

A Service Model for Corporate Management

For decades, the term *Service* has been used in the IT industry. The meaning, however, has changed considerably over the years. First, ITIL changed the view on Services, and now also current topics such as virtualization, cloud and agile development methods influence the meaning of the term. For technologies that are increasingly connected the division into Services is an opportunity to make this complexity manageable and give the individual areas a high degree of self-responsibility. By taking non-IT Services into consideration, you can further develop this model into a complete management model, which is very well suited for supporting skills such as self-responsibility and self-organization.

Comparing Services to Processes

Processes

According to Wikipedia, a business process is a series of logically connected individual activities (tasks, activities) that are carried out with the purpose to reach a business or organizational goal.

Services

According to ITIL, a Service is an opportunity to bring added value to the customer. In this case added value means that the customer is enabled and supported in reaching certain goals.

In other words you might say that the process description is all about *HOW* things are done and the service describes *WHAT* is done.

The description of individual activities is always useful, when basic conditions are the same, and we are aiming at a consistent quality. The pharmaceutical industry is a good example. Here, under laboratory conditions, results must be generated that are absolutely reproducible. Generally, this applies to most industrial production processes.

Complex services, like for instance incident handling in an IT environment are difficult to describe. They can only be defined in very abstract terms. According to ITIL, the Incident process normally looks like this:

Receipt of Incident → Categorizing of Incident → Incident troubleshooting → Closing of Incident

There is a reason why in this specific environment the description of a *Service* as a service was established. An Event-driven Process Chain (EPC) is either too superficial or too complex and inflexible to be helpful with (a more or less detailed) description. Various basic conditions and a lot of decisions quickly lead to a very complex and therefore useless diagram.

Instead it is often easier to describe the service as a kind of blackbox that contains information about what are the entry values and what the result should look like. This is exactly the content we find in Service Level Agreements (SLAs). They describe exactly what the provided service should look like, when it should be available and what customers need to contribute. How this service is provided in the end is the sole responsibility of the service providers.

This approach has many advantages: Services described this way can easily be handled by an external third party provider via an outsourcing contract. In addition, it is easier to compare the

service to competitors in the market. Furthermore, the responsible team can work self-responsibly and self-organized. This encourages creativity, and improvements can be implemented quicker, as no described activities must be documented from scratch. When the performance and the price are fixed, the motivation for improvement is high, as the relevant team/service provider directly benefits from the improvement.

Why Services?

Below, two current developments in IT organizations are introduced as examples of changes that can be described with an IT Service model.

(1) Agile Methods

In software development more and more agile methods are implemented. This way development is able to deliver new releases in a very short time frame.. However, these releases often get stuck in traditionally organized deployment processes. How can *test centers* provide their users with new releases to be tested within a certain timeframe when development is delivering new so called daily builds every day?

Agile Development requires automated deployment processes

Agile development also requires a corresponding automated deployment process. That doesn't mean testing is not required or that the integration of an application into the whole architecture doesn't need to be considered carefully. In a virtual environment, however, these considerations only lead to parameters of an automated deployment process. Also, test cases for new functionality must be entered into the test software. When this is done, the deployment process can automatically run and the new release with its functionality will soon be available for the user.

Only now, the new release cannot simply be thrown over the fence and into the next department. The intellectual effort that was distributed across different departments before, now needs to be brought together.

The actual challenge of Devops is change

The industry is constantly talking about DevOps. With DevOps, the major challenge is not the process or processes as such. It is the change process. Activities and tasks handled by different departments with opposite goals, are now supposed to be handled together in one team. This change can be implemented a lot easier when participants self-responsibly manage the service portfolio. In this context a detailed process definition would always take too long, be too complex and full of conflicting interests.

(2) Cloud

Cloud Services have some very obvious advantages: Organizations offering services with time specific demand and delivery are able to deliver their services from the cloud efficiently. Traditional data centers often have extra capacities that they only need for a few hours every month. A dynamic and automated allocation of resources for service delivery in a Cloud infrastructure usually allows using these resources more efficiently.

Traditional processes are too complex and expensive

However, a *support process* for infrastructures like this is usually more complex. A traditional process is either too slow, when the components involved are used sequentially, or too expensive if the components involved are checked in parallel.

When pilot projects are successfully completed, usually a so called *competence team* is implemented to secure smooth operation. This kind of interdisciplinary team decides what must be done in order to secure the availability of the services or how to restore them

Competence Teams replace traditional operational and structural organizations

When this operation model is no longer an exception, but the rule, the traditional operational and structural organization is no longer needed, because the *competence team* is responsible for all the relevant tasks. This newly created organization can be described using services a lot easier and even more accurately.

Services as Corporate Management Models

Many IT organizations describe their whole value chain as services. They distinguish between Business Services and supporting Services. Business Services are services sold to an external third party whereas supporting services are services that are internally available and a prerequisite to provide Business Services. In order to supply a customer with an SAP installation for example, you need a server with an operating system, storage, databases and a network.

This model can also be applied to non-IT services and extended towards a complete management system. Administrative tasks such as accounting or infrastructure tasks such as facility management are then also defined as services. .

This model clearly shows the value each and every task or activity is adding to the value chain. Internal service providers (accounting, facility management, HR) immediately recognize who is using their service (internally and externally) and for what purpose it is used. At the same time the model is supporting the relevant service providers to improve the services they offer or to develop new services.

A clear overview over who delivers what and who gets what from others is one of the biggest advantages of this kind of management model. A major risk lies in a rather formalistic treatment of each other in the context of the internal customer/supplier relationship. In case of conflict focus on the common goal might even get lost. An appropriate business culture though is able to reduce or eliminate this risk. This should also be taken into account with relationships to external suppliers.

How Services Are Defined

Even in very similar organizations Service Catalogs might look very different. Ten years ago Service Catalogs mostly consisted of categories that were strongly oriented towards the organization's overall structure. Later they were more focused on customers. Services were defined that end-users could actually relate to. Catalogs also differed from service catalogs with only a few details to catalogs with hundreds of detailed entries.

There is no right or wrong when it comes to the complexity of service catalogs. The complexity basically only depends on the answers and the information the catalog is supposed to provide.

Models like Service Catalogs are supposed to simplify a complex reality. They are supposed to explain certain phenomena or to make it easier to recognize coincidences. Models like this have their limits and sometimes different models are required to represent different scenarios. When people try to develop an all-in-one solution suitable for every purpose, the actual benefit, the simplification of complex relationships, is getting lost.

The first step in creating a Service Catalog is to determine and define the scope of questions the catalog is supposed to answer. If the goal is to define service agreements with customers, cost factors need to be included. Also availability needs to be measured and defined. After that you can start building up the actual catalog. For each level of differentiation you need to define the questions requiring differentiation. And who is responsible for individual elements. Of course, each additional entry in a service catalog is related to costs. Therefore the cost-benefit-ratio needs to be examined carefully before (additional) entries are made to the catalog.

About the author



Jürgen Groß, Team- and Project Coach

I have been working in the IT industry for almost 30 years helping customers improve their IT services. For the past 15 years I have also been working with enterprise culture for the improvement of common trade – dealing with team building, meeting culture, transparency, organizational structure, communication, cooperation, management systems, strategies, visions, guiding principles, management methods, in short everything that affects the way people relate to each other in companies.

In addition to the standard ITSM and CRM tool implementations, I also support project managers with the changes and projects described above.

Furthermore, I am doing consulting for companies who want to change in the direction described above and who want to become more independent and self-organized. From my point of view this kind of change provides the greatest potential for increasing efficiency in businesses today. Changes of this kind have some very basic prerequisites. If they are not met, the effort to change often fails and time is wasted. With my professional background and experience I make sure that the whole process is started the right way, and that the necessary steps are taken in the right order.

Contact Information:

Jürgen Groß
Team- und Projektcoaching
Kleine Wies 2
85354 Freising
Germany

T: +49 (8161) 231 711
F: +49 (8161) 231 712
Mail: jgross@gross-team.com

