

Methode für das Design von SLA-fähigen IT-Services

Christian Mayer¹, Stefan Link², Matthias Racke², Stefan Popescu², Tobias Vogel²,
Oliver Mehl², Sebastian Abeck²

¹ Universität Karlsruhe (TH), Fakultät für Informatik, Dekanat,
Arbeitsgruppe Lehrunterstützung, Postfach 6980, 76128 Karlsruhe
mayerl@ira.uka.de

² Universität Karlsruhe (TH), Fakultät für Informatik, Institut für Telematik,
Cooperation & Management, Postfach 6980, 76128 Karlsruhe
[link | racke | popescu | vogel | mehl | abeck]@cm-tm.uka.de

Zusammenfassung. Da es sich bei IT-Services um immaterielle Produkte von IT-Betreibern handelt, basieren SLA-Verhandlungen zwischen Dienstleistern und Dienstnehmern auf IT-Service-Beschreibungen. Derartige Beschreibungen müssen als Kriterien erfüllen, dass die Funktionalitäten und Qualitäten der angebotenen IT-Services mit den verfügbaren Ressourcen des IT-Dienstleisters realisiert und entsprechend den Beschreibungen ausgehandelte SLAs nach Abschluss überwacht und erfüllt werden können. Wengleich bestehende Ansätze Strukturen für die Beschreibung von IT-Services und SLAs aufzeigen, so fehlt bis jetzt ein systematisches Vorgehen, nach dem auf Basis technischer und ablauforganisatorischer Ressourcen IT-Services und deren Beschreibungen entwickelt werden können. Der vorliegende Beitrag zeigt eine Service-Design-Methode als Prozess kooperierender Rollen eines IT-Dienstleisters zur Gewinnung von IT-Service-Beschreibungen auf. Ziel der Methode ist die Definition SLA-fähiger IT-Services, anhand deren Beschreibungen Verhandlungen zwischen IT-Dienstleistern und Dienstnehmern geführt werden können.

1 Einleitung

Heutige Betreiber von Informationstechnologien (IT) positionieren sich als Dienstleister gegenüber Anwendern von IT, die als Dienstnehmer im Kontext ihrer Geschäftsprozesse Information verarbeitende Funktionalität mit definierter Qualität nachfragen. Gegenstand der Kooperation zwischen IT-Dienstleister und Dienstnehmer sind IT-Services, bzgl. derer Leistungs- und Qualitätseigenschaften verhandelt und *Service Level Agreement* (SLA) vereinbart werden. Das Spektrum der IT-Services reicht von IT-Kerndiensten, die Funktionen zur Verarbeitung von Informationen bereitstellen und auf Funktionen von Netz-, System- und Anwendungssoftware-technologien zurückzuführen sind, bis hin zu Zusatzdiensten, die Informations-, Beratungs-, Schulungs-, Problembhebungs-, Änderungsdurchführungsleistungen usw. umfassen und durch definierte Betriebsprozesse innerhalb eines IT-Dienstleisters erbracht werden [7]. Im Zusammenhang mit den gewünschten Services werden Qualitätsanforderungen einerseits an die zu betreibende technische Service-Infrastruktur und andererseits an die durchzuführenden Betriebsprozesse des IT-Dienstleisters gestellt.

Nach Abschluss von SLAs steht ein IT-Dienstleister in der Pflicht, die definierten Services mit der vereinbarten Qualität den jeweiligen Dienstnehmern zur Verfügung

zu stellen [14]. Sowohl Dienstnehmer als auch Dienstleister sind an einem Nachweis der SLA-Erfüllung und damit an einer Qualitätsüberwachung interessiert. Folglich setzt die garantierte Einhaltung von SLAs eine Überwachung der IT-Services voraus, deren Leistungs- und Qualitätseigenschaften auf den zur Verfügung stehenden Ressourcen des IT-Dienstleisters beruhen, deren technische und ablauforganisatorische Eigenschaften gemessen werden können. Umgekehrt werden diese wiederum auf die Eigenschaften der bereitzustellenden Services abgebildet. Mit dem Ziel, abgeschlossene SLAs überwachen und erfüllen zu können, ist ein IT-Dienstleister deshalb bereits beim Design der Services an einem strukturierten Vorgehen interessiert, mit dessen Hilfe die verfügbaren Leistungsmöglichkeiten als Angebot gegenüber den Dienstnehmern beschrieben und zur Verfügung gestellt werden können.

Bestehende Ansätze im Bereich *IT Service Management* [2] zeigen Lösungen für die Beschreibung und Verwaltung von IT-Services auf, anhand derer SLAs verhandelt und abgeschlossen werden können. Probleme bei der Überwachung und Einhaltung von abgeschlossenen SLAs lassen sich jedoch oft auf ein nicht durchgängig strukturiertes Vorgehen bei der Konstruktion der angebotenen IT-Services zurückführen, das eigentlich von den technischen sowie ablauforganisatorischen Möglichkeiten eines IT-Dienstleisters ausgehen sollte. Es fehlt ein methodisches Vorgehen, wie ein IT-Dienstleister bereits beim Design auf Basis seiner Ressourcen seine verfügbaren Leistungen erfassen und entsprechende IT-Services entwickeln kann.

In diesem Beitrag wird eine Methode vorgestellt, mit deren Hilfe ein IT-Dienstleister auf der Basis seiner vorhandenen technischen und ablauforganisatorischen Möglichkeiten Leistungen als IT-Services beschreiben kann. Die so erstellten Service-Beschreibungen abstrahieren von den zugrunde liegenden Ressourcen und stellen so den fundierten Ausgangspunkt für den Abschluss von SLAs dar, deren Einhaltung und Erfüllung überwacht und nachgewiesen werden kann. Im nächsten Abschnitt werden hierzu zunächst bestehende Ansätze diskutiert und bewertet. Im darauf folgenden Abschnitt wird die Service-Design-Methode als kooperativer Prozess verteilter Betreiberrollen definiert. Es schließt sich ein Abschnitt über Service-Management-Werkzeuge zur Unterstützung der Methode an. Der Beitrag endet mit einer Zusammenfassung der wichtigsten Erkenntnisse und einem Ausblick auf aktuell anstehende Weiterentwicklungen.

2 Bestehende Ansätze

Im IT-Lebenszyklus endet die Phase der Entwicklung mit der Auslieferung des fertig gestellten IT-Produkts [13]. Unter der Prämisse, dass die Funktionalität einer IT als Service produktiv genutzt werden soll, schließt sich an die Entwicklung der IT die Einführung und Integration des IT-Produkts in das Betriebsszenario eines IT-Dienstleisters an. Neben der technischen Installation, Konfiguration und Integration müssen die Leistungs- und Qualitätseigenschaften der neuen Technologie erfasst und die den Dienstnehmern anzubietenden, immateriellen IT-Services entworfen werden.

Im noch jungen Forschungsbereich *Service Engineering* wird das Service-Design als Entwurfsphase für immaterielle Produkte und Geschäftsbeziehungen zwischen Organisationen in Analogie zum Entwurf materieller Produkte gesehen [1, 8]. Dabei beschäftigt sich die Service-Design-Phase mit der Gestaltung wahrnehmbarer Eigen-

schaften von Dienstleistungen [11]. Wenngleich bestehende Ansätze im Bereich *Service Engineering* ein ingenieurmäßiges Vorgehen zu Analyse, Design, Bereitstellung und Ablösung von Services als notwendig ansehen, so fehlt eine Konkretisierung der Ansätze insbesondere im Bereich von IT-Services.

Die wohl umfangreichste Sammlung von Vorgehensbeschreibungen im Bereich *IT Service Management* (ITSM) stellt die *IT Infrastructure Library* (ITIL) dar [13]. ITIL gilt als De-facto-Standardwerk für die Definition von IT-Betriebsprozessen, an dem sich neben IT-Dienstleistern auch viele Entwickler von Management-Werkzeugen zur Unterstützung der Prozesse orientieren. Beiträge für die Bereitstellung von IT-Services liefert ITIL mit den Empfehlungen zum *Service Support* und *Service Delivery*, und hier insbesondere mit der Definition des Prozesses für das *Service Level Management* (SLM). Dieser beginnt mit der Identifikation von Kundenwünschen und dem Aushandeln von Services und SLAs und endet mit der SLA-Überwachung und der Erstellung eines Qualitätsnachweises. Der SLM-Prozess legt den Hauptfokus auf die Interaktion zwischen einem IT-Dienstleister und einem Dienstnehmer. Obwohl erwähnt wird, dass die Anforderungen seitens des Dienstnehmers mit den Leistungsmöglichkeiten des IT-Dienstleisters abgeglichen werden müssen, schweigt sich die ITIL über ein konkretes Vorgehen im Bereich IT-Service-Design aus.

[15] beschäftigt sich mit Fragen des Betriebs und des Managements speziell von Telekommunikationsdiensten und -infrastrukturen. Ziel ist die Automatisierung der Geschäftsprozesse im Telekommunikationsbereich. Ein wesentliches Ergebnis dieser Arbeit ist die *enhanced Telecom Operations Map* (eTOM), in welcher Betriebsprozesse von Telekommunikationsunternehmen beschrieben werden [5, 6]. Neben den Prozessdefinitionen stellt die *Next Generation Operations Systems and Software* (NGOSS) ein von der Industrie anerkanntes Rahmenwerk zur Entwicklung betreibergerechter Systeme dar. NGOSS folgt den aus der Lehre der Softwaretechnik bekannten Schritten zur Softwareentwicklung (Geschäftsbereichsanalyse, Systemanalyse, Systementwurf und Systemtest) und liefert Ansätze für die Entwicklung von Management-Werkzeugen, welche die entstehenden IT-Lösungen und die darauf aufbauenden IT-Services qualitätsgesichert betreibbar machen. Jedoch fehlt bei eTOM eine systematische Herleitung und Dokumentation der Prozesse, wie z.B. für das Service-Design.

Aktuelle Forschungsarbeiten tragen zu ausgewählten Teilfragestellungen in den Bereichen *Service Engineering* und *IT Service Management* bei. [4] stellt beispielsweise ein strukturiertes Dienstmodell vor, welches die vereinheitlichte Basis für weitere Konzepte des *IT Service Managements* bildet. Das Modell unterscheidet zwischen einer kundenspezifischen, einer betreiberspezifischen und einer seitenunabhängigen Sicht auf einen Service. Der kundenspezifische Anteil beschreibt die für den Dienstnehmer über einen definierten *Service Access Point* abrufbare Funktionalität. Der betreiberspezifische Anteil ergänzt Service-Informationen, die für den Betrieb und weniger für den Dienstnehmer relevant sind. Eine ausführliche Beschreibung des Dienstmodells findet sich in [5]. [9] zeigt den Zusammenhang zwischen der Modellierung von IT-Services und dem Abschluss von SLAs auf. Jedoch fehlt auch in den aktuellen Forschungsergebnissen ein systematischer Ansatz, der beschreiben würde, wie ausgehend von den Anforderungen an IT-Services diese auf Basis der Leistungsmöglichkeiten eines IT-Dienstleisters entworfen werden können.

3 Service-Design-Methode

Ziel der Service-Design-Methode ist die systematische Konstruktion von Service-Beschreibungen auf Basis vorhandener Ressourcen eines IT-Dienstleisters. Da ein IT-Service eine immaterielle, nicht lagerbare Leistung darstellt, liefern Service-Beschreibungen die Grundlage für SLA-Verhandlungen zwischen IT-Dienstleistern und IT-Dienstnehmern. Neben einer dienstnehmergerechten Abstraktion von den IT-Services zugrunde liegenden Ressourcen muss ein IT-Service Kriterien wie technische und ablauforganisatorische Realisierbarkeit, Abrechenbarkeit sowie Überwachbarkeit erfüllen. Beispielhaft soll im Folgenden der IT-Service Backup zur Datensicherung betrachtet werden.

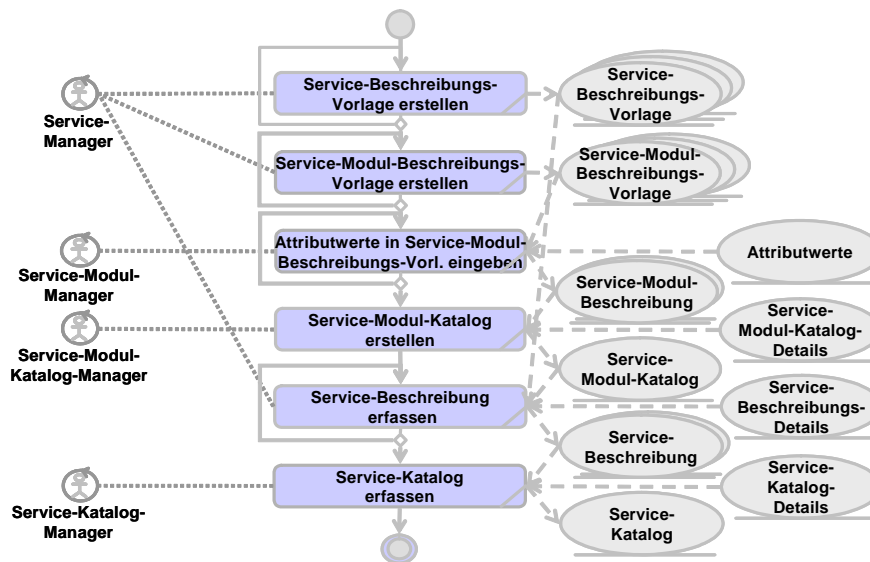


Abbildung 1 – Aktivitäten der Service-Design-Methode

Ausgehend von einer zeitlich vorangegangenen Analyse der grundlegenden Anforderungen an IT-Services, in welcher der IT-Dienstleister ein Verständnis über die Bedürfnisse und die Sichtweise der Dienstnehmer aufbaut, erstellt der Service-Manager mit den gewonnenen Erkenntnissen Vorlagen für eine einheitliche und strukturierte Beschreibung der IT-Services (vgl. Abbildung 1). Bei der Erstellung der Vorlagen ist zu berücksichtigen, dass die späteren, darauf aufbauenden Service-Beschreibungen von Dienstnehmern verstanden werden, die beschriebenen IT-Services einen Mehrwert für den Dienstnehmer liefern, die versprochene Funktionalität und Qualität überwachbar ist, für jeden Service ein Verantwortlicher benannt wird, die Inanspruchnahme eines Service einem intuitiven Ablauf folgt und mit einem klar definierten Preis ausgezeichnet werden kann. Für die Beschreibung eines Service werden in der Service-Beschreibungs-Vorlage Attribute definiert, die sich in die folgenden Attributgruppen einteilen lassen:

- **Identifikationsattribute** identifizieren einen IT-Service innerhalb des Gesamtangebots eines IT-Dienstleisters eindeutig: Name, ID, Version, Varianten-Name usw.
- **Statusattribute** geben Auskunft über die Vorlaufzeit für die Bereitstellung und den Zustand eines IT-Service im Rahmen seines Lebenszyklus: in Planung, Pilotphase, in Produktion, in Ablösung, im Archiv usw.
- **Organisationsattribute** beschreiben organisatorische Randbedingungen rund um die Erbringung eines IT-Service: Verantwortlicher, Stellvertreter, Organisationseinheit usw.
- **Leistungsattribute** spezifizieren die (funktionale) Leistung eines IT-Service: Kurz- und Langbeschreibung, Leistungseinheit, Lieferergebnis, Voraussetzungen, benötigte Service-Module, Ausschlüsse, Risiken usw.
- **Qualitätsattribute** beschreiben die Qualität der Dienstleistung: Verfügbarkeit, Speicherkapazität, Reaktionszeit usw.
- **Kostenattribute** zeigen die Kosten bzw. den Preis eines IT-Service bzw. dessen Inanspruchnahme auf: Zielkostensatz, Verrechnungspreis, Einrichtungskosten, Kostenmodell, Kostentreiber usw.

Um die Service-Beschreibung auf ein technisches und ablauforganisatorisches Fundament zu stellen, erhebt der Service-Manager Informationen über die Teilleistungen, sog. Service-Module, die in einen IT-Service und dessen Service-Beschreibung eingehen. Aufgrund der Verteilung und Komplexität heutiger IT, fallen Informationen über die Teilleistungen ebenso verteilt z.B. über unterschiedliche organisatorische Abteilungen an. Zudem lassen sich an der Technologie und an herstellereinspezifischen Produkten ausgerichtete Organisationseinheiten für die Bereiche Netzwerke, (Server-) Systeme und Anwendungssoftware erkennen. Um weitestgehend einheitliche Service-Modul-Beschreibungen zu erhalten, erstellt der Service-Manager Vorlagen für die Service-Modul-Beschreibungen. Da Service-Modul-Beschreibungen zu Service-Beschreibungen verarbeitet werden, bauen die beschreibenden Attribute eines IT-Service auf den Attributen von Service-Modulen auf und ähneln sich in ihrer Struktur. Zusätzlich muss ein Service-Modul als eigenständige Einheit überwachbar und ganzheitlich im Kontext eines Service genutzt werden können.

Mit Hilfe einer Service-Modul-Beschreibungs-Vorlage beschreibt ein Service-Modul-Manager z.B. innerhalb seiner Abteilung die für einen IT-Service notwendige Teilleistung. Der Beispiel-IT-Service Backup setzt sich aus den Service-Modulen Backup-Server aus dem Bereich Server-Systeme und Backup-Zugang aus dem Bereich Netzwerke zusammen. Durch die Belegung der durch die Vorlage abgefragten Attribute mit Attributwerten nimmt der jeweils zuständige Service-Modul-Manager eine abstrakte Beschreibung der technischen Ressourcen und der ablauforganisatorischen Teilprozesse als Service-Modul vor. Die abgegebenen Attributwerte definieren die vorhandenen Leistungsmöglichkeiten einzelner Bereiche des IT-Dienstleisters, z.B. technologischer Bereiche oder Abteilungen.

Fertig gestellte Service-Modul-Beschreibungen werden in einem Service-Modul-Katalog zusammengeführt, der in seiner Gesamtheit die technischen und ablauforganisatorischen Leistungsmöglichkeiten eines IT-Dienstleisters beschreibt. Der Service-Modul-Katalog wird anhand von Kriterien basierend auf den Attributen und Attributwerten der Modul-Beschreibungen zur Weiterverarbeitung strukturiert und durch

Service-Modul-Katalog-Details, wie z.B. Gültigkeit, Geschäftsjahr usw. näher spezifiziert. Sinnvolle und technisch realisierbare Modul-Kombinationen spannen zudem den Raum für Service-Varianten auf.

Die Service-Modul-Beschreibungen im Service-Modul-Katalog stellen das vorhandene Fundament für die Definition von IT-Services dar. Mit der technischen Installation, Konfiguration und Integration der IT geht die Zusammenführung der beteiligten Service-Modul-Beschreibungen zu IT-Service-Beschreibungen einher. Dabei müssen einzelne Attributwerte der Service-Beschreibung je nach Attributgruppe aus den Attributwerten der beteiligten Service-Modul-Beschreibungen berechnet, zusammengesetzt oder als Service-Beschreibungs-Details individuell hinzugefügt werden. Vorschriften für Identifikationsattribute des IT-Service sind allenfalls für die Art und Weise der Bezeichnung festzulegen. So kann sich ein Service-Varianten-Name des IT-Service Backup an den Qualitätsattributen seiner Service-Module, z.B. Speicherkapazität des Moduls Backup-Server in Verbindung mit Datendurchsatz des Moduls Backup-Zugang, orientieren. Auf jeden Fall müssen die Identifikationsattribute eindeutig gegenüber den Dienstnehmern definiert werden. Statusattribute verschiedener Service-Module können aufeinander abgebildet werden. Beispielsweise kann sich die Vorlaufzeit des IT-Service Backup aus der größten Vorlaufzeit der beteiligten Service-Module ergeben, falls beide Module weitestgehend parallel eingerichtet werden können. Der Zustand des konstruierten IT-Services entspricht dem Zustand mit der geringsten Qualitätssicherung der verbauten Service-Module; eine Verarbeitung eines Service-Moduls im Zustand in Ablösung oder im Archiv ist nicht mehr möglich, wohl aber im Zustand Pilotphase. Organisationsattribute müssen, ähnlich wie Identifikationsattribute, individuell festgelegt werden. Die Beschreibung der Leistung eines IT-Service kann aus der Summe der Teilleistungen der beteiligten Service-Module gewonnen werden. Die Korrelation der Qualitätsattribute von Service-Modulen zu Qualitätsattributen von IT-Services ist bezogen auf die in den SLAs auszusprechende Garantie der Attributwerte ein nicht trivialer Aspekt. Neben der Berechnung von Qualitätsattributwerten nach komplexen Korrelationsvorschriften hat sich die Zusammenführung von an Service-Modulen gemessenen Erfahrungswerten, z.B. nach statistischen Prinzipien, bewährt. Hierbei können Ansätze zur Anwendung kommen, die den schlechtesten bzw. geringsten Wert als den maximal garantierbaren Wert darstellen (pessimistischer Ansatz). Als Erfahrungswerte gewonnene Verfügbarkeiten der Service-Module Backup-Service von 99,9% und Backup-Zugang von 95% führen beispielsweise zu einer garantierbaren Verfügbarkeit des IT-Service Backup von 95%. Qualitätsattribute, wie z.B. Speicherkapazität des Backup-Servers in Verbindung mit Datendurchsatz des Backup-Zugangs, deren Werte sich nicht unmittelbar miteinander vergleichen oder aufeinander abbilden lassen, können für eine erhöhte Transparenz der verbauten Service-Module nebeneinander in die Service-Beschreibung aufgenommen werden. Die Kostenattribute des IT-Service werden nach betriebswirtschaftlichen Gesichtspunkten aus den Kosten der Service-Module berechnet.

In der letzten Aktivität der Service-Design-Methode werden die IT-Service-Beschreibungen in einem Service-Katalog zusammengefasst. Der Service-Katalog ist in erster Linie für die Auswahl der IT-Services und die SLA-Verhandlungen mit den Dienstnehmern ausgelegt. Entsprechende Strukturierungs- und Sortiermöglichkeiten,

z.B. nach bestimmten Attributen, werden durch Service-Katalog-Details festgelegt und vereinfachen die Suche nach gewünschten IT-Services.

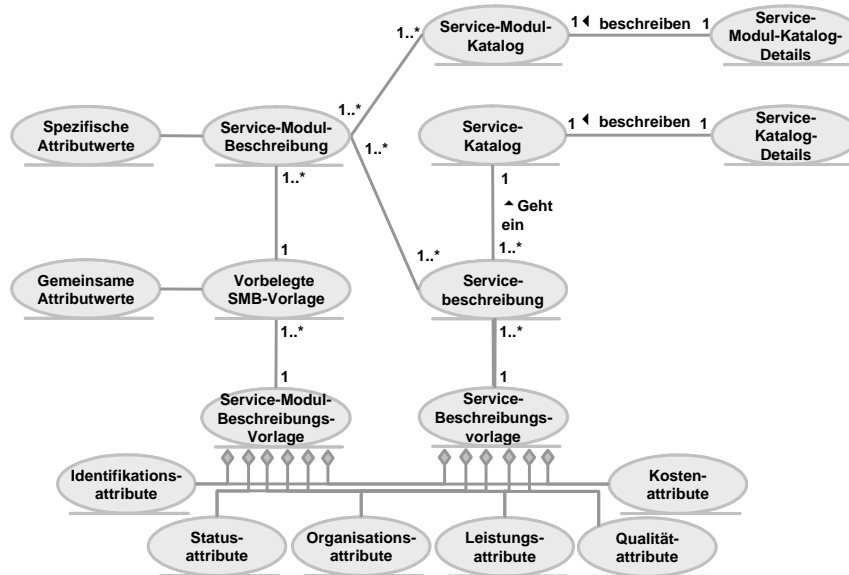


Abbildung 2 – Geschäftsobjekte im Kontext der Service-Design-Methode

Abbildung 2 fasst die Geschäftsobjekte zusammen, die im Kontext der Service-Design-Methode zum Einsatz kommen und die für die Entwicklung von unterstützenden Service-Management-Werkzeugen verfeinert und auf eine geeignete Werkzeug-Architektur abgebildet werden müssen.

4 Unterstützende Werkzeuge

In Analogie zum eTOM-Ansatz wurden für eine effiziente Anwendung der Service-Design-Methode unterstützende Service-Management-Werkzeuge entwickelt: Der Service-Katalog-Manager (SKM) unterstützt Aktivitäten von der Erhebung der Service-Modul-Beschreibungen bis hin zur Verwaltung von IT-Service-Beschreibungen im Service-Katalog. Der Anteil des Service-Katalogs, der für die Kommunikation mit den Dienstnehmern benötigt wird, wird über ein Service-Portal (SP) veröffentlicht, das zwischen IT-Dienstleister und den Dienstnehmern positioniert ist.

Die Benutzer des SKM lassen sich aus dem Aktivitätsdiagramm der Service-Design-Methode ableiten (vgl. Abbildung 1). Neben dem Service-Modul-Manager und dem Service-Manager werden die Rollen zweier Katalog-Manager unterschieden, die für die Katalogisierung der Service-Modul- und der Service-Beschreibungen zuständig sind.

Beispielhaft wird im Folgenden der Ablauf zum Verwalten der Service-Modul-Beschreibungen vertieft. Die Erstellung einer Service-Modul-Beschreibung kann in zwei Schritten erfolgen:

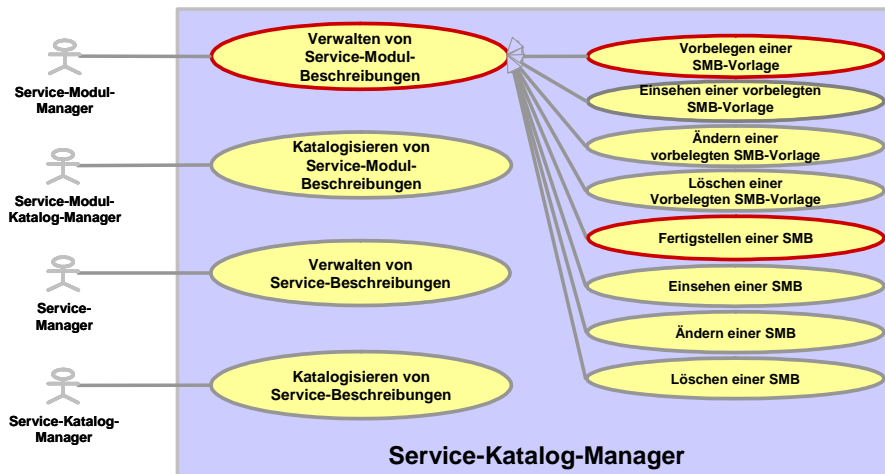


Abbildung 3 – Anwendungsfälle des Service-Katalog-Managers

Der Service-Modul-Manager – z.B. als Leiter einer Abteilung – belegt Attributwerte in der Service-Modul-Beschreibungs-Vorlage vor, die mehreren Modul-Beschreibungen gemeinsam sind. Dies können z.B. Organisationsattribute sein, die allen Beschreibungen einer Abteilung gleichen. Die Vorbelegung erhöht die Effizienz bei der Erstellung der Modul-Beschreibungen und sorgt für ein hohes Maß an Einheitlichkeit.

Eine individuelle Service-Modul-Beschreibung wird dann durch Angabe der spezifischen Attributwerte vervollständigt. Zu den spezifischen Attributen zählen neben eindeutigen Identifikationsattributen insbesondere Leistungs- und Qualitätsattribute, die sich z.B. aufgrund herstellereigenschaftlicher Eigenheiten der eingesetzten und zu betreibenden Technologien ergeben.

Die grundlegende Architektur des Service-Katalog-Managers (vgl. Abbildung 4) folgt einer Drei-Schichten-Architektur mit einer Trennung zwischen Präsentations-, Geschäftslogik- und Datenhaltungsschicht. Zusätzlich kommt das Prinzip eines *thin client* zum Einsatz. Beide Architekturprinzipien erhöhen die Flexibilität bei der Weiterentwicklung des SKM und vereinfachen dessen Integration in die Management-Werkzeug-Umgebung eines IT-Dienstleisters.

Die Aufgaben der SKM-Komponenten sind wie folgt zu verstehen: Die `Service-CatalogManagerBusinessProcessControl (SCMBPC)` koordiniert Geschäftsprozesse, die sich über mehrere Systemanwendungsfälle erstrecken. Wird von der SCMBPC ein Anwendungsfall wie etwa die Verwaltung von Service-Modul-Beschreibungen identifiziert, so wird die Abwicklung der anfallenden Aufgaben an die entsprechende Anwendungsfallsteuerungskomponente (`UseCaseControl, UCC`) delegiert. Diese steuert den Ablauf eines Anwendungsfalles, der während der Analyse durch ein anwendungsfallbeschreibendes Aktivitätsdiagramm festgelegt wurde. Um diesen Ablauf zu realisieren, kooperieren UCCs mit der Funktionalität der Dialogsteuerungskomponenten (`DialogControl, DC`) und der Fachkomponenten (`BusinessObject, BO`). DCs bilden die Brücke zwischen der Präsentations-

und Geschäftslogikschicht der Drei-Schichten-Architektur. Für jeden Hauptanwendungsfall (vgl. Abbildung 3) existiert eine eigene UCC: `SCMAdministrationOfSmdsUCC` für das Verwalten von Service-Modul-Beschreibungen, `SCMCatalogOfSmdsUCC` für das Katalogisieren von Service-Modul-Beschreibungen, `SCMAdministrationOfSdsUCC` für das Verwalten von Service-Beschreibungen, `SCMCatalogOfSdsUCC` für das Katalogisieren von Service-Beschreibungen.

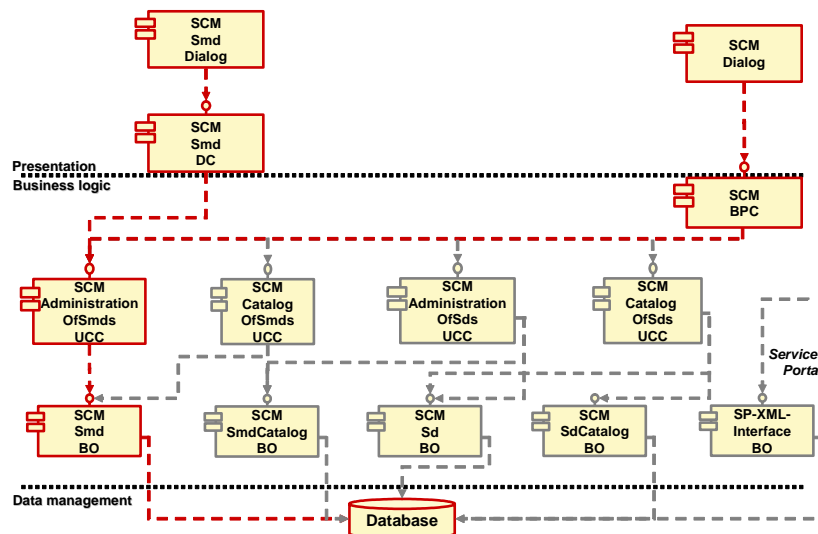


Abbildung 4 – Architektur des Service-Katalog-Managers

Die Fachkomponenten realisieren die für den entsprechenden Anwendungsfall benötigte, geschäftsobjektbezogene Funktionalität. Sie sind damit hauptsächlich für die Verarbeitung der Daten verantwortlich: Sei es, die von der Anwendungsfallsteuerungskomponente übergebenen Daten zu speichern, die von ihr geforderten Daten zu liefern oder sonstige Berechnungen durchzuführen. Ein weiterer Grund, dazu separate Komponenten in der Geschäftslogikschicht zu entwerfen, besteht darin, die Datenbank vom Rest der Anwendung abzukapseln und anderen Komponenten höherwertige Schnittstellen anzubieten. Dies hat den Vorteil, dass Zugriffe auf das Persistenzmedium nur in Fachkomponenten realisiert werden müssen und sich andere Komponenten nicht darum kümmern müssen, wie die Daten der Geschäftslogik entsprechend abgelegt oder wieder aufbereitet werden. Folgende Fachkomponenten wurden entworfen: `SCMSmdBO` und `SCMSmdCatalogBO` für das Verwalten und Katalogisieren der Service-Modul-Beschreibungen, `SCMSdBO` und `SCMSdCatalogBO` für das Verwalten und Katalogisieren der Service-Beschreibungen sowie `SPXMLInterfaceBO` als XML-konforme Schnittstelle.

Für jede Anwendungsfallsteuerungskomponente existiert eine entsprechende Fachkomponente, da jeder Anwendungsfall andere Funktionalitätsanforderungen hat und somit für jeden Fall eine eigene Fachkomponente modelliert werden muss. Dies bedeutet jedoch nicht, dass jede UCC nur eine Fachkomponente ansteuert. So nutzt die

Anwendungsfallsteuerungskomponente `SCMCatalogOfSmdsUCC` auch Funktionen der Fachkomponente `SCMSmdBO`. Als Beispiel ist hier das Einsehen der Service-Modul-Beschreibungen zu nennen.

In der Präsentationsschicht steuert die Dialogsteuerungskomponente komplexere Dialoge, die unabhängig von der Anwendungsfallsteuerung aufgebaut und gesteuert werden können. Die Hauptaufgaben dieser Komponente sind die Abfolge von Dialogen zu steuern sowie syntaktische Eingabevalidierungen durchzuführen. Bisher wurde beim Service-Katalog-Manager der Fokus auf die Dialogsteuerungskomponente für den Anwendungsfall „Verwalten von SMB“, `SCMSmdDC`, gelegt. Mit den Dialogkomponenten werden die Dialogoberflächen realisiert. Sie bilden die Schnittstelle zu den SKM-Benutzern. Die Eingaben des Benutzers werden durch die Dialogkomponente an die ihr bekannte Dialogsteuerungskomponente weitergeleitet. Die Komponente `SPXMLInterfaceBO` realisiert eine definierte Schnittstelle, über die andere Werkzeuge auf Daten und Funktionen des SKM zugreifen können. Eines dieser Werkzeuge ist das Service-Portal (SP), das u.a. Informationen über das Service-Angebot des IT-Dienstleisters zur Verfügung stellt. Neben dem SP-Administrator und einem Content-Manager, der über den Inhalt des Portals wacht, wird das SP vor allem durch die (potenziellen) Dienstnehmer genutzt, die an den Service-Beschreibungen interessiert sind. Eine wesentliche Aufgabe des SPs im Zusammenhang mit der Service-Design-Methode ist die Veröffentlichung des für Dienstnehmer gedachten Anteils des Service-Katalogs.

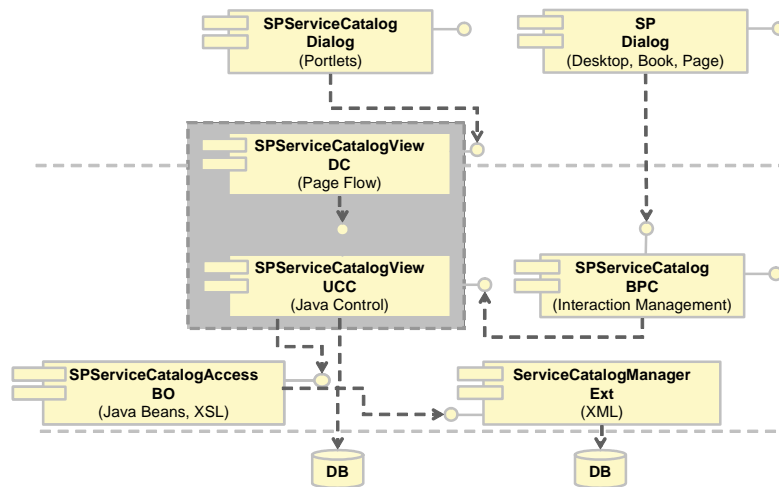


Abbildung 5 – Architektur des Service-Portals (SP)

Wie der SKM folgt auch das Service-Portal (SP) einer Drei-Schichten-Architektur (vgl. **Abbildung 5**). Neben Dialog- und Dialogsteuerungskomponenten verfügt auch das SP über eine UCC, die den Dienstnehmern den Zugriff auf die Inhalte des Service-Katalogs ermöglicht: `SPServiceCatalogViewUCC`. Um redundante Datenhaltung und damit Konsistenzprobleme zu vermeiden, werden innerhalb des Portal-Systems ausschließlich Portal-spezifische Daten, wie z.B. Rollen und Zugriffsrechte

usw., gespeichert. Die Service-Beschreibungen werden über eine Zugriffskomponente (*SPServiceCatalogAccessBO*) realisiert, die auf die Schnittstelle des SKM (*ServiceCatalogManagerExt*) zugreift. Für die Darstellung der Inhalte des Service-Katalogs werden die Struktur des Katalogs an sich, mit deren Hilfe ein Dienstnehmer im Katalog navigieren kann, sowie nach Auswahl eines IT-Service dessen Beschreibung vom SKM an das SP übertragen. Der Austausch der Daten an dieser Schnittstelle erfolgt mittels XML.

Die SKM-Implementierung wurde mit Hilfe der J2EE-Technologie realisiert. Der Prototyp wird auf einem *Sun Java System Application Server 8* betrieben. Als Entwicklungsumgebung kam die IDE *Eclipse* in Verbindung mit dem UML-Modellierungswerkzeug *Enterprise Architect 3.6* zum Einsatz. Für die Implementierung des Service-Portals hingegen stand im Szenario die Portal-Entwicklungsumgebung von BEA zur Verfügung. Beide Technologie-Umgebungen wurden für die Implementierung der Prototypen beibehalten, um die Integrationsfragestellung zu untersuchen und zu lösen.

5 Zusammenfassung und Ausblick

Die Service-Design-Methode sowie die unterstützenden Management-Werkzeuge wurden in Kooperation mit IT-Dienstleistern entwickelt, deren Ziel die Bereitstellung qualitätsgesicherter Netz-, System- und auch Anwendungsdienste ist. Ausgehend von der allgemeinen Konzeption der Service-Design-Methode wurden Konkretisierungen der Methode in den jeweiligen Szenarien vorgenommen. Unter Berücksichtigung Szenarien-abhängiger Anforderungen wurden Prototypen der prozessunterstützenden Werkzeuge als Machbarkeitsstudien der Methode entwickelt.

Eine wesentliche Erkenntnis der Arbeiten im Zusammenhang mit der Service-Design-Methode ist die Tatsache, dass es sich beim Entwurf von SLA-fähigen IT-Services um einen grundlegenden Geschäftsprozess verteilter und kooperierender Rollen innerhalb von IT-Dienstleistern handelt. Ziel der Methode ist die Beschreibung von IT-Services auf der Basis aufgezeigter bestehender Ansätze, die einerseits ein technisches und ablauforganisatorisches Fundament in den Leistungsmöglichkeiten des IT-Dienstleisters finden und andererseits als Ausgangspunkt für SLA-Verhandlungen dienen können, deren Erfüllung nach Abschluss auch überwacht und nachgewiesen werden kann.

Neben einer Verfeinerung der Methode, insbesondere im Bereich der Zusammenführung von Service-Modulen zu IT-Services, steht die Einbettung der Service-Design-Methode in den sonstigen Betrieb eines IT-Dienstleisters an und hier vor allem die Identifikation von Querbeziehungen zu anderen Betriebsprozessen. Die Definition von ablauforganisatorischen wie auch technischen Schnittstellen soll zur Schärfung der Methode selbst beitragen und vermindert Akzeptanzprobleme bei Mitarbeitern des IT-Dienstleisters. Beispielsweise werden Zusammenhänge zu Prozessen des Konfigurationsmanagements untersucht, innerhalb derer bereits die Konfiguration zu betreibender IT-Ressourcen mittels standardisierter Informationsmodelle, wie dem *Common Information Model (CIM)* [3], beschrieben werden [12]. An der Schnittstelle hin zum Dienstnehmer werden auf Basis des Service-Katalogs die Prozesse im Bereich *Service Level Management (SLM)* weiter ausgebaut.

[10] beschreibt eine integrierte Service-Management-Architektur (SMA), die neben IT-Management-Komponenten zur Überwachung und Steuerung der zu betreibenden IT auch Komponenten zur Unterstützung der kooperativen Betriebsprozesse sowie des IT-Service-Managements vorsieht. Die in diesem Beitrag vorgestellten Management-Werkzeuge sollen in eine SMA-konforme Gesamtinfrastruktur eines IT-Dienstleisters integriert werden. Da sich ausgehend von den sich ändernden Anforderungen an die IT-Services auch die Betriebsprozesse und mit ihnen die Anforderungen an die einzusetzenden Management-Werkzeuge ändern, soll die Integration der einzelnen Management-Werkzeuge über eine lose Kopplung der sonst selbständigen Werkzeuge bewerkstelligt werden. Dabei soll untersucht werden, inwieweit sich das Paradigma Service-orientierter Architekturen (SOA) auch auf die Entwicklung und Integration von Management-Werkzeugen anwenden und etablieren lässt.

6 Literatur

1. T. Böhmman, M. Junginger, H. Krcmar: Modular Service Architectures – A Concept and Method for Engineering IT services, International Conference on System Sciences (HICSS'03), Hawaii, 2003.
2. J. v. Bon, G. Kemmerling, D. Pondman: IT Service Management – An Introduction, Van Haren, 2002.
3. Distributed Management Task Force (DMTF): Common Information Model (CIM) Specification: Version 2.2, 1999; CIM Schema: Version 2.8.1, 2004.
4. G. Dreo Rodosek, A Generic Model for IT Services and Service Management. Symposium on Integrated Network Management (IM 2003), Colorado Springs, 2003.
5. G. Dreo Rodosek, H.-G. Hegering: IT-Dienstmanagement, Herausforderungen und Lösungsansätze, PIK, K.G. Saur, 2004.
6. enhanced Telecom Operations Map (eTOM) – The Business Process Framework For The Information and Communications Services Industry, GB921 v4.0, 2004.
7. H.-G. Hegering, S. Abeck, B. Neumair: Integrated Management of Networked Systems: Concepts, Architectures, and Their Operational Application: Morgan Kaufmann, 1999.
8. A. Keller, H. Ludwig: Defining and Monitoring Service Level Agreements for dynamic e-Business, System Administration Conference (LISA 2002), Philadelphia, 2002.
9. C. Mayerl, S. Abeck, M. Becker, A. Köppel, O. Mehl, B. Pauze: Dienstbeschreibung und –modellierung für ein SLA-fähiges Service-Management, KIVS 2003.
10. C. Mayerl, Z. Nochta, M. Müller, M. Schauer, A. Uremovic, S. Abeck: Specification of a Service Management Architecture to Run Distributed and Networked Systems, Trends towards a Universal Service Market (USM'2000), München, 2000.
11. J. McConell: Practical Service Level Management – Delivering High-Quality Web-Based Services, Cisco Press, 2004.
12. O. Mehl, M. Becker, A. Köppel, P. Paul, D. Zimmermann, S. Abeck: A Management-Aware Software Development Process Using Design Patterns, Symposium on Integrated Network Management (IM 2003), Colorado Springs, 2003.
13. Office of Government Commerce (OCG): IT Infrastructure Library (ITIL) – Service Support (ISBN 0113300158), 2000; Service Delivery (ISBN 0113300174), 2001; Planning to Implement Service Management (ISBN 0113308779), 2002; Application Management (ISBN 0113308663), 2002.
14. R. Sturm, W. Morris, M. Jander: The Heart of the Matter: Service Level Agreements, 2001.
15. TeleManagementForum: TeleManagementForum-Portal, <http://www.tmforum.org>, 2004.