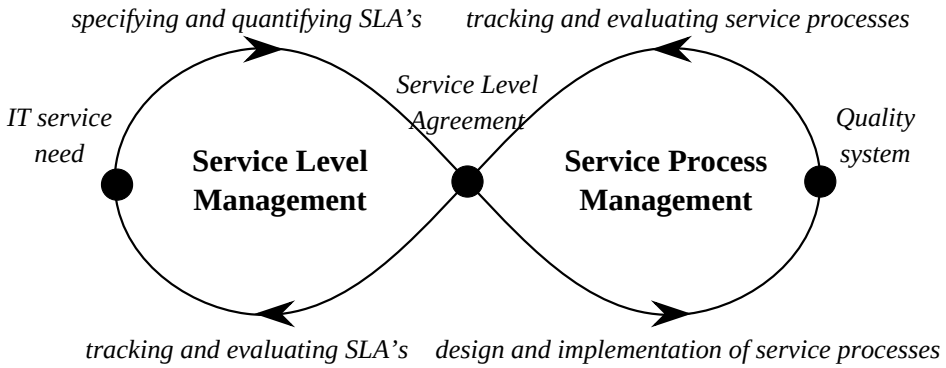

Figure 1: IT Service Lemniscate

We are involved in a multi-partner research effort that aims to develop a method to specify and control IT services. The process model used to describe the dynamics of IT service management is depicted in figure 1 [25]. The left part of the lemniscate concerns the specification of IT services (upper arrow) and the evaluation and monitoring of the performance of the service provider (lower arrow). The right part concerns the evaluation and monitoring of service processes (upper arrow) and the design and organization of those processes. The service level agreement (SLA) plays a pivotal role in this scheme.

In the course of this research, a number of case studies were done to introduce and test parts of this framework in different organizations. These case studies had mixed results. We observed that some service providers were more mature as regards their service capabilities than others. Based on these experiences, we propose an IT Service CMM, similar in form and scope to the Software CMM. This IT Service CMM allows us to explain many of the differences observed.

A service provider will traverse the lemniscate of figure 1 a number of times. Based on experiences with customer needs and the services provided to fulfill those needs, the service provider may try to improve its service quality. Our IT Service CMM not only allows us to assess the maturity of service processes, but identifies directions for improvement as well. The combination of the IT Service CMM and the service management process model of figure 1 then results in a quality improvement framework for IT services.

This paper is organized as follows. Section 2 discusses the literature with respect to quality improvement in IT services and process improvement. Section 3 discusses a number of case studies that we did. Section 4 presents the IT Service CMM, and section 5 summarizes the results so far.

2 Related developments

In this section, we look at some of the work that has been done in the area of process improvement. See [22] for an overview of the many models, frameworks, and standards that exist in this area. We first look at the Software CMM, which is the oldest and by far best known software process improvement framework. Next, in section 2.2, we turn to Trillium as an example of a software process improvement framework that pays explicit attention to service issues. Finally, in section 2.3, we describe the IT Infrastructure Library, which is a set of best practices targeted at IT services.

2.1 Software CMM

The Software Capability Maturity Model[1] [5], developed by the Software Engineering Institute (SEI) of the Carnegie Mellon University, is probably the best known software process improvement method. Many newer software process improvement methods, such as BOOTSTRAP [19], Trillium [26], and ISO/IEC 15504 (formerly known as SPICE) [23, 12] have been largely based on, or are extensions of, the Software CMM. Also, different variants of the Software CMM have been developed, for example the People CMM [6, 7].

The Software CMM measures a software organization's software process capability on a five-level ordinal scale. The software process capability is defined as the range of expected results that can be achieved by following a software process [5, p. 9]. The model distinguishes the following five maturity levels:

1. *Initial*: The software process is characterized as ad hoc, and occasionally even chaotic. Few processes are defined, and success depends on individual effort and heroics.

2. *Repeatable*: Basic project management processes are established to track cost, schedule and functionality. The necessary process discipline is in place to repeat earlier successes on projects with similar applications.

3. *Defined*: The software process for both management and engineering activities is documented, standardized, and integrated into a standard software process for the organization. All projects use an approved, tailored version of the organization's standard software process for developing and maintaining software.

4. *Managed*: Detailed measures of the software process and product quality are collected. Both the software process and products are quantitatively understood and controlled.

5. *Optimizing*: Continuous process improvement is enabled by quantitative feedback from the process and from piloting innovative ideas and technologies.

Each maturity level is characterized by a number of processes that an organization residing on that level should perform. These processes are grouped in key process areas, see figure 2. Each key process area consists of activities that an organization needs to implement. These activities are grouped according to their common features. Five common features are distinguished: commitment to perform, ability to perform, activities performed, measurement and analysis, and verifying implementation.

The criteria set forth by the CMM in the form of key process areas can be used as a reference framework for software process improvement. The Software Engineering Institute's process improvement method IDEAL[1] [20] (Initiate, Diagnose, Establish, Act, Leverage) uses the Software CMM as the basis for diagnosing an organization's software process.

The Software Capability Maturity Model has received quite some attention in the literature, both positive and negative. Some authors criticize the lack of a formal theoretical basis for the model, the use of a relatively short questionnaire to assess an organization as a whole, and the lack of empirical support for the claims made by the SEI [1, 3, 13]. Also, the focus of the model on large-scale software development caused difficulties for an organization that applied the model to a maintenance-only software organization [11].

However, empirical support for the CMM is growing, see for example [8, 9, 10, 15, 16]. Empirical results are regularly summarized by the SEI, see [14] for a recent report.

[1] Capability Maturity Model, CMM, and IDEAL are service marks of Carnegie Mellon University.

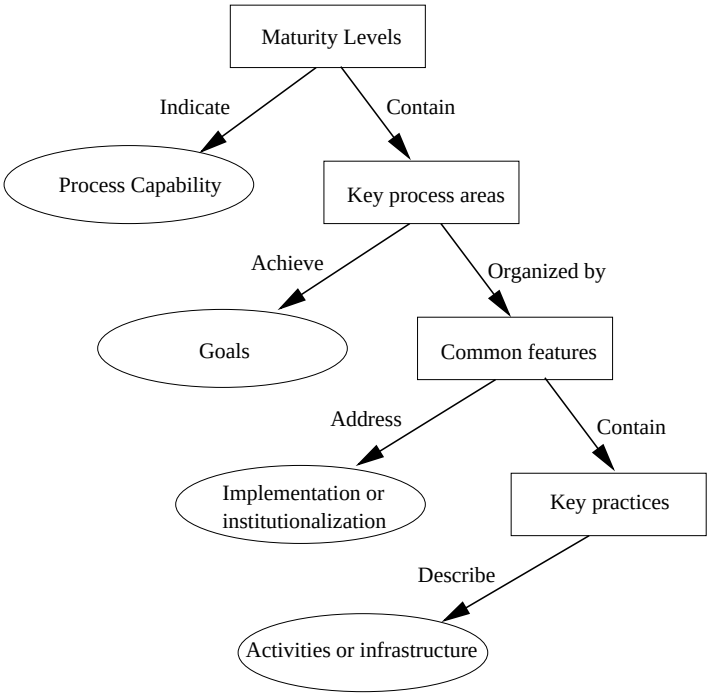

Figure 2: The CMM structure (taken from [5])

2.2 Trillium

Trillium is a telecommunications product development and support capability model, developed by Bell Canada, Northern Telecom and Bell-Northern Research. It is used by Bell Canada to assess the product development and support capability of their existing and potential suppliers [26]. The Trillium model is targeted at improving the capability of development organizations to consistently deliver products or enhancement to products:

- that meet customer requirements;
- with minimal defects;
- for the lowest life-cycle cost, and,
- in the shortest time.

The Trillium model is based on the Software-CMM. In addition it contains aspects of several other sources, for example ISO 9001, ISO 9000-3 and the Malcolm Baldrige National Quality Award. The scope of the model has been widened to include both software and hardware.

The model is divided into eight Capability Areas, see table 1, each of which contains different practices, grouped in roadmaps. As opposed to the key processes of the Software CMM, Trillium roadmaps contain practices on several maturity levels. Thus, as an organization advances to the next maturity level, it will implement the more mature variants of the practices of the different roadmaps.

Contrary to the Software CMM, Trillium explicitly covers support activities in one of its capability areas. The Customer Support Capability Area consists of six roadmaps that cover the problem

Organizational Process Quality
Human Resource Development and Management
Process
Management
Quality System
Development Practices
Development Environment
Customer Support

Table 1: Trillium Capability Areas adapted from [26]

response system, usability engineering, life-cycle cost modeling, user documentation, customer engineering, and user training. Still, these activities are regarded as add-on (after-sales) services to the products supplied by the organization, as opposed to the services themselves being regarded as products. As such, Trillium cannot be easily applied to the IT service industry.

2.3 IT Infrastructure Library

According to [18], the primary objective of the IT Infrastructure Library is 'to establish best practices and a standard of IT service quality that customers should demand and providers should seek to supply.' ITIL was originally developed by the British government through their Central Computer & Telecommunications Agency (CCTA). Nowadays, ITIL is being maintained by the Netherlands IT Examinations Institute (EXIN).

The library consists of several sets of booklets that contain those 'best practices' in IT service delivery. The booklets are divided into nine sets. The first six sets are called the IT service provision and IT infrastructure management sets. The other three are called the Environmental sets. These latter three sets cover the environmental infrastructure for IT, such as the building, cabling and service facilities. We will only look at the IT service provision and IT infrastructure management sets. The six sets cover the following practices (each described in a separate booklet):

- The Service Support set covers configuration management, problem management, change management, help desk, and software control and distribution.
- The Service Delivery set covers service level management, capacity management, contingency planning, availability management, and cost management for IT services.
- The Managers' set deals with managing facilities management and customer liaison.
- The Software Support set describes software life-cycle support and testing an IT service for operational use.
- The Computer Operations set covers computer operations management, unattended operating, third party and single source maintenance, and computer installation and acceptance.
- Finally, the Network set describes the management of local processors and terminals.

Each booklet describes the practices in terms of planning; implementation; audits; benefits, cost and possible problems, and tool support. Attention is given to operational procedures, roles, responsibilities, dependencies, support processes, training, etc.

Although the booklets cover a wide range of issues regarding IT services, there are still a number of important issues that need more attention. Examples are:

- The specification of service level agreements. Although ITIL does promote the use of SLAs, it does not provide much help on how to develop them.
- The use of service catalogs. ITIL does promote the use of a service catalog to facilitate the communication with the customers, but again does not say much about the contents or how to develop it.
- ITIL implementation. ITIL itself does not provide much information on the best way to implement the different processes and on how to decide on the best order of implementation.
- The distinction between service producing processes and service support processes. In our opinion, ITIL does not clearly distinguish between those two types. For example, the ITIL help desk is both used for communication with the end-users (needed for incident handling) and for user support (a service).

While over the years different companies have been selling services that complement ITIL, such as education, training, and consulting on ITIL implementation, ITIL still lacks an overall approach to the improvement of service processes. Improvement is not an integral part of the library.

3 Case studies

In this section we present a number of case studies that were done in the course of our research project. As mentioned in section 1, we use a generic process model (displayed in figure 1) as the basis for our research. Guided by the lemniscate, different research issues have been identified, including the specification of service level agreements, evaluation of service quality, the use of service catalogs and problem management. These issues have been investigated in several case studies that are presented below.

To facilitate the translation of diffuse IT needs of customers into measurable service level agreements (upper-left arrow of the lemniscate) a SLA specification method was developed. Several case studies were performed to evaluate and improve the method. Sections 3.1 and 3.2 describe two of them.

An important question with respect to service level agreements is whether the right levels have been established. In one case study we investigated the use of ServQual to evaluate service level agreements (lower-left arrow). This case is described in section 3.3.

According to ITIL, problem management is an important aspect of service provision. The ITIL Problem Management process aims at minimizing the impact of failures ('incidents') and on correcting root causes of failure. This makes it part of both the upper-right and the lower-right arrows of the IT service lemniscate. In the case study described in section 3.4 we looked at the problem management process.

As mentioned in section 2.3, ITIL advocates the use of a service catalog but does not provide directions on how to implement it. Therefore, we did two case studies on the development of service catalogs, described in sections 3.5 and 3.6.

3.1 Case A – developing a service level agreement

This case study was part of an education improvement program undertaken by a Dutch university. Part of the program is the supply of notebooks to all students, including different services such as end-user support and repair maintenance. The notebooks and services are delivered by a large Dutch IT service provider.

During the case study a service level agreement between the service provider and the university was developed. The SLA specification method was used to derive the needed service levels, taking the students – the end-users – as the starting point. This was the first time that this service provider used the SLA specification method to develop service level agreements, and it was also the first time they delivered this particular service. Despite the lack of experience, the service level agreement was developed according to the method without major problems.

3.2 Case B – developing a generic service level agreement

This case study was held in the information technology department of a large Dutch governmental organization. The study was part of a larger program to implement a quality system in the organization. The case study concerned the introduction of the SLA specification method and the development of a generic service level agreement.

This organization had a quite formal organizational structure, but at the same time this formal structure was being ignored to be able to react to organizational and technical problems. The organization was not used to draw up service level agreements with its customers. Agreements between the department and its customers were in the form of effort obligations, not results. No quality standards, such as ITIL or ISO 9000, were being used.

It seemed to us that this organization was not quite ready for the introduction of generic service level agreements, without first gaining practical experience with the use of result oriented contracts.

3.3 Case C – evaluating service quality

During this case study, the quality of the services delivered by the IT department of a decentralized governmental organization were evaluated. We used ServQual to measure the perceived quality of the IT services by the end-users. ServQual is a measurement method targeted at measuring the quality of services. See [21, 27] for examples of the application of ServQual in measuring IT service quality.

The IT department manages and maintains the IT infrastructure of the local governmental organization and provides end-user support. The department does use service level agreements, but these are mainly used to specify procedures and opening times, and do not address concrete and measurable service levels.

The case study was quite successful: users of the IT services were very well capable of detailing their opinion on the quality of the service provision. Apparently, the evaluation of service quality does not depend on the presence of specified quality levels.

3.4 Case D – incident and problem management

This organization is the IT department of a large organization, responsible for carrying out part of the Dutch social security system. As of the beginning of 1996, the organization has been split into a non-profit public body and a private for-profit organization – part of which is the IT department.

The IT department provides a large number of IT services to its customers, which are mainly departments from the sibling organization. To manage the communication with customers regarding those services, the department has implemented helpdesk management and problem management processes. The implementation of these processes has been based on the Information Technology Infrastructure Library (ITIL). Helpdesk Management is used to guarantee the continuity of services, while Problem Management is used to improve the level of service in the future. So, Helpdesk Management deals with *incidents*, whereas Problem Management is concerned with solving the *problems* that cause these incidents.

The goal of this case study was to assess the quality of the Problem Management process. It soon became apparent that the organization was not able to execute the Problem Management process properly, because the Help Desk Management process did not result in the necessary data needed to adequately analyze and solve problems. For example, many incidents were not classified in the right incident code, or not classified at all. This resulted in a low validity of the incident database: it was estimated that more than 30% of the incidents were coded incorrectly.

It was found necessary to first implement a clear and consistent registration of the incidents that occur during service delivery, before attempting to improve the problem management process.

3.5 Case E – developing a service catalog

This case study was done in the central IT department of a large Dutch governmental organization. The IT department develops, operates, and maintains hardware and software for the decentralized governmental organization. The goal of the case study was to investigate the possibility for using a service catalog to improve communication between the IT department and its customers. The purpose of the service catalog would be to facilitate the negotiation of service levels by providing a set of services combined with standard service levels that the IT department is able to provide, together with standard prices.

When the case study started the IT department had already developed a document that was supposed to be a service catalog. However, closer investigation showed that this document did not contain the information necessary to negotiate service levels: it hardly contained any quantitative data and no indications of costs of services. Further research showed that the organization did not only omit this information from the service catalog, but also that it did not have the necessary data available. This made it impossible to implement a full scale service catalog during the time-span of the case study.

3.6 Case F – developing a service catalog

This case study was done with an IT organization that delivers a wide spectrum of IT services, ranging from PC installation to system management. The organization uses ITIL to implement its service management processes. The organization has been split in a number of business units that work together to deliver integrated services. The organization has been using result-oriented service level agreements for a number of years and generally looks like an IT service provider that has become of age.

The goal of this case study was to implement part of a service catalog. The organization felt that a service catalog would be a good step towards their goal of quality improvement. The case study had the full commitment of both management and employees and resulted in a prototype service catalog that was used in the negotiations with a large customer.

3.7 Lessons learned

Although the six case studies discussed cover a wide range of issues and different organizations, we feel that several important lessons can be learned from these, and other case studies that we did. The most important lesson is that IT service improvement can only be successful if the organizational preconditions have been fulfilled. From our case studies we identified a number of these preconditions.

Several of our case studies were rather unsuccessful, mainly because the organization was not 'ready' for the new methodology introduced. The reason that an organization is not ready can be caused by cultural issues, but also by the lack of certain practices that are needed for the improvement. For example, the lack of historical data on services provided makes it impossible for the IT department from case E to develop a full fledged service catalog. Another example is the problem management process of the IT department in case D which cannot be properly executed due to the low quality of the incident database.

On the other hand, there are a number of case studies that were successful, despite the apparent low maturity of the organizations. For example, the ServQual evaluations of the service delivered by the case C organization were successful despite the lack of measurable service level agreements. Another example is the successful use of a service level agreement between customer and service provider in case A, despite the fact that this is the first time the IT organization provides this particular service.

We have seen that several practices need other practices to be performed:

- problem management needs consistent incident management;
- implementation of a service catalog needs historic information on service level agreements and performance.

And several other practices can be introduced in any IT service organization:

- service evaluation;
- service specification and reporting.

These lessons are reflected in the IT Service Capability Maturity Model presented in the next section.

4 The IT Service Capability Maturity Model

In this section we describe the proposed IT Service Capability Maturity Model. First, in section 4.1, we discuss some of the design decisions made in the development of the model. Next, the objectives of the IT Service CMM are laid out. The maturity levels of the IT Service CMM are described in section 4.3. Section 4.4 presents the key process areas of the model. Finally, section 4.5 presents the goals and common features of one of the key process areas as an example.

4.1 Design choices

During the development of the IT Service CMM a number of design choices have been made. In this section we discuss the two major decisions made and the motivation for these decisions. First, the focus on service capability is discussed in section 4.1.1. Second, the choice for an improvement and assessment based capability maturity model is discussed in section 4.1.2.

4.1.1 Scope: a Service Capability Model

A major difference between software and hardware development on the one hand, and software maintenance, system operation, network management, etc., on the other hand, is the fact that the first result in a *product,* whereas the latter result in a *service* being delivered to the customer. Usually, a service is defined as an essentially intangible set of benefits or activities that are sold by one party to another. The main differences between products and services are:

a) Services are transitory by nature, products are not. Hence, services can not be easily held in stock.

b) Product delivery results in a transfer of ownership, service delivery does not.

c) The use of products can be separated from the production of products. Services are produced and consumed simultaneously.

d) Services are largely intangible, whereas products are largely tangible.[2]

The difference between products and services is not clear-cut. Often, services are augmented with physical products to make them more tangible, for example, luggage tags provided with a travel insurance. In the same way, products are augmented with add-on services, for example a guarantee, to improve the quality perception of the buyer. Moreover, customers might even consider the quality of service more important than the characteristics of the product itself, e.g. [24].

Often, products and services are intertwined. An example is a newspaper subscription, in which case both the product – the newspaper itself – and the service – the daily delivery – are essential to the customer. This means that the quality of such a product-service mix will be judged on both product and service aspects: is the newspaper delivered on time, and does it contain the desired information.

Like the newspaper, IT management and maintenance can very well be a mixture of product and service. For example, in a situation where a software maintainer analyzes change requests for a fixed price per period and implements change requests for a price per change request, software maintenance is a product-service mixture. Here, the *service* is the customer having the possibility to have change requests analyzed, and the *product* is the implemented change.

Looking at IT management and maintenance activities from a service perspective, a number of issues that pertain to the quality of these activities emerge:

- If the activities are performed in an ongoing relationship with the customer, which they will almost always be, the service provider needs to facilitate communication between end-users and its organization. Moreover, this communication needs to be managed and controlled.

- The customer and the service provider have to agree on the quality levels with which the service will be delivered. Examples are: the maximum number of change requests that will be implemented per period, the availability of IT systems and networks, etc.

- The service provider and customer need to evaluate the service on a regular basis: is the service still what the customer needs?

- Possibly, the service provider has to cooperate with third parties to perform its job. For example, new software may be developed by a software house, and is subsequently maintained by the service provider. Or the software may be operated by a separate computer center and maintained by the service provider.

[2] Software obviously is not tangible, but it is still a product because of its other characteristics (a, b, and c).

Although each of the above points plays a role in software and hardware development too, the conjecture is that these activities are more important as service aspects are more prevalent. Regardless of the exact circumstances in which an IT service organization operates, sufficient emphasis should be on processes like the ones mentioned above, to be able to deliver quality IT services.

4.1.2 Form: a Capability Maturity Model

There are a two reasons why it was decided to use the capability maturity framework developed at the SEI as a basis for our service improvement model. First, the Software CMM is a widely used and well-known software process improvement model. We felt that its structure is generic enough to facilitate other areas besides software processes. This has already been shown by the development of other capability maturity models, such as the People CMM [6, 7] and the System Engineering CMM [2].

Second, we wanted to provide organizations with a mechanism with which they can perform stepwise improvement. Improvement should be an integral part of the framework. This is the case with the CMM where the framework functions as a prescriptive model and assessments are used to compare the actual situation with the model.

The granularity of the improvement steps of the CMM is rather coarse – an organization resides on one of five different levels. Other software process improvement models, such as BOOTSTRAP [19] or Trillium [26], use a more detailed architecture. BOOTSTRAP, for example, distinguishes between the maturity of the organization and the maturity of projects. Both Trillium and BOOTSTRAP rate the maturity of organizations with respect to different processes: this makes it possible that an organization rates level three for one process and level four for another process, for example. However, we decided to use the simpler approach of the CMM for practical reasons: we wanted to construct a fairly complete framework with limited resources, within limited time.

4.2 Primary objectives of the IT Service CMM

The objective of the IT Service CMM is twofold:

1. to enable IT service providers to assess their capabilities with respect to the delivery of IT services, and,

2. to provide IT service providers with directions and steps for further improvement of their service capability.

The IT Service CMM fulfills these goals by measuring the capability of the IT service processes of organizations on a five level ordinal scale. Each level prescribes certain key processes that have to be in place before an organization resides on that level. Key processes implement a set of related activities that, when performed collectively, achieve a set of goals considered important for enhancing service process capability. Hence, organizations can improve their service capability by implementing these key processes.

More formally, we define *IT service process capability* as the range of expected results that can be achieved by following a service process. *IT service process performance* represents the actual results achieved by following an IT service process. *IT service process maturity* is the extent to which a specific process is explicitly defined, managed, measured, controlled and effective. The IT Service CMM focuses on measuring and improving the IT service process maturity of IT service organizations.

An organization that scores high on the IT Service CMM scale will be able to:

- deliver quality IT services, tailored to the need of its customers;
- do so in a predictable, cost-effective way;
- combine and integrate different services, possibly delivered by different service providers, into a consistent service package;
- continually improve service quality in a customer-focused way.

4.3 The maturity levels of the IT Service CMM

We measure the service process maturity of organizations on a five level ordinal scale. The first – initial – level has no associated key process areas. This is the level where all IT service organizations reside that have not implemented the level two key process areas. Level two is the repeatable level. Organizations that have reached level two will be able to repeat earlier successes in similar circumstances. Thus the emphasis of level two is on getting the IT services right for one customer. On level three, the defined level, the service organization has defined its processes and is using tailored versions of these standard processes to deliver the services. By using common organization-wide standard processes, the process capability to deliver services consistently is improved. At level four, the managed level, organizations gain quantitative insight into their service processes and service quality. By using measurements and an organization-wide measurement database organizations are able to set and achieve quantitative quality goals. Finally, at level five, the optimizing level, the entire organization is focused on continuous process and service improvement. Using the quantitative measurements the organization prevents problems from recurring by changing the processes. The organization is able to introduce new technologies and services into the organization in an orderly manner.

More formally, we define the five maturity levels as follows:

1. *Initial level*: The IT service delivery process is characterized as ad hoc, and occasionally even chaotic. Few processes are defined, and success depends on individual effort and heroics.
2. *Repeatable level*: Basic service management processes are established to track cost, schedule and performance of the IT service delivery. The necessary discipline is in place to repeat earlier successes on projects with similar services and service levels.
3. *Defined level*: The IT service processes are documented, standardized, and integrated into standard service processes. All projects use approved, tailored versions of the organization's standard service processes for delivering IT services.
4. *Managed level*: Detailed measurements of the IT service delivery process and service quality are collected. Both the service processes and the delivered services are quantitatively understood and controlled.
5. *Optimizing level*: Continuous process improvement is enabled by quantitative feedback from the processes and from piloting innovative ideas and technologies.

4.4 The key process areas of the IT Service CMM

As stated in section 4.2, for an organization to reside on a certain maturity level, it needs to implement all key processes for that maturity level – and those for lower levels. The term *key* process area merely

Process categories **Levels**	**Management** Service planning, management, etc.	**Enabling** Support and standardization.	**Delivery** Actual service delivery.
Optimizing	Process Change Management	Technology Change Management	Problem Prevention
Managed	Quantitative Process Management		Service Quality Management
Defined	Integrated Service Management	Organization Process Focus Organization Process Definition Training Program	Service Delivery
Repeatable	Service Planning and Evaluation Service Tracking and Oversight Subcontract Management	Configuration Management Event Management Service Quality Assurance	
Initial	Ad hoc processes		

Table 2: Key process areas, assigned to process categories

means that these processes are seen as the key to reach a certain maturity level. There might be more – non-key – processes, but these are not strictly necessary to reach the next maturity level.

Table 2 gives an overview of the key process areas. The key process areas are grouped into three process categories: management, enabling and delivery. The first group is concerned with the management of services. The second category deals with enabling the delivery process by means of support processes and standardization of processes. The third category consists of the processes that result in the consistent, efficient delivery of services according to the appropriate quality levels. Below we present the key process areas for each of the maturity levels of the IT Service CMM.[3]

Level 1: Initial

There are no key process areas prescribed for level one.

Level 2: Repeatable

The key process areas for level two are concerned with establishing the processes that enable the organization to repeat earlier successful services in similar situations. We distinguish between two kinds of processes that an organization has to implement on this level. The first category deals with service management: the planning, specification, tracking and evaluation of services. The second category is concerned with service support: processes that support the activities that actually deliver the services.

[3]Note that because the model is still under development, the key process areas for level four and five have been specified in less detail than the level two and three key process areas.

The management processes on this level look as follows. First, the service provider and the customer draw up an agreement about the services to be delivered, the quality of the services – specified in terms of service levels – and the costs of the services. To ensure that the service levels are realistic, the service provider draws up a service plan that shows the feasibility of the service levels (Service Planning and Evaluation). During service delivery, the service provider tracks the realized service levels and reports these to the customer on a regular basis to demonstrate that the provider has indeed delivered the services against the promised service levels (Service Tracking and Oversight). After a period of service provision, the customer and the service provider review the service level agreement to see whether it still conforms to the IT needs of the customer (Service Planning and Evaluation). Just like the organization draws up a service level agreement with its customer, the organization should also use service level agreements when it delegates parts of the service delivery to third parties (Subcontract Management).

We identify three support processes that a level two organization needs to implement. First, almost all IT services concern the management, operation or maintenance of hardware and software components. Therefore, where necessary for consistent service delivery, these components are put under configuration control. This ensures that at all times the status and history of these components is known (Configuration Management). Second, during the period that the services are delivered, events can occur that need to be resolved by the service provider. These events range from simple requests for service to serious incidents that prevent the customer from using its information technology. All these events need to be identified, tracked, resolved and reported to the customer (Event Management). To service the request and to resolve incidents, changes to the configuration may be necessary. The change requests are evaluated by the configuration control board[4] with respect to the service level agreement and risk for the integrity of the configuration. Only after a change request has been approved by the change control board, will the configuration be changed (Configuration Management). Finally, to ensure the quality of the services, the service provider deploys quality assurance techniques, such as reviews and audits (Service Quality Assurance).

Next follows a description of the level two key process areas:

1. Service Planning and Evaluation:

   Purpose: Services are planned and realistic service levels are negotiated with the customer in order to deliver services that satisfy the customer's need for IT services. The delivered services, the specified service levels and the customer's service needs are reviewed with the customer on a regular basis. When necessary, the service level agreement is adjusted.

   There are three basic issues targeted by this key process area: first, the service to be delivered is specified in a contract – the service level agreement – containing *measurable* service levels. Second, the service levels specified should address the business needs of the customer. Third, the service provider should draw up a planning which shows his ability to deliver the agreed upon services. The service level agreement should at a minimum specify:

   (a) the services itself, i.e. a specification of the services to be delivered;

   (b) with what levels of service, i.e. how fast, how reliable, etc., specified in a measurable manner. Service levels need to be measurable because the organization has to report the realized service levels, see Service Tracking and Oversight.

   (c) the conditions the customer should obey. Examples of such conditions could be that the customer should run certain checks on data before running a query, or that the customer can process a maximum of 100,000 transactions per day.

[4] Note that this is a role, and not an actual organizational unit.

(d) what happens if the service provider does not reach the agreed upon service levels while the customer did not violate the customer conditions.

(e) when and what will be reported to the customer regarding the actual delivered services.

(f) when and how the service level agreement will be reviewed.

(g) under which circumstances (calamities) service is not guaranteed.

2. Service Tracking and Oversight:
   Purpose: Service delivery is being tracked. The realized service levels are compared with the specified service levels and are reported to the customer and management on a regular basis. Corrective actions are taken when actual service delivery deviates from the specified service levels.

   The service provider reports to the customer the actual services delivered (1a), the actual service levels (1b) and, when relevant, calamities that hindered accurate service delivery (1g). The service level reports are used as input for the evaluation of service level agreements (see Service Planning and Evaluation).

3. Subcontract Management:
   Purpose: Select qualified IT subcontractors and manage them effectively.

   The service provider can select and hire subcontractors to delegate parts of the service. If this is the case, the service to be delivered by the subcontractors is laid down in a service level agreement. The service provider keeps track of the actual services delivered by the subcontractor and takes corrective actions when the actual service levels deviate from the specified service levels.

4. Configuration Management:
   Purpose: The integrity of products which are subject to or part of the IT services is established and maintained.

   Configuration Management involves the identification of the relevant hardware and software components which need to be put under configuration control. This includes components owned by the customer that are being managed by the service provider, components owned by the provider that are used by the customer and components owned by the provider that are used to deliver the service. Changes to the configuration are evaluated with respect to the service level agreement and with respect to possible risks for the integrity of the configuration.

5. Event Management:
   Purpose: Events regarding the service are identified, registered, tracked, analyzed, and resolved. The status of events is communicated with the customer and reported to management.

   This key process area concerns the management of events that causes or might cause service delivery to deviate from the agreed upon service levels. Events can be either:

   - Requests for service from users. For example, requests for a new feature in the software;
   - Incidents that cause or will cause service levels to be lower than agreed upon if no action is being taken. For example, a server that is down might cause the specified maximum down-time to be exceeded if it is not restarted quick enough.

   To resolve requests for service and incidents, changes to the configuration might be necessary. The decision whether to implement the change request that results from a service request or incident is the concern of Configuration Management.

6. Service Quality Assurance:
   Purpose: Management is provided with the appropriate visibility into the processes being used and the services being delivered.

   Service Quality Assurance involves the reviewing and auditing of working procedures, service delivery activities and work products to see that they comply with applicable standards and procedures. Management and relevant groups are provided with the results of the reviews and audits. Note that where Service Tracking and Oversight is concerned with measuring service quality in retrospect, from an external point of view, Service Quality Assurance is concerned with measuring quality in advance, from an internal point of view.

Level 3: Defined

At level three, an organization standardizes its processes and uses tailored versions of these standard processes to deliver the IT services. This results in more predictable performance of the processes and hence it increases the ability of the organization to draw up realistic service level agreements. The level three key process areas each fall into one of the three process categories: management, enabling or delivery.

The first category – service management – is concerned with the tailoring of the standard service processes to the customer and the service level agreement at hand. Also, the actual service processes need to be integrated with each other and with third party service processes (Integrated Service Management). The second category – enabling – deals with making standard processes available and usable. The organization develops and maintains standard processes for each of the services it delivers. Usually, organizations will provide several services to one customer at the same time. Hence, not only the service processes themselves, but also the integration of these processes has to be standardized as much as is feasible (Organization Process Definition). To coordinate process efforts across services and organizational units and over time, organizational support is institutionalized (Organization Process Focus). Also, to teach people how to work with the standards and how to perform their roles, a training program needs to be put in place (Training Program). The third category – service delivery – concerns the actual delivery of the services to the customer using the tailored service processes (Service Delivery).

The level three key process areas are described as follows:

1. Organization Process Definition:
   Purpose: Develop and maintain a usable set of service process assets that improve process performance across services, and provide a basis for cumulative, long-term benefits to the organization.

   This key process area involves the creation and maintenance of standard service processes, and a process database which contains historic data on used processes, including the service level agreements, the service planning, the service level reports and the event management database. Based on historic service processes a service catalog is developed and maintained which contains the services and service levels that the organization provides.

2. Organization Process Focus:
   Purpose: Establish organizational responsibility for service process activities that improve the organization's overall service process capability.

   The activities needed to assess, develop, maintain and improve the organization's and projects' service processes are resourced and coordinated across current and future services.

3. Training Program:
   Purpose: Develop the skills and knowledge of individuals so they can perform their roles effectively and efficiently.

4. Integrated Service Management:
   Purpose: Integrate the IT service and management activities into a coherent, defined IT service process that is derived from the organization's standard service process.

   The service planning is based on this tailored service process and describes how its activities will be implemented and managed. The service planning takes the organization-wide capacity and availability of resources into account. Cooperation with third parties that also deliver IT services or products to the customer, is planned. Note that these third parties can be external providers or organizational units of the customer itself. An example of this could be the customer having its own helpdesk which relays reports of hardware failures to the service provider. Procedures need to be put in place on how these reports will be delivered to the service provider and whether the helpdesk or the service provider will inform the user of the status of the report. An example which involves coordination with third parties that deliver products to the customer, is software development. Suppose a third party is developing software for the customer that is to be managed and maintained by the service provider. Involvement of the service provider in the development process can ensure that maintenance and management of the software is sufficiently being taken into account during development.

5. Service Delivery:
   Purpose: Consistently perform a well-defined service delivery process that integrates all service delivery activities to deliver correct, consistent IT services effectively and efficiently.

   Service Delivery involves the performing of service delivery activities using a tailored version of the services' defined service processes (which is the output of the Integrated Service Management key process area). Because the service activities depend on the particular services being provided, there is no fixed list of activities to be performed. However, all service projects should perform the activities as defined in the level two key process areas. The list of activities will be filled in depending on the services at hand. For example, in the case of software maintenance the general service activities will be extended with the software engineering tasks mentioned in the key process area Software Product Engineering of the Software CMM [5, pp. 241–261].

Level 4: Managed

At level four, organizations gain a quantitative understanding of their standard processes by taking detailed measures of service performance and service quality (Quantitative Process Management) and by using these quantitative data to control the quality of the delivered services (Service Quality Management).

There are two level four key process areas:

1. Quantitative Process Management:
   Purpose: Control the process performance of the service project quantitatively.

2. Service Quality Management:
   Purpose: Develop a quantitative understanding of the quality of the services delivered and achieve specific quality goals.

Level 5: Optimizing

At level five, service providers learn to change their processes to increase service quality and service process performance (Process Change Management). Changes in the processes are triggered by improvement goals, new technologies or problems that need to be resolved. New technologies are evaluated and introduced into the organization when feasible (Technology Change Management). Problems that occur are prevented from recurring by changing the processes (Problem Prevention).

The level five key process areas are:

1. Process Change Management:
   Purpose: Continually improve the service processes used in the organization with the intent of improving service quality and increasing productivity.

2. Technology Change Management:
   Purpose: Identify new technologies and inject them into the organization in an orderly manner.

3. Problem Prevention:
   Purpose: Identify the cause of problems and prevent them from recurring by making the necessary changes to the processes.

4.5 Common features of the key process areas

Each key process area is defined in terms of its goals and its common features. Common features are the activities that an organization needs to perform to properly implement a key process area. The common features are divided into five categories: commitment to perform, ability to perform, activities performed, measurement and analysis and verifying implementation.

This section presents the common features of the key process area Service Planning and Evaluation as an example.

Goals

Goal 1 Service commitments and service delivery activities are planned and documented.

Goal 2 Service commitments and services delivered are evaluated.

Commitment to Perform

Commitment 1 A service manager is designated to be responsible for negotiating service commitments and developing the service delivery plan.

Commitment 2 The service is planned and evaluated according to a written organizational policy.

Ability to perform

Ability 1 Responsibilities for developing the service plans and commitments are assigned.

Ability 2 Adequate resources and funding are provided for planning the service delivery.

Ability 3 The service managers, service engineers and other individuals involved in the service planning are trained in the estimating and planning procedures applicable to their areas of responsibility.

Activities performed

Activity 1 The IT service needs of the customer are documented.

Activity 2 The service commitments are documented.

Activity 3 Commitments made to individuals and groups external to the organization are reviewed with senior management according to a documented procedure.

Activity 4 The service plan is developed according to a documented procedure.

Activity 5 The service plan is documented.

Activity 6 Software and hardware products that are needed to establish and maintain control of the service delivery are identified.

Activity 7 Estimates for the service delivery workload are derived according to a documented procedure.

Activity 8 Estimates for the service delivery effort and costs are derived according to a documented procedure.

Activity 9 The service delivery schedule is derived according to a documented procedure.

Activity 10 The risks associated with the cost, resource, schedule and technical aspects of the service are identified, assessed, and documented.

Activity 11 Plans for the service facilities and support tools are prepared.

Activity 12 Service planning data are recorded.

Activity 13 Service commitments, service planning and service delivery are evaluated with the customer on both a regular and an event-driven basis.

Measurement and Analysis

Measurement 1 Measurements are made and used to determine the status of the service planning activities.

Verifying Implementation

Verification 1 The activities for service planning are reviewed with senior management on a periodic basis.

Verification 2 The activities for service planning are reviewed with the service manager on both a periodic and event-driven basis.

Verification 3 The service quality assurance group reviews and/or audits the activities and work products for service planning and evaluation and reports the results.

5 Conclusions

Having described the IT Service CMM, we now look at the case studies again from a service maturity perspective. Because each case study addressed only part of the issues that are important for service quality, we cannot do a complete assessment of the organizations. We can, however, indicate what happened during the case studies, and how that connects with certain aspects of the maturity of the organization.

Case A – developing a service level agreement Here the SLA specification method was successfully used to develop a result-oriented, measurable service level agreement. The SLA specification method was just a formalization of a practice that the service provider already uses in its service delivery process.

Case B – developing a generic service level agreement Here the goal was to introduce the SLA specification method and develop a generic service level agreement for the organization. However, the organization had little experience with result-oriented service delivery and this made it virtually impossible to develop a generic SLA. From this case we conclude that generic, organization-wide SLAs can only be introduced after sufficient experience with result-oriented service agreements and service delivery.

Case C – evaluating service quality The goal of this case was to evaluate the quality of the IT services delivered. Although the organization can be considered rather immature, the case study was still quite successful. We conclude that an evaluation of the IT services delivered can be done regardless of the maturity level of the service provider.

Case D – incident and problem management Here we conclude that problem management needs sound incident management. This is reflected in the IT Service CMM by the fact that the key process area Event Management, which includes incident management, is positioned on a lower level than Problem Prevention, which includes problem management.

Case E and F – developing a service catalog A service catalog needs experience with services and service level agreements. This is reflected in the model by the fact that the key process area Organization Process Definition, which includes the development of a service catalog, is positioned on level three, i.e. after the basic service management processes of level two have been implemented.

The experiences we have gained with these and other case studies have been assimilated in our IT Service CMM.

The IT Service Capability Maturity Model as presented in section 4 is work in progress. We are still working with our research partners on the precise definition of the key process areas, and on an accompanying assessment method. The assessment method is currently being developed and tested with our partners. Together, the IT Service CMM and the assessment method should result in an improvement methodology for IT service processes.

Although the IT Service CMM is still in its infancy, we feel that it can provide a valuable contribution to quality improvement of IT services. Our business partners indicate that this IT Service CMM fills an important gap between high level quality management frameworks, such as Total Quality Management (TQM) and best practices such as promoted by ITIL.

Acknowledgments

This research was partly supported by the Dutch Ministry of Economic Affairs, projects 'Concrete Kit', nr. ITU94045, and 'KWINTES', nr. ITU96024. Partners in these projects are Cap Gemini, Twijnstra Gudde, the Tax and Customs Computer and Software Centre of the Dutch Tax and Customs Administration, and the Technical Universities of Delft and Eindhoven.

References

[1] James Bach. The Immaturity of the CMM. *American Programmer*, September 1994.

[2] Roger Bate, Dorothy Kuhn, Curt Wells, James Armitage, Gloria Clark, Kerinia Cusick, Suzanne Garcia, Mark Hanna, Robert Jones, Peter Malpass, Ilene Minnich, Hal Pierson, Tim Powell, and Al Reichner. A Systems Engineering Capability Maturity Model, Version 1.1. Technical Report CMU/SEI-95-MM-003, Software Engineering Institute/Carnegie Mellon University, November 1995.

[3] Terry B. Bollinger and Clement McGowan. A Critical Look at Software Capability Evaluations. In IEEE Software [17], pages 25–41.

[4] Special Issue on 'The Quality Approach: Is it Delivering?'. *Communications of the ACM*, 40(6), June 1997.

[5] Carnegie Mellon University/Software Engineering Institute. *The Capability Maturity Model: Guidelines for Improving the Software Process.* SEI Series in Software Engineering. Addison-Wesley Publishing Company, 1995.

[6] Bill Curtis, William E. Hefley, and Sally Miller. Overview of the People Capability Maturity Model. Technical Report CMU/SEI-95-MM-01, Software Engineering Institute/Carnegie Mellon University, September 1995.

[7] Bill Curtis, William E. Hefley, and Sally Miller. People Capability Maturity Model. Technical Report CMU/SEI-95-MM-02, Software Engineering Institute/Carnegie Mellon University, September 1995.

[8] Michael K. Daskalantonakis. Achieving Higher SEI Levels. *IEEE Software*, 11(4):17–24, July 1994.

[9] Michael Diaz and Joseph Sligo. How Software Process Improvement Helped Motorola. *IEEE Software*, 14(5):75–81, September/October 1997.

[10] Raymond Dion. Process Improvement and the Corporate Balance Sheet. *IEEE Software*, 10(4):28–35, July 1993.

[11] Daniel W. Drew. Tailoring the Software Engineering Institute's (SEI) Capability Maturity Model (CMM) to a Software Sustaining Engineering Organization. In *Proceedings of the Conference on Software Maintenance*, Orlando, Florida, November 9-12, 1992. IEEE Computer Society.

[12] Suzanne M. Garcia. Evolving Improvement Paradigms: Capability Maturity Models and ISO/IEC 15504 (PDTR). *Software Process – Improvement and Practice*, 3(1):47–58, March 1997.

[13] E.M. Gray and W.L. Smith. On the limitations of software process assessment and the recognition of a required re-orientation for global process improvement. *Software Quality Journal*, 7(1):21–34, January 1998.

[14] James Herbsleb, David Zubrow, Dennis Goldenson, Will Hayer, and Mark Paulk. Software Quality and the Capability Maturity Model. In CACM [4], pages 30–40.

[15] Craig Hollenbach, Ralph Young, Al Pflugrad, and Doug Smith. Combining Quality and Software Improvement. In CACM [4], pages 41–45.

[16] Watts S. Humphrey, Terry R. Snyder, and Ronald R. Willis. Software Process Improvement at Hughes Aircraft. In IEEE Software [17], pages 11–23.

[17] *IEEE Software*, 8(4), July 1991.

[18] *The IT Infrastructure Library - An Introduction.* Information Technology Infrastructure Library. CCTA, 1992.

[19] Pasi Kuvaja, Jouni Similä, Lech Krzanik, Adriana Bicego, Günter Koch, and Samuli Saukkonen. *Software Process Assessment and Improvement: The BOOTSTRAP Approach*. Blackwell Publishers, 1994.

[20] Bob McFeeley. IDEAL^{SM}: A User's Guide for Software Process Improvement. Handbook CMU/SEI-96-HB-001, Software Engineering Institute/Carnegie Mellon University, February 1996.

[21] Leyland F. Pitt, Richard T. Watson, and C. Bruce Kavan. Service Quality: A Measure of Information Systems Effectiveness. *Management Information Systems Quarterly*, 19(2):173–187, June 1995.

[22] Sarah A. Sheard. The Frameworks Quagmire, A Brief Look. In *Proceedings of the 1997 INCOSE Conference*, Los Angeles, 3-7 August 1997.

[23] Standard for Software Process Improvement and Capability Determination. Version 1.0, ISO/IEC, 1995.

[24] T. Stålhane, P.C. Borgersen, and K. Arnesen. In Search of the Customer's Quality View. *The Journal of Systems and Software*, 38(1):85–93, July 1997.

[25] J.M. Trienekens and M. van der Zwan. Zonder harde afspraken geen verbetering dienstverlening in IT. *Automatisering Gids*, (39):19–21, 27 September 1996.

[26] Trillium Model – For Telecom Product Development and Support Process Capability. Model Issue 3.2, Bell Canada, May 1996.

[27] Richard T. Watson, Leyland F. Pitt, and C. Bruce Kavan. Measuring Information Systems Service Quality: Lessons from Two Longitudinal Case Studies. *Management Information Systems Quarterly*, 22(1):61–79, March 1998.