



Download-Anhang 11 zum Buch Lean IT-Management

11. Ableitung von Business-Services

„Veränderungen begünstigen nur den, der darauf vorbereitet ist.“

Louis Pasteur (1822 – 1895), französischer Chemiker und Mikrobiologe

Die IT muss sich auf Veränderungen in den Geschäftsmodellen und kürzer werdende Innovations- und Produktlebenszyklen sowie Merger & Akquisitionen, Outsourcing und Umstrukturierungen vorbereiten. Neue oder veränderte Geschäftsanforderungen müssen schnell und in hoher Qualität umgesetzt werden.

Dies ist nicht so einfach, wenn die IT-Landschaft „historisch gewachsen“ ist. Schon kleine Änderungen können in dem komplexen Gesamtsystem verheerende Auswirkungen haben. Um flexibel und schnell auf neue Anforderungen reagieren zu können, müssen Änderungen möglichst lokal an wenigen Stellen durchgeführt werden können. Änderungen an Geschäftsregeln oder Workflows sollten einfach modelliert oder konfiguriert werden können. Eine modulare IT-Landschaft mit einer Integrationsarchitektur ist hierzu erforderlich.

Wichtig

Wenn Sie Ihre IT auf Veränderungen im Business ausrichten wollen, dann müssen Sie diese in den relevanten Bereichen „komponentisieren“ und mit einer Integrationsarchitektur versehen.

Die IT-Landschaft muss so in Komponenten zerlegt werden, dass fachlich zusammengehörige Funktionalitäten und Daten in einer Komponente angesiedelt sind. Die

Komponenten sind untereinander lose über z.B. eine Regel-, Workflow-Engine oder ein Enterprise Service Bus gekoppelt.

Eine Integrationsarchitektur liefert unternehmensspezifische Vorgaben für die serviceorientierte Umsetzung von Geschäftsanforderungen. Hierzu zählen Technologie-, Softwarearchitektur- und Infrastrukturaspekte für Entwicklung, Betrieb und Governance der involvierten Einzelsysteme und deren Zusammenspiel (End-to-end).

Beispiele hierfür sind Architekturvorgaben für die lose Kopplung von Komponenten über einen ESB (Enterprise Service Bus) oder aber die Herauslösung der Geschäftsregeln und Ablaufsteuerung aus dem Programmcode und die Hinterlegung dieser in einer Rules Engine und einem BPMS (Business Process Management System). Änderungen an Geschäftsregeln (zum Beispiel veränderte Preisberechnung) und Abläufen (zum Beispiel Änderung des Genehmigungsverfahrens) ziehen keine Auswirkungen an den funktionalen Modulen nach sich. Funktionale Module, wie zum Beispiel Provisionsabrechnung, können einfach ersetzt werden. Die Änderung erfolgt lokal und kann schnell und kostengünstig durchgeführt werden, da nur reduzierte Integrations- und Testaufwände anfallen.

Die Komponenten, häufig auch Services genannt, sollten sich an den Business Capabilities des Unternehmens ausrichten. Beispiele für Services sind Vertriebscontrolling oder Provisionsabrechnung.

Gewachsene IT-Landschaften mit vielen funktionalen und Datenredundanzen, vielen und komplexen spezifischen Schnittstellen, heterogenen Technologien sowie vielen Inkompatibilitäten und Abhängigkeiten müssen „aufgeräumt“ („SOA-fiziert“) werden, um Änderungen schnell, kostengünstig und risikoarm durchzuführen und in Betrieb zu nehmen. So wird die erforderliche Flexibilität (Anpassungsfähigkeit) erzielt und darüber hinaus werden die Wartungs- und Betriebsaufwände und damit die IT-Kosten nachhaltig gesenkt.

Wichtig

Serviceorientierte Architekturen (SOA) versprechen Agilität in der Veränderung des Geschäftsmodells, indem Business-Services zu Geschäftsprozessen flexibel orchestriert und damit einfach und kostengünstig an veränderte Geschäftsanforderungen oder Randbedingungen angepasst werden können. Sie zielen auf die Komponentisierung der IT-Landschaft in fachliche Komponenten und eine flexible Kopplung der Komponenten ab.

SOA (serviceorientierte Architektur) hat eine fachliche und eine technische Dimension:

Fachliche Funktionen (auch Business Capability oder Geschäftsfähigkeit genannt) als inhaltliche Implementierungs-, Strukturierungs- und Granularitätsvorgabe für Business-Services (siehe Abschnitt 3.1.4 im Buch)

Technische Umsetzung mittels einer für das Unternehmen standardisierten Integrationsarchitektur (u.a. lose Kopplung und Kontrakte) mit Integrationstechnologien, wie z.B. WebServices, Workflow und Rule Engines oder Enterprise Service Bus

Das Business Capability Management (siehe Abschnitt 3.5.4 im Buch) leistet hierzu einen wesentlichen Beitrag. Die Business Capabilities geben fachliche Bezugspunkte für die Planung und Steuerung der IT-Umsetzung vor. Das Anwendungsportfolio wird in Bezug auf die aktuellen und zukünftigen Geschäftsfähigkeiten optimiert. Hierbei wird die Soll-IS-Landschaft entsprechend der zukünftigen Business Capability Map komponentisiert, das heißt serviceorientiert gestaltet. Auf dieser Basis können mit Hilfe einer Integrationsarchitektur neue Abläufe einfach durch Zusammensetzen und Orchestrieren der Business-Services konfiguriert werden. Die IT kann so schneller auf sich verändernde Anforderungen reagieren. Die Orchestrierung der Business-Services, also deren Abfolge und Wechselwirkungen, wird durch die Geschäftsprozesse vorgegeben. Lassen sich die einzelnen Business-Services einfach miteinander verknüpfen, dann lassen sich auch Änderungen in den Prozessabläufen, unter der Annahme des Einsatzes derselben Services, schnell und ohne Programmieraufwand umsetzen.

In Bild 11.1 finden Sie ein Beispiel für eine Soll-Informationssystemlandschaft, die entsprechend der Business Capabilities serviceorientiert gestaltet wurde. Die Business Capability Map bildet die Zielvorgabe für das IT-Service-Modell der zukünftigen IT-Landschaft.

Fachlich zusammenhängende Business Capabilities (fachliche Domäne in Bild 11.1) werden in der idealisierten Zielvorstellung durch ein Informationssystem bereitgestellt. Die Business Capabilities geben die Strukturierung für die Soll-Informationssysteme vor. Die Soll-Informationssysteme werden entsprechend der Business Capabilities in Teil-Informationssysteme, auch IT-Funktionalitäten genannt, zerlegt. Für jede Business Capability existiert eine IT-Funktionalität (ein Service eines Informationssystems), der direkt eingebunden werden kann.

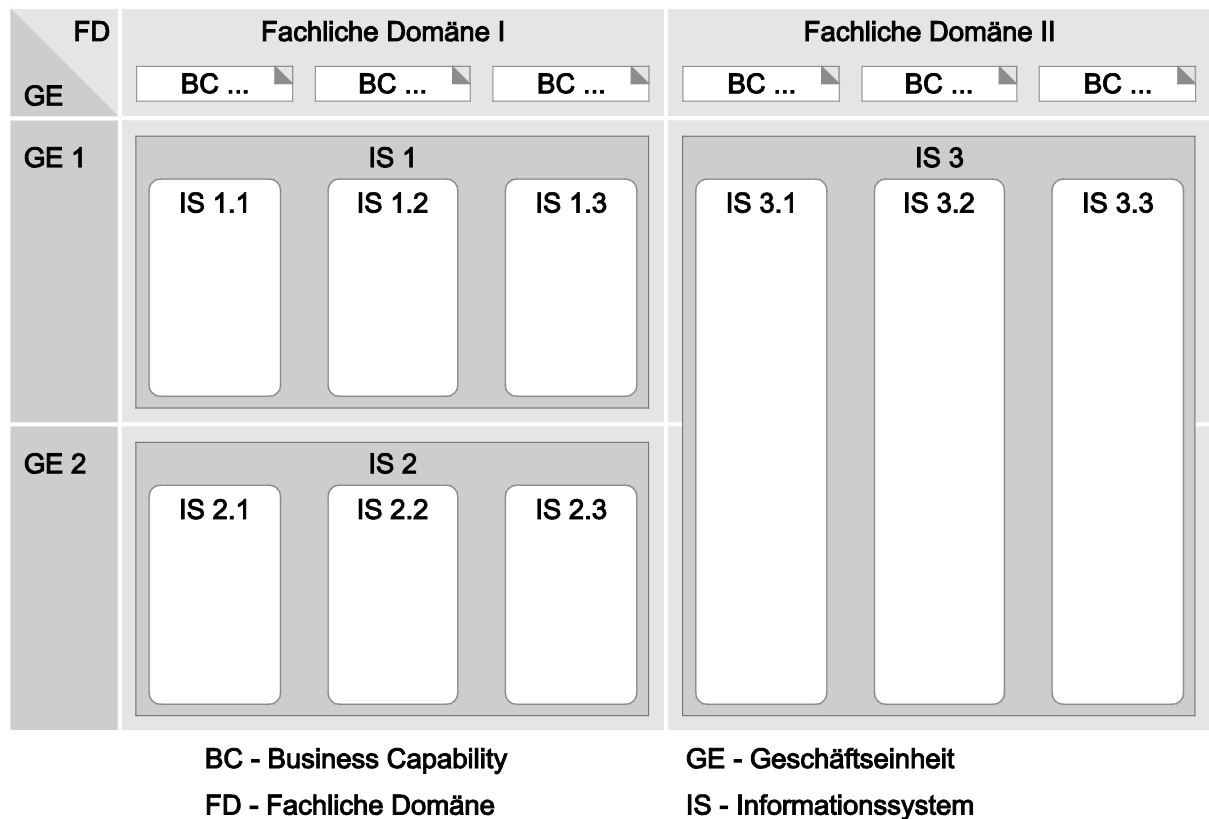


Bild 11.1 Serviceorientierte Gestaltung der IS-Landschaft

Die Gestaltung der zukünftigen IT-Landschaft erfolgt im Rahmen der Bebauungsplanung (siehe Download-Anhang 9) auf der Basis der Business Capability Map entsprechend der strategischen Vorgaben und Randbedingungen. Durch die strategischen Vorgaben und Randbedingungen können zum Beispiel, wie in Bild 11.1 dargestellt, die Informationssysteme für die verschiedenen Geschäftseinheiten für die fachliche Domäne I durchaus unterschiedlich ausgeprägt werden.

Durch die „Komponentisierung“ der IS-Landschaft entsprechend des funktionalen Referenzmodells und das Einziehen einer Integrationsarchitektur mit zum Beispiel einer Regel- und Workflow-Engine können Änderungen an Geschäftsregeln oder Geschäftsabläufen einfach modelliert und konfiguriert werden. So sieht zumindest das Zielszenario aus, um Business-Veränderungen schnell umsetzen zu können. Für die Vorbereitung der IT bedarf es aber in der Regel eines langen Atems.

Wichtig

Business Capabilities und ihre Verknüpfung mit IT-Strukturen gepaart mit einer Integrationsarchitektur sind ein wichtiger Schritt in Richtung einer flexiblen IT.

So können funktionale Redundanzen in der Geschäftsunterstützung beseitigt und vor allen Dingen die IT auf Veränderungen im Business vorbereitet werden. Serviceorientierung in Business und IT und die konsequente Nutzung einer Integrationsarchitektur sind der Schlüssel dazu. Weitere Informationen zu Integrationsarchitekturen und Hilfestellungen für die Komponentisierung finden Sie in Abschnitt 3.5.4 im Buch.

Serviceorientierung ist aber kein Selbstzweck und nicht immer adäquat. Durch die Zerschlagung und lose Kopplung von Komponenten werden Performance- und Sicherheitsanforderungen zum Teil schwieriger zu erfüllen. Viele Layer und Komponenten-Schnittstellen sind zu überwinden (siehe SOA-Referenzarchitektur in Abschnitt 3.5.4 im Buch). Zudem dauert die Migration einer gewachsenen IT-Landschaft zur SOA-Landschaft viele Jahre und verschlingt große Summen (siehe [Sla11] und [Eng08]).

Empfehlung

Identifizieren Sie mögliche und sinnvolle Anwendungsbereiche für die Flexibilisierung der IT. Untersuchen Sie hierzu Ihre Unternehmens- und IT-Strategie sowie Ihre Geschäftsarchitektur und deren IT-Unterstützung. Analysieren Sie Ihre Geschäftsanforderungen im Hinblick auf die geforderte Business-Agilität. Erstellen Sie ein funktionales Referenzmodell für Ihr Unternehmen (siehe Abschnitte 6.6 im Buch) und legen Sie für die verschiedenen fachlichen Domänen fest, ob und in welcher Art und Weise die Serviceorientierung zukünftig Einzug halten soll (Service-Strategie).

Business Capabilities (fachliche Funktionen) und ihre Umsetzung als Business-Services bieten darüber hinaus noch folgende weitere Vorteile:

- Fundierte Sourcing-Entscheidungen sind möglich. Anhand der festgelegten fachlichen Funktionen und deren Klassifikation bezüglich Wettbewerbsdifferenzierung kann entschieden werden, welche Fähigkeiten vom Unternehmen selbst erbracht oder zugekauft werden sollen. Zudem lässt sich das Anforderungsprofil für die Dienstleister konkreter festlegen.
- Fachliche Funktionen sind für die Abstimmung mit Stakeholdern im Anforderungsmanagement geeignet, weil sie die Funktionalität des zukünftigen Produkts beschreiben.
- Fundierter Input für Business-Transformationen, u.a. Merger und Akquisitionen, wird bereitgestellt (siehe Einsatzszenario „Business-Transformation“).

- Projekte lassen sich anhand der fachlichen Funktionen klassifizieren und bewerten.
- Die Inkrementbildung bei Projekten wird unterstützt. Die Priorisierung und die Zerlegung in Inkremente können anhand der fachlichen Funktionen (gegebenenfalls auf Teilfunktionsebene) erfolgen.

Die Ableitung von Business-Services erfolgt in der Regel innerhalb von Projekten auf einem niedrigen und schon sehr lösungsorientiertem Granularitätsgrad. Aufgrund der Bedeutung in der Praxis möchten wir dieses Thema trotzdem hier adressieren.

Hintergrund für die Serviceorientierung und damit einhergehend die Ableitung von Business-Services sind Agilitätsanforderungen. Unternehmen müssen ihr Geschäftsmodell schnell an veränderte Marktanforderungen anpassen können, um wettbewerbsfähig zu bleiben. Beispiele hierfür sind die Veränderung des Produkt- und Dienstleistungsangebots, die Art der Kundenansprache und Kundenbindung sowie Fusionen, Akquisitionen oder aber die Einbindung externer Partner und das Outsourcing von einzelnen Unternehmensbereichen.

Voraussetzung für Business-Agilität ist eine flexible IT. Serviceorientierung sowie Integrationsarchitekturen sind wichtige Mittel zur Umsetzung. Die Geschäftsfunktionen werden als Business-Services gekapselt und zu Geschäftsprozessen orchestriert (siehe Abschnitt 3.5.3 im Buch). Bei Veränderungen in Geschäftsanforderungen werden neue Business-Services ermittelt oder bestehende modifiziert, bereitgestellt und in die geschäftlichen Abläufe integriert. So kann sich das Unternehmen auf die sich rasch verändernden Geschäftsanforderungen vorbereiten und die bestehenden Geschäftsprozesse schrittweise verbessern.

Empfehlung

Die Flexibilisierung der IT, wenn nicht von Anfang an, so angegangen, ist teuer und langwierig. Daher müssen Sie sorgfältig mögliche und sinnvolle Anwendungsbereiche auswählen. Untersuchen Sie hierzu Ihre Unternehmens- und IT-Strategie sowie Ihre Geschäftsarchitektur und deren IT-Unterstützung. Analysieren Sie Ihre Geschäftsanforderungen im Hinblick auf die geforderte Business-Agilität. Erstellen Sie ein funktionales Referenzmodell für Ihr Unternehmen (siehe Abschnitt 6.6 im Buch) und legen Sie für die verschiedenen fachlichen Domänen fest, ob und in welcher Art und Weise Serviceorientierung zukünftig Einzug halten soll (Service-Strategie).

11.1 Ermittlung der Agilitätsanforderungen

Für die Analyse der Geschäftsanforderungen in Hinblick auf Business-Agilität hat sich in vielen Projekten die Klassifikation der fachlichen Funktionen entsprechend ihrer Veränderungsdynamik und Wettbewerbsdifferenzierung bewährt (siehe Abschnitt 3.3.2 im Buch). In Bild 11.2 finden Sie eine Portfolio-Grafik, in der fachliche Funktionen nach diesen Kriterien klassifiziert und darüber für die Quadranten Strategien für die IT-Unterstützung gegeben werden. Serviceorientierung wäre hier nur für den rechten oberen Quadranten, der mit "Agil" gekennzeichnet ist, als IT-Umsetzungsstrategie relevant. In Bild 11.3 finden Sie hierzu ein Beispiel.

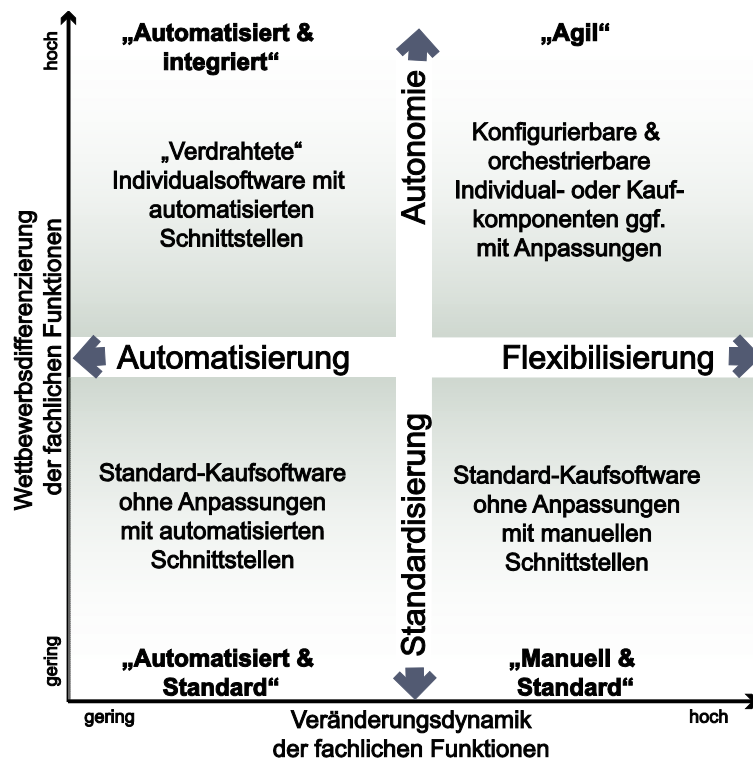


Bild 11.2 Portfolio zur Ermittlung der Agilitätsanforderungen

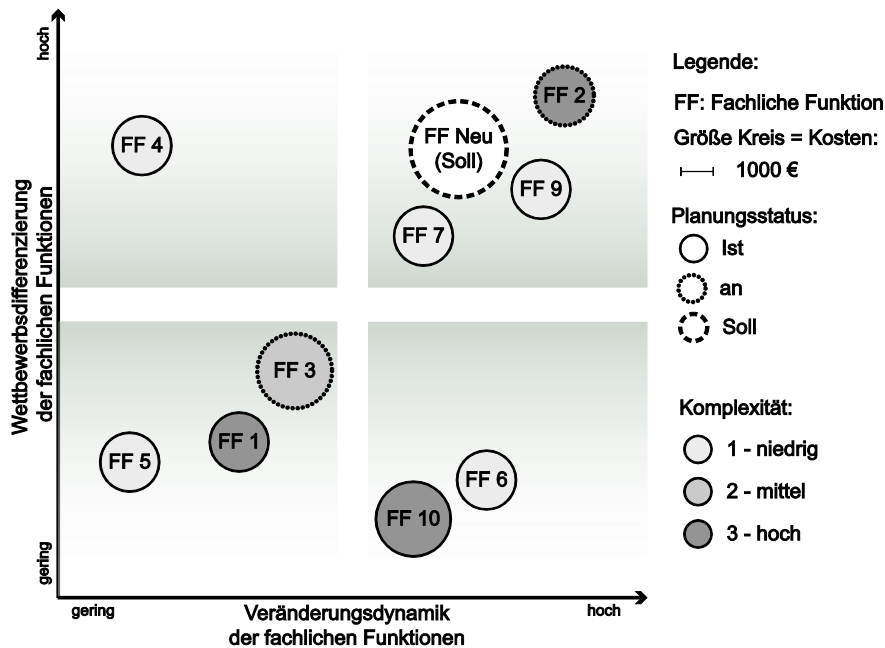


Bild 11.3 Beispiel für Informationssystemstrategie in Abhängigkeit von der Klassifikation der unterstützten fachlichen Funktionen

In Bild 11.4 finden Sie ein weiteres Beispiel, wie sich die Anwendungsbereiche für die Serviceorientierung charakterisieren lassen. Hier wird ein vereinfachtes fachliches Domänenmodell (siehe Download-Anhang 13) aus dem Bankenbereich mit Angabe der Service-Strategie für die verschiedenen fachlichen Domänen abgebildet. Eine fachliche Domäne ist hierbei eine fachliche Einteilung, die kohärentes Wissen und Vorgehensweisen bündelt.

- Die **fachliche Domäne Vertrieb** ist aufgeteilt in die Frontoffice-Applikationen und die Backoffice-Multikanal-Vertriebsunterstützung. Für die Frontoffice-Applikationen für Privatkunden, privilegierte Privatkunden und Geschäftskunden wird keine Service-Strategie vorgegeben, da diese im Beispiel in der Hoheit von anderen Unternehmen liegen und zudem durch verschiedene Kaufprodukte beziehungsweise fremdentwickelte Zugangsportale umgesetzt sind. Deshalb lässt sich die Gestaltung kaum beeinflussen. Lediglich CI- und Integrationsanforderungen werden gestellt, so dass einerseits das User Interface der jeweiligen Frontoffice-Applikation entsprechend der unternehmensspezifischen Vorgaben gestaltet ist und andererseits definierte Protokolle und Technologien für die Kommunikation mit dem Backoffice genutzt werden.

In der Multikanal-Vertriebsunterstützung werden die verschiedenen Dienstleistungen über die verschiedenen Kanäle, wie z.B. Internet, Mobile und Filialen, angeboten. Dies wird über die Service-Strategie zur Kanalbündelung unterstützt. Basis-Services stellen Teilinhalte für komposite (zusammengesetzte) Services zur Verfügung, die diese Teilinhalte entsprechend den Anforderungen des Vertriebskanals bündeln. Die Kanalbündelung kümmert sich also im Wesentlichen um die Ausgabe-Streams und nicht um Daten oder Anwendungsfunktionen.

- In der **fachlichen Domäne Produktbündelung** werden die Produkte für den Multikanalvertrieb bereitgestellt. Hier gibt es eine Service-Strategie, die verschiedenen Dienstleistungsbausteine redundanzfrei als Services zur Verfügung zu stellen und entsprechend der Anforderungen aus den Kanälen zu den geforderten Dienstleistungen zusammenzustellen. Die Services innerhalb der Produktbündelung stellen Anwendungsfunktionen und Produktdaten bereit, die sie aus den Abwicklungssystemen beziehen.
- Die **fachliche Domäne Abwicklung** besteht aus monolithischen Systemen für die verschiedenen Produkte und die Partnerverwaltung. Die Anwendungsfunktionen werden über Wrapper (Applikations-Services) für die Produkt-Services zur Verfügung gestellt.

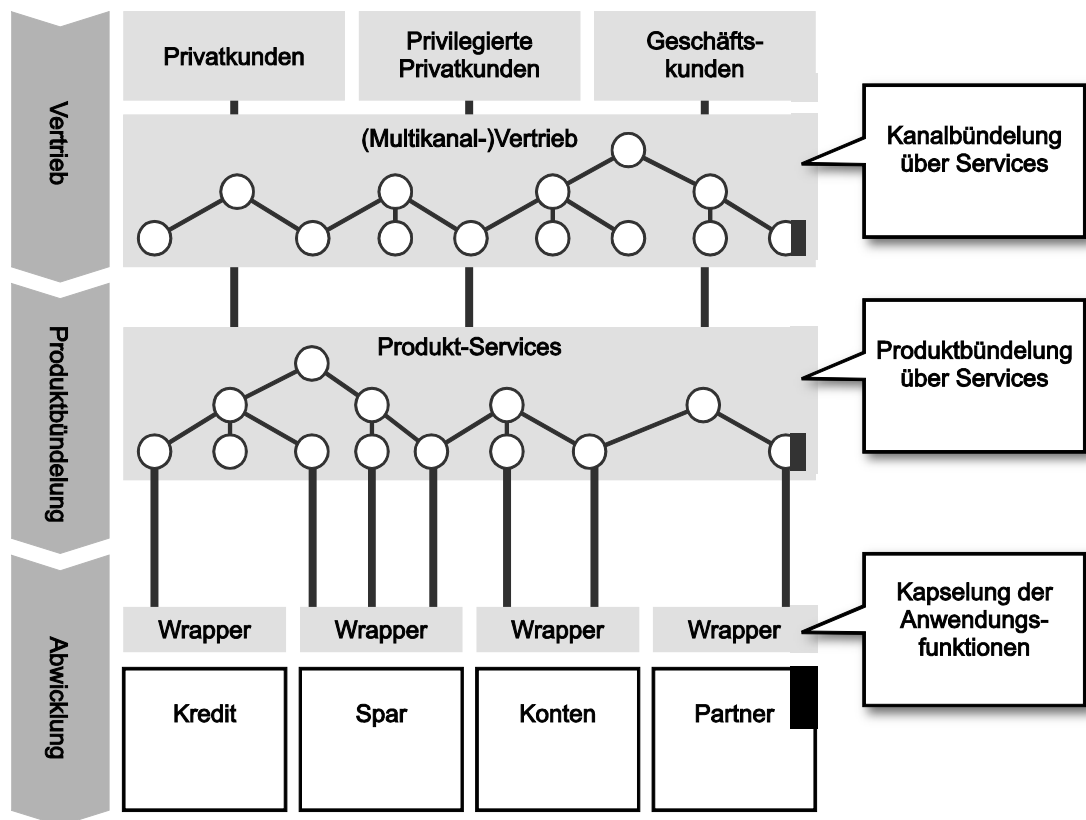


Bild 11.4 Fachliches Domänenmodell mit Angabe der Servicestrategie

Nachdem die Agilitätsanforderungen bestimmt wurden, können die Business-Services festgelegt werden. Dies hört sich in der Theorie sehr einfach an. In der Praxis stellt sich im Kontext der Business-Analyse insbesondere die Frage: "Wie findet man die richtigen Business-Services?" Hierzu erhalten Sie hier konkrete Hilfestellungen. Sie finden neben Begriffsdefinitionen eine Schritt-für-Schritt-Anleitung für die Ableitung von Business-Services mit praktischen Beispielen und Hilfestellungen für das Tailoring für Ihren Bedarf.

Die Ableitung von Business-Services wird in den folgenden Abschnitten detailliert beschrieben. Zuerst schauen wir uns aber Services noch etwas genauer an.

11.2 Servicekategorien

Es gibt verschiedene Arten von Services. In Bild 11.5 finden Sie in Anlehnung an [Kra04] verschiedene Arten von Services als Zwischenschicht zwischen Geschäftsprozessen und Informationssystemen. Die Geschäftsprozesse sind mit ihren Aktivitäten und Teilabläufen dargestellt. Bei den Services unterscheiden wir vier Kategorien von Services in Anlehnung an [Kra04]:

- **Orchestration Services**

Orchestration Services decken komplette Abläufe ab und steuern den Prozessablauf eines (Teil-)Geschäftsprozesses. Die Abläufe sind häufig langandauernd und beinhalten Benutzerinteraktionen. Orchestration Services nutzen Services der anderen Kategorien. Die Umsetzung erfolgt häufig über ein BPMS¹.

- **Aufgabenorientierte Services**

Die Services dieser Kategorie sind anwendungsfallorientiert. Es sind in der Regel Aktivitäten aus Sicht der Anwender. Sie realisieren Use-Cases wie z.B. „Markieren eines Eintrags in einer Kreditliste zur Weiterverarbeitung“ oder Aktivitäten eines Geschäftsprozesses. Änderungen im Use-Case beziehungsweise Geschäftsprozess führen direkt zu Änderungen an den Services. Aufgabenorientierte Services können andere Aufgabenorientierte Services und Services der in Bild 11.5 darunterliegenden Serviceschichten nutzen.

- **Entitätenzentrierte Services**

Entitätenzentrierte Services bilden die Geschäftslogik von Kerngeschäftsobjekten ab. Diese Services sind sehr datenzentrisch und bieten Verwaltungsfunktionen für das Erzeugen, Lesen, Verändern und Löschen der Geschäftsobjekte. Diese Operationen werden häufig mit "CRUD" bezeichnet (C – Create, R – Read, U – Update, D – Delete). Darüber hinaus werden Geschäftsregeln für Geschäftsobjekte unterstützt. Ein Beispiel für eine Geschäftsregel des Kerngeschäftsobjekts „Kreditkarten“ ist die „Überprüfung des Kreditlimits“. Entitätenzentrierte Services können andere entitätenzentrierte Services und Applikations-Services nutzen.

- **Applikations-Services**

Applikations-Services stellen Interfaces zu Anwendungsfunktionen von bestehenden Informationssystemen zur Nutzung für andere Services bereit. Sie kapseln die Anwendungsfunktionalität von Legacy-Anwendungen und stellen möglichst generische und wiederverwendbare Services bereit. Bei Legacy-Anwendungen sind häufig Adapter notwendig (siehe [Sta09]). Applikations-

¹ Business Process Management System oder Workflow-Engine

Services sind technologie- oder produktabhängig und zugeschnitten auf die bestehenden Informationssysteme, deren Anwendungsfunktionen sie zur Nutzung bereitstellen.

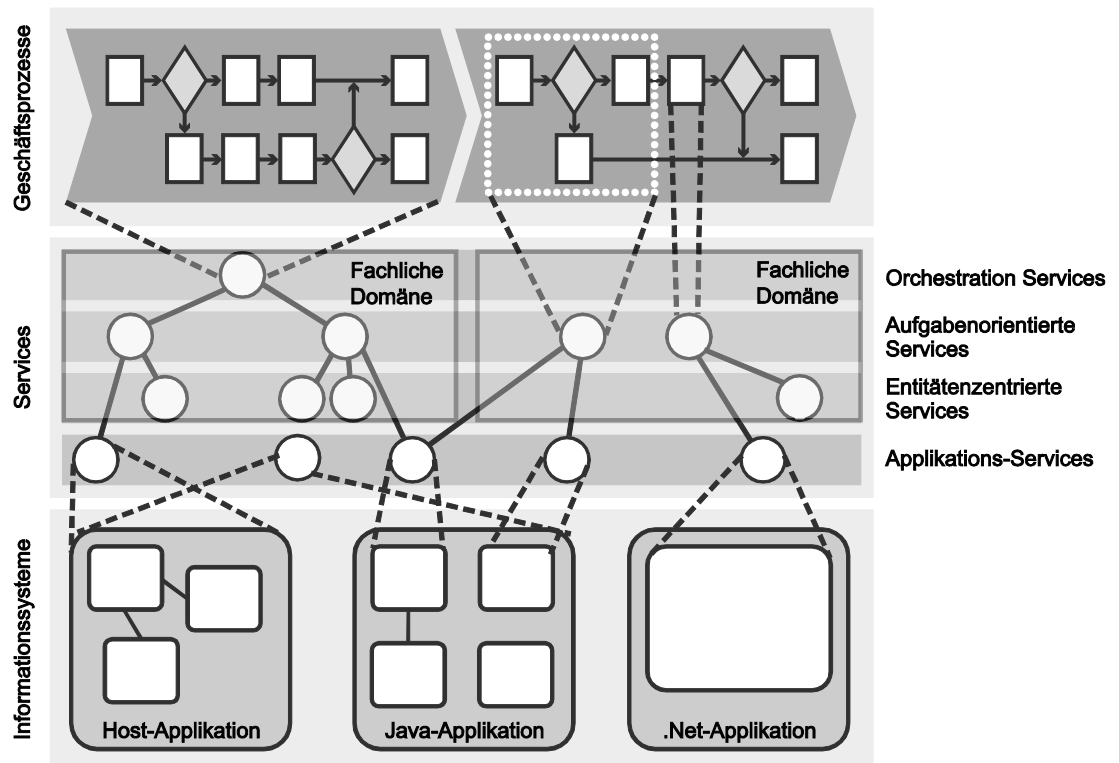


Bild 11.5 Arten von Services

Bis auf Applikations-Services sind alle Servicekategorien fachlich orientiert und damit Business-Services, wenn sie für die Anwender sichtbar sind. Häufig folgen die Business-Service-Einheiten insbesondere bei aufgabenorientierten Services der Segmentierung der Prozesse (Prozessorientierung), wie in Bild 11.6 verdeutlicht wird.

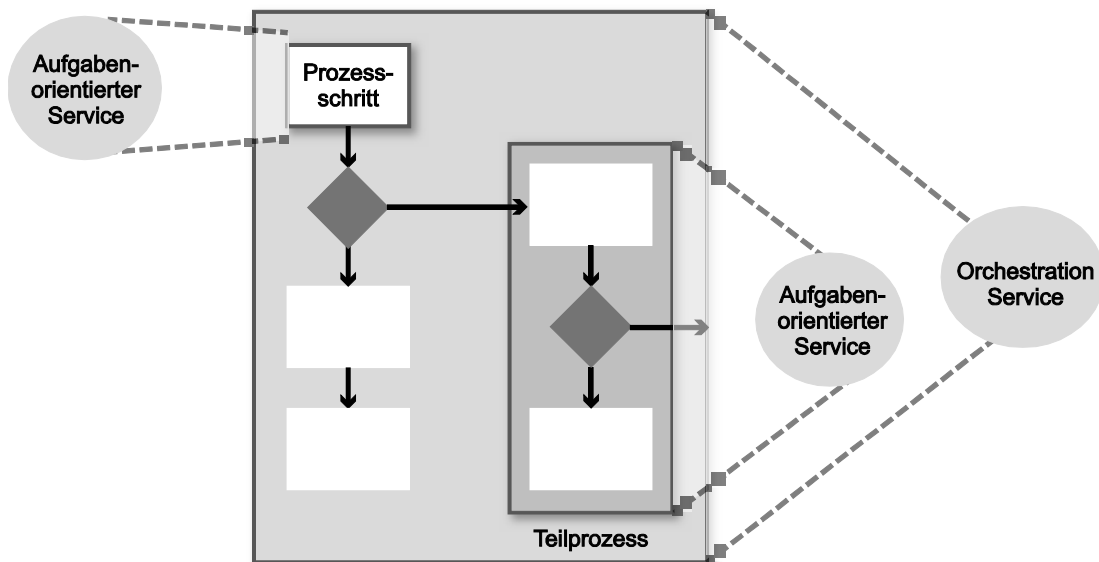


Bild 11.6 Prozessorientierte Business-Services

Business-Services sind in der Regel in möglichst unabhängige fachliche Domänen ("funktionale Cluster") strukturiert. Dies vereinfacht das Management, da in der Regel für fachliche Domänen klare Verantwortlichkeiten bestehen (siehe [Han11]). Ein Beispiel für fachliche Domänen finden Sie im funktionalen Referenzmodell in Bild 11.7.

Applikations-Services sind letztlich technische Serviceimplementierungen oder Interfaces für den Zugriff auf die Informationssysteme wie z.B. Standardsoftware oder bestehende Individualsoftware. Die Applikations-Services weisen bei gewachsenen Landschaften in der Regel unterschiedliche Granularitäten auf. Informationssysteme auf Basis einer SOA-Referenzarchitektur sind bereits in Komponenten strukturiert und bieten als Services entitätenzentrierte oder aufgabenorientierte Services oder Orchestration Services.

Definition Service-orientierte Architektur (SOA)

Eine Serviceorientierte Architektur (SOA) ist ein IT-Architekturstil, in dem Services (Dienste) die zentrale Rolle spielen. Die IT-Landschaft wird in modulare Services strukturiert. Jeder Service trägt direkt oder indirekt zur Wertschöpfung bei. Er kann einfach und flexibel kombiniert und regelbasiert gesteuert werden.

Definition SOA-Referenzarchitektur

Eine SOA-Referenzarchitektur macht unternehmensspezifische Vorgaben für die technische Umsetzung der Serviceorientierung. Technologie-, Softwarearchitektur- und Infrastrukturaspekte für Entwicklung, Betrieb und Governance der involvierten Einzelsysteme und deren Zusammenspiel (End-to-end) werden festgelegt.

Bei Legacy- und auch Standardsoftware verhält sich dies häufig anders. Wenn diese monolithisch aufgebaut sind, lassen sich deren Anwendungsfunktionen nicht direkt von außen über Schnittstellen ansprechen. Aus Machbarkeits- oder Aufwandsgründen werden diese Systeme häufig nicht zerlegt (komponentisiert), sondern nur "gewrappt" oder einzelne isoliertere Komponenten werden neu implementiert. Die Anwendungsfunktionen sind also nicht herausgelöst, nicht von Business-Services so ohne weiteres separat nutzbar und damit auch nicht orchestrierbar.

Zudem verfolgen die verschiedenen monolithischen Anwendungen (auch „Silos“ genannt) häufig unterschiedliche SOA-Ansätze mit unterschiedlichen technischen Infrastrukturen und Implementierungen, die häufig nicht zusammenarbeiten können. In diesen Fällen ist es kaum möglich, "sauber" geschnittene Business-Services bereitzustellen. Der Abgleich mit den Gegebenheiten ist daher ein wichtiger Aspekt beim Entwurf von Business-Services (siehe Abschnitt 11.4).

Tabelle 11.1 Gegenüberstellung der verschiedenen Servicekategorien

	Orchestration Service	Aufgaben-orientierter Service	Entitäten-zentrierter Service	Applikations-Service
Charakteristika	Prozesslogik in Service gekapselt Häufig mit BPMS umgesetzt	Anwendungsfall-spezifische "Unit of Work" Häufig keine Orchestration Layer	Anwendungsfall- und prozessübergreifende Kapselung von entitäten-bezogener Businesslogik	Technologie und produktabhängige Services
Vorteile	Flexible Änderbarkeit	Niedriger Analyse-aufwand aufgrund vorliegender Anwendungsfälle	Hohe Wiederverwendung	Technische Entkopplung

Nachteile	Komplexe Middleware oder Produkte	Geringe Wiederverwendung	Hoher Analyseaufwand wegen übergreifendem Ansatz	Hoher Aufwand bei der QoS ² -Sicherstellung
------------------	-----------------------------------	--------------------------	--	--

Wichtig

Sie müssen nicht alle Servicearten verwenden. Sie können z.B. auf Orchestration Services verzichten und die Ablauflogik direkt in aufgabenorientierten Services abbilden. Ein weiteres Beispiel sind Mischformen, die aufgabenorientierte, entitätenzentrierte und Applikations-Services kombinieren. Dies ist eine typische Ausprägung in einem frühen Stadium der SOA-Adaption. Hier liegt der Schwerpunkt häufig auf der Bereitstellung von Infrastruktur- und Integrationsdiensten mit einem Pilotcharakter. Auf der Basis der dadurch geschaffenen SOA-Infrastrukturdienste können Projektinitiativen folgen, in denen höherwertige Business-Services erstellt werden.

Die Unterscheidung zwischen aufgabenorientierten und entitätenzentrierten Services ist wichtig, da sich deren Charakter und die Art ihrer Herleitung unterscheiden.

Die Ableitung von Business-Services wird in den folgenden Abschnitten detailliert beschrieben.

11.3 Top-down- versus Bottom-up-Ableitung

Bei der Identifikation von Business-Services kann sowohl top-down als auch bottom-up vorgegangen werden. Der Top-down-Ansatz wird häufig als "strategisch" und der Bottom-up-Ansatz als "opportunistisch" bezeichnet. Im Folgenden werden die beiden Ansätze getrennt grob vorgestellt und im Anschluss wird eine bewährte Mischform vorgestellt. Details zum vorgeschlagenen Vorgehen finden Sie in Abschnitt 11.4.

Top-down-Ansatz

Beim Top-down-Ansatz werden die übergreifenden Strukturen und strategischen Vorgaben (Vision, Strategie, Soll-Bebauung, Blueprint, Roadmap) als Grundlage für die SOA-Governance gesetzt. Von besonderer Bedeutung ist dabei der im Enterprise Architecture Management entwickelte fachliche und technische Bauplan der SOA-Landschaft (siehe [Han09]).

² QoS steht für Quality of Service.

Der **fachliche Bauplan** beschreibt die Geschäftsarchitektur des Unternehmens als Rahmenvorgabe für die Weiterentwicklung des Geschäfts und der IT-Landschaft. Im Kontext der Ableitung von Business-Services wird er in der Regel über ein funktionales Referenzmodell (siehe Bild 11.7) gegebenenfalls detailliert durch ein fachliches Komponentenmodell (siehe Bild 11.8) oder/und Swimlane-Diagramme (siehe Download-Anhang 13) und Prozessablaufdiagramme (siehe Bild 3.8) sowie fachliche Klassenmodelle (siehe [HGG12]) beschrieben. Anstelle eines funktionalen Referenzmodells wird in der Business-Analyse häufig eine Prozesslandkarte verwendet (siehe Abschnitt 6.5 im Buch), da sie etwas grobgranularer und damit "einfacher" zu erstellen und abzustimmen ist. Für die Ableitung von Business-Services eignet sich jedoch ein funktionales Referenzmodell mehr, da es bereits fachliche Funktionen und damit Kandidaten für Business-Services beinhaltet.

In Bild 11.7 finden Sie einen Ausschnitt des funktionalen Referenzmodells des *EasyHouse*-Beispiels aus [HGG12]. Gruppiert nach fachlichen Domänen werden die für das Geschäft erforderlichen fachlichen Funktionen benannt. Die fachlichen Funktionen beschreiben in einer High-level-Sicht die erforderlichen Fähigkeiten des Unternehmens, die durch Business-Services unterstützt werden.

Hinweis

Fachliche Funktionen können in Teilfunktionen aufgeteilt werden. In Bild 11.7 ist eine bereits sehr detaillierte Sicht dargestellt, da lediglich ein Ausschnitt des Unternehmens adressiert wird. Im funktionalen Referenzmodell für das Gesamtunternehmen werden in der Regel die fachlichen Funktionen auf einer gröberen Ebene dargestellt.

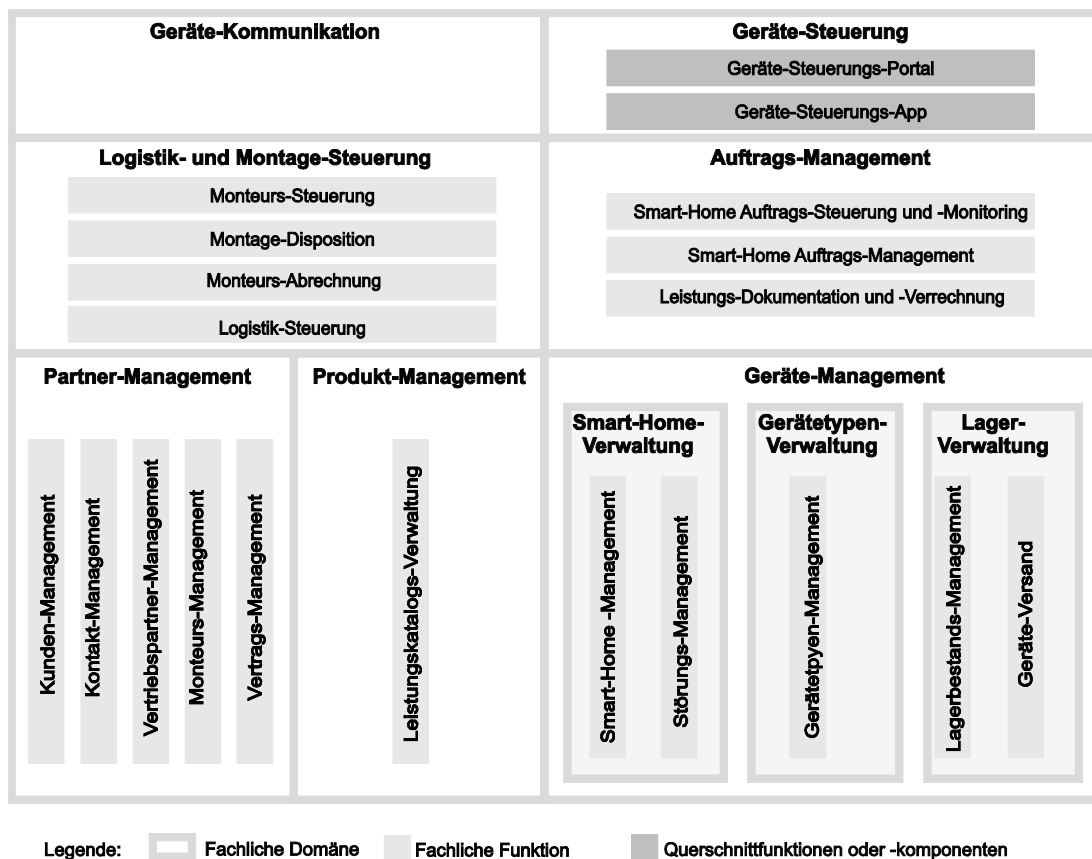


Bild 11.7 Beispiel funktionales Referenzmodell aus dem Kontext EasyHouse (siehe [HGG12])

In Bild 11.8 finden Sie ein fachliches Komponentenmodell (siehe [HGG12]). Im fachlichen Komponentenmodell werden die fachlichen Domänen und fachlichen Funktionen des funktionalen Referenzmodells in der Regel weiter detailliert und Abhängigkeiten zwischen den fachlichen Komponenten dargestellt. Die Detaillierung erfolgt in der Regel im Rahmen des Demand Managements oder aber von Projekten. Das (oder mehrere) fachliche Komponentenmodell kann bottom-up genutzt werden, um ein funktionales Referenzmodell abzuleiten beziehungsweise das Bestehende zu aktualisieren oder zu vervollständigen.

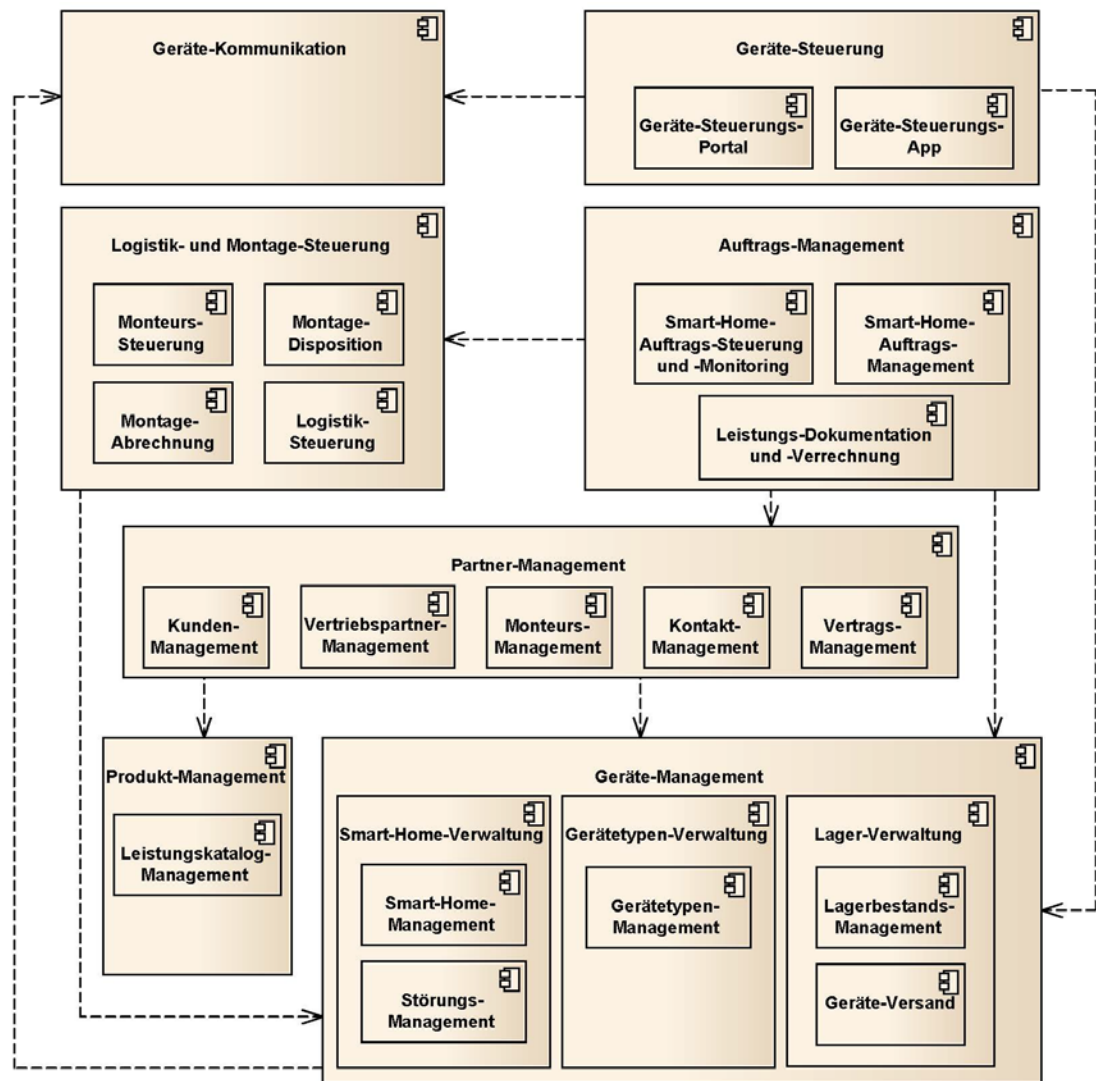


Bild 11.8 Beispiel fachliches Komponentenmodell (siehe [HGG12])

Für die Ableitung von Business-Services sind insbesondere feingranulare fachliche Funktionen von Bedeutung, die im funktionalen Referenzmodell auf detaillierter Ebene oder in fachlichen Komponentenmodellen visualisiert werden. Durch die fachlichen (Teil-)Funktionen werden die für das aktuelle und zukünftige Geschäft erforderlichen Business-Services top-down grob vorgegeben.

Der **technische Bauplan**, z.B. über eine SOA-Referenzarchitektur, gibt hingegen vor, wie die Business-Services umgesetzt werden sollen. Durch die übergeordnete Koordination auf Basis dieser Baupläne wird verhindert, dass in unterschiedlichen Projekten funktional sich überschneidende Services entstehen, die gegebenenfalls unterschiedlich technisch realisiert und damit nicht interoperabel sind.

Häufig wird eine SOA-Referenzarchitektur als technischer Bauplan eingesetzt (siehe Abschnitt 3.2 im Buch). Für die Top-down-Ableitung einer SOA-Referenzarchitektur kann ein allgemeines Raster vorgegeben werden, in dem alle relevanten Aspekte aufgeführt sind. Dieses Raster ist dann unternehmensspezifisch auszufüllen. Wesentliche Aspekte sind dabei:

- **Prinzipien und Leitlinien** für die Serviceorientierung, wie z.B. lose Kopplung
- **Technologische Standards** für Informationssysteme, Protokolle, Middleware und Infrastrukturelemente sowie Methoden und Werkzeuge für Modellierung, Softwareentwicklung, Test, Integration, Betrieb und Management
- **Softwarearchitektur-Vorgaben** für alle unternehmensspezifisch festgelegten Kategorien von Individualsoftware (z.B. JEE) und Standardsoftware (z.B. SAP).

Unter anderem muss für die verschiedenen Kategorien von Standardsoftware festgelegt werden, wie Services für die Nutzung durch einen ESB³ oder ein BPMS⁴ bereitgestellt werden (z.B. als WebService). Ein weiteres Beispiel sind Vorgaben für die Umsetzung von Lese- und Schreibzugriffen in verteilten Systemen, um Applikations-Services zu implementieren.

Wenn Erweiterungen an der Standardsoftware erforderlich werden können, um notwendige Capabilities zu implementieren, müssen auch **Entwicklungsvorgaben** für die Umsetzung von Services gegeben werden. So kann z.B. als Rahmenvorgabe gesetzt werden, dass Erweiterungen nur außerhalb der Standardsoftware entlang definierter Aussprungpunkte vorgenommen werden.

- **Integrationsarchitektur** mit der Festlegung der Integrationstechnologien (z.B. ESB und BPMS) und deren Anwendung sowie der Einbindung der verschiedenen Kategorien von Individual- und Standardsoftware, wie z.B. „Wann wird SOAP-over-HTTP und wann SAP RFC für das Ansprechen von SAP-Funktionen verwendet?“

³ Enterprise Service Bus

⁴ Business Process Management System

- **Nichtfunktionale Anforderungen**, insbesondere unter Berücksichtigung von Aspekten wie End-to-end Performance, Skalierbarkeit, Logging, Monitoring und Sicherheit

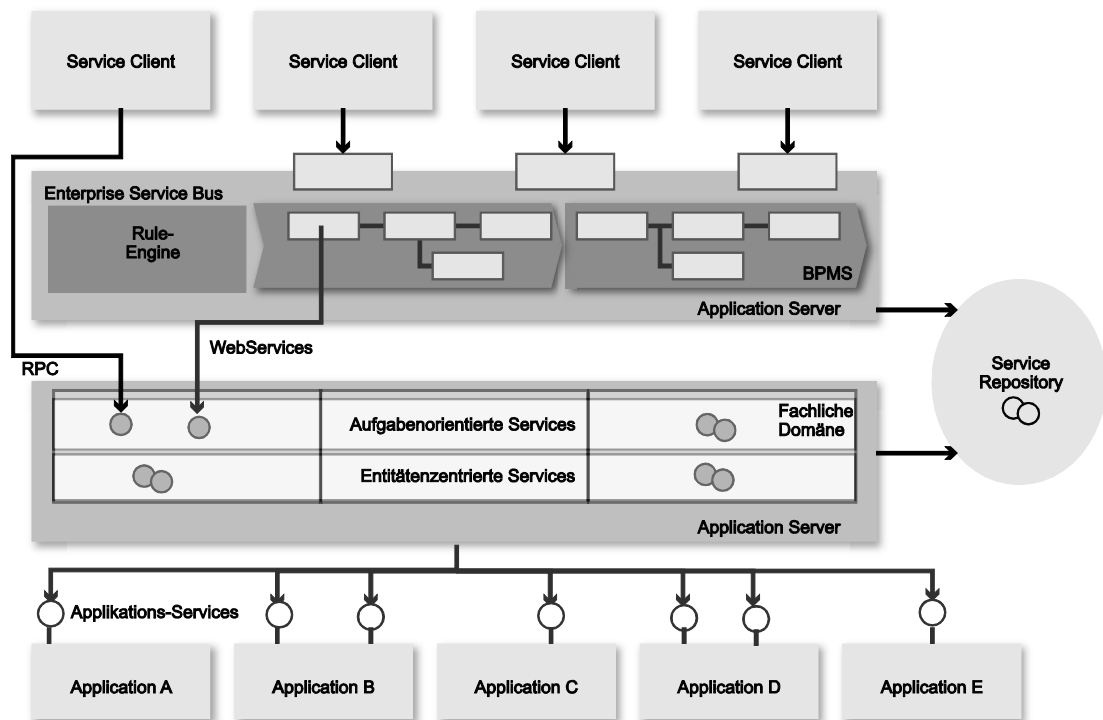


Bild 11.9 Beispiel einer SOA-Referenzarchitektur

Durch die unternehmensspezifische Ausprägung der Aspekte wird die technische Umsetzung von Services und deren Integration und Orchestrierung ermöglicht. In Bild 11.9 finden Sie ein Beispiel einer SOA-Referenzarchitektur, die sowohl aufgabenorientierte als auch entitätenzentrierte und Applikations-Services nutzt. Die Business-Services sind in einem Service-Repository hinterlegt. Als Integrationsinfrastruktur dient ein Enterprise Service Bus (ESB) mit einer integrierten Workflow- und Rule-Engine. Diese greift mittels ESB-spezifischer Adapter auf die Business-Services zu. Zudem ist ein Direktzugriff von den verschiedenen Clients auf die Business-Services in Umgehung der Integrationsinfrastruktur in einzelnen zu genehmigenden Ausnahmefällen möglich. Die Business-Services sind in fachliche Domänen gruppiert. Die Applikations-Services kapseln die bestehende IT-Landschaft und liefern die Anwendungsfunktionen, die von den Business-

Services benötigt werden. Hierzu müssen gegebenenfalls mehrere Altsysteme angesprochen werden.

Die unternehmensspezifische Festlegung der SOA-Referenzarchitektur sollte schrittweise erfolgen (siehe [Eng08]). Häufig wird mit der Festlegung von Technologien und einer groben Integrationsarchitektur begonnen. Deren Einhaltung muss über entsprechende Governance-Strukturen und -Prozesse sichergestellt werden (siehe [Kra04] und [Han10]).

Wichtig

Der Top-down-Ansatz liefert den fachlichen und technischen Bauplan für die SOA-Landschaft. Der fachliche Bauplan, insbesondere das funktionale Referenzmodell, ist ein Ziel-Bild, das über Projekte und Wartungsmaßnahmen schrittweise umgesetzt werden muss. Der technische Bauplan, z.B. eine SOA-Referenzarchitektur, setzt Rahmenvorgaben für die Umsetzung und damit auch für die Ableitung von Business-Services.

Bottom-up-Ansatz

Der Bottom-up-Ansatz ist in der Regel projektgetrieben. Ausgehend von den Projektanforderungen werden die Business-Services und Applikations-Services entsprechend der technischen Vorgaben, wie z.B. eine SOA-Referenzarchitektur, erstellt, zugekauft oder eine bestehende angepasst und so schrittweise eine komponentisierte flexible IT-Landschaft geschaffen. Das Ziel-Bild steht jedoch bei einem reinrassigen Bottom-up-Ansatz zu Beginn noch nicht fest.

Anwendungsfunktionen von bestehenden IT-Systemen werden im Rahmen konkreter Umsetzungsprojekte gekapselt und als Applikations-Services bereitgestellt. Die Business-Services können die Applikations-Services zum Ansprechen der Anwendungsfunktionen nutzen. So lautet auf jeden Fall die Theorie. In der Praxis muss man aber die zum Teil historisch gewachsene bestehende IT-Landschaft berücksichtigen, da die Anwendungsfunktionen erst herausgelöst und als Applikations-Services in die Landschaft integriert werden müssen.

Vorhandene IT-Systeme sind häufig monolithisch. Um den Ziel-Zustand zu erreichen, haben Sie dann zwei Möglichkeiten. Sie können entweder Systeme durch neue selbst entwickelte oder zugekaufte modulare Systeme ablösen oder aber vorhandene Systeme in Komponenten zerlegen. Die Umsetzung erstreckt sich bei komplexen Systemen in beiden Fällen über viele Jahre. Der Aufwand und die Zeit für die Umsetzung der "idealen" Business-Services sind häufig nicht vertretbar und durchsetzbar. Aufwand und Nutzen des für "ideale" Business-Services erforderlichen Umbaus müssen sorgfältig gegeneinander abgewogen werden.

Besonders schwierig ist dies bei historisch gewachsenen Legacy-Anwendungen. Schon kleine Änderungen können in dem komplexen Gesamtsystem unüberschaubare und verheerende Auswirkungen haben. Diese wartungsintensiven und inflexiblen Anwendungen werden entsprechend dem Grundsatz "never touch a running system" über lange Zeit nur am Leben gehalten und nur verändert, wenn es zwingend notwendig erscheint. Erweiterungen werden häufig nicht sauber integriert, sondern als "Rucksack" implementiert. Dies führt zu einer unnötigen und unüberschaubaren Komplexität infolge unnötiger Schnittstellen, Redundanzen und einer technologischen Vielfalt. Eine schrittweise Ablösung oder Umstrukturierung ist zwar wie eine "Operation am offenen Herzen" mit großen Risiken verbunden, aber häufig ist das die einzige Chance, auf Dauer dem Problem Herr zu werden. Es erfordert aber einen langen Atem. Kurzfristige Erfolge sind selten zu erzielen. Insofern ist die Abwägung sicherlich nicht einfach.

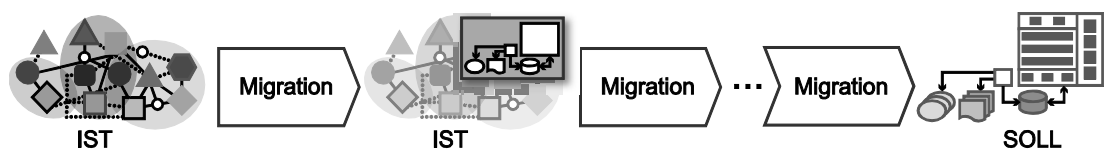


Bild 11.10 Schrittweise Migration in Richtung Ziel-Zustand

In Bild 11.10 ist die schrittweise Migration vom Ist-Zustand bis zum Ziel-Zustand symbolisch dargestellt. Dies macht die Herausforderungen gut sichtbar.

Wichtig

Die Komponentisierung der IT-Landschaft und die Einführung einer Integrationsarchitektur sind wichtige Bestandteile der IT-Konsolidierung. Eine solche IT-Konsolidierung können Sie nur mit großem Durchhaltewillen über viele Jahre hinweg im Rahmen von Projekten und Wartungsmaßnahmen umsetzen. Dies verursacht sicherlich bei den ersten Projekten Mehrkosten, um die fachliche und technische Architektur und Infrastruktur zu schaffen. Nur so erreichen Sie aber die gewünschte Flexibilität.

Nutzen Sie möglichst jedes Projekt und auch Wartungsmaßnahmen, um Ihre IT schrittweise vorzubereiten (Komponentisierung und Integrationsarchitektur). Die Argumentation gegenüber der Unternehmensführung ist sicherlich nicht einfach. Wenn die Fachbereiche bislang unzufrieden mit Dauer und Kosten der Umsetzung von Geschäftsanforderungen sind, können Sie diese "Pains" für Ihre Argumentation nutzen.

Sowohl für die Identifikation der "idealen" Business-Services als auch für den Abgleich mit den bestehenden Systemen und für die Migrationsplanung liefert das Business-Analyse-Instrumentarium wertvolle Unterstützung. Die statischen und ablauforientierten Modelle, die bottom-up entstehen, sind feingranularer als die Top-down-Modelle. Use-Cases, Prozessablaufdiagramme und fachliche Klassenmodelle sind neben verfeinerten fachlichen Komponentenmodellen die wesentlichen Bottom-up-Modelle (siehe hierzu Abschnitt 11.4).

In der Tabelle 7.2 finden Sie die wesentlichen Vor- und Nachteile des Bottom-up- und Top-down-Ansatzes gegenübergestellt.

Tabelle 7.2 Gegenüberstellung des Bottom-up- und des Top-down-Ansatzes

	Bottom-up-Ansatz	Top-down-Ansatz
Charakteristika	„Projektgetrieben“ Entwicklung eines Service wird durch Projektanforderungen initiiert	Unternehmensübergreifend Services werden im Rahmen des Enterprise Architecture Management definiert
Vorteile	Verwendbarkeit durch reale Anforderungen sichergestellt Erfahrungen werden „im Kleinen“ gesammelt	Business-IT-Alignment Design für unternehmensübergreifenden Einsatz Risiko redundanter Services reduziert Nutzen von Investitionen leichter transparent zu machen
Herausforderungen	Berücksichtigung von strategischen und unternehmensübergreifenden Aspekten Projektgranularität für Enterprise Level passend? "Siloed" SOA führt nicht zwangsläufig zu einer unternehmensübergreifenden Lösung	Unternehmensübergreifendes SOA erfordert eine hohe Anfangsinvestition Die Komplexität des übergreifenden Ansatzes muss beherrscht und reale Services müssen bereitgestellt werden

Empfehlung: Eine hybride und agile Vorgehensweise

Wir empfehlen eine hybride und gleichzeitig agile Vorgehensweise, um die Vorteile von beiden Ansätzen zu vereinen. Übergeordnete Vorgaben und Strukturen für Business-Services sollten im Rahmen des Enterprise Architecture Managements gesetzt und projektgetrieben, d.h. bottom-up, mit Leben gefüllt bzw. justiert werden.

Insbesondere durch ein funktionales Referenzmodell und fachliche Komponentenmodelle geben Sie Top-down-Kandidaten für Business-Services vor. Die top-down vorgeschlagenen Kandidaten dienen als Input für das Service-Design in Projekten. Konkret nutzbare Business-Services entstehen erst im Rahmen von Projekten und Wartungsmaßnahmen (bottom-up). Die bottom-up bereitgestellten Services können sich von den Top-down-Servicekandidaten durchaus unterscheiden, da die bereitgestellten Services die realen Projektanforderungen umsetzen. Sie ersetzen oder ergänzen die Top-down-Servicekandidaten zumindest teilweise.

Das funktionale Referenzmodell ist der fachliche Bauplan und Ihr spezifisches SOA-Referenzmodell ist der technische Bauplan für Ihre serviceorientierte Landschaft. Durch eine geeignete SOA-Governance erwecken Sie diese schrittweise zum Leben. Siehe hierzu [Kra04] und [Eng08].

Wichtig

Durch einen regelmäßigen Abgleich der Top-down-Vorgaben mit den Erkenntnissen aus den Projekten wird der fachliche und technische Bauplan schrittweise justiert und damit immer „realitätsnaher“. So lässt sich die Angemessenheit der Business-Services für die Umsetzung der Geschäftsanforderungen sicherstellen.

Das Zusammenwirken von den Top-down- und Bottom-up-Vorgehen wird in Bild 11.11 dargestellt.

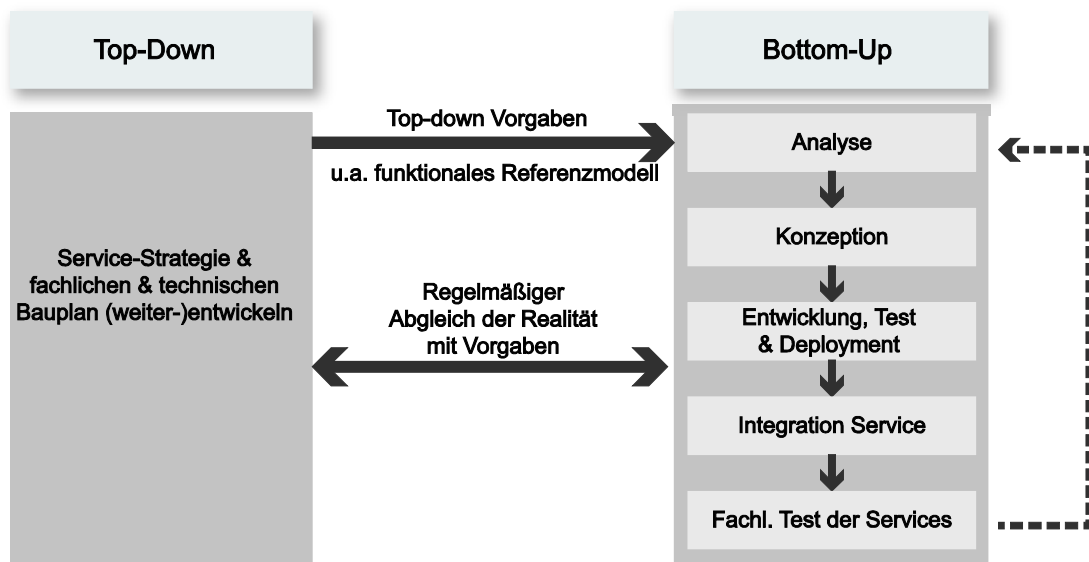


Bild 11.11 Agile Entwicklungsstrategie für Services

Top-down werden in der Strategieentwicklung und insbesondere im Enterprise Architecture Management der fachliche und technische Bauplan und Service-Strategien gestaltet und für die Bottom-up-Umsetzung im Rahmen von Projekten und Wartungsmaßnahmen vorgegeben.

In Projekten findet für die Liefereinheiten (Inkremente) eine Analyse, Konzeption, Entwicklung, Integration der Services statt (siehe Abschnitt 11.4). Spätestens dann muss erneut ein Abgleich mit dem dann bestehenden fachlichen und technischen Bauplan erfolgen. Die Abweichungen fließen als Input sowohl in die Top-down- als auch in die Bottom-up-Weiterentwicklung ein. Idealerweise sollten Unstimmigkeiten aber bereits dann, wenn sie (z.B. bei der Analyse oder der Konzeption) festgestellt werden, zurückgespielt und eine gemeinsame Lösung gesucht werden.

Startpunkt ist die Analyse der Geschäftsanforderungen zur Identifikation von Servicekandidaten. In der Konzeption werden die zu implementierenden Services und deren Umsetzungseinheiten festgelegt und die Services gegebenenfalls in Inkrementen entwickelt, integriert und getestet. Anschließend startet der Kreislauf neu. Die Gesamtsituation wird wieder analysiert und das nächste Inkrement oder Service geplant.

Die eigentliche Entwicklung wird in Bild 11.12 weiter detailliert. Hier wird zwischen der Entwicklung des Services und des für die Nutzung des Services häufig notwendigen Applikations-Frontend unterschieden. Das Applikations-Frontend ist in der Regel dann notwendig, wenn der Service oder aber die Integrationsinfrastruktur (z.B. Worklist einer Workflow-Engine) selber keine Benutzerschnittstelle hat. Für die Entwicklung von Service und Applikations-Frontend sind eine Reihe von Aktivitäten notwendig. Dies fängt mit der Implementierung eines Test-Clients beziehungsweise Test-Services an, mit dem die Funktionalität als Ganzes getestet werden kann. Dann folgen die eigentliche Entwicklung, der Entwicklungstest und das Deployment der Softwarepakete für den Service und das Applikations-Frontend, die dann in die Gesamtumgebung integriert werden.

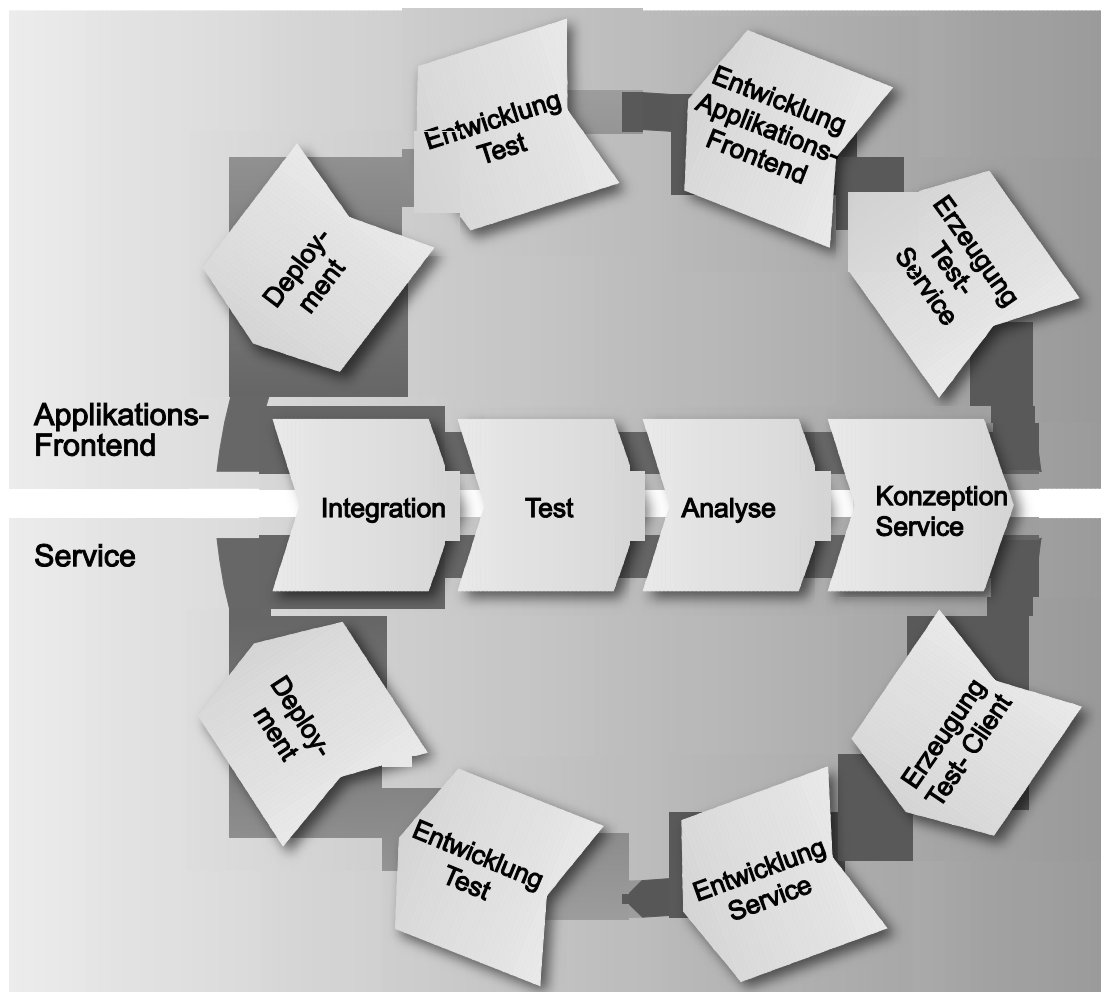


Bild 11.12 Vorgehen bei der Entwicklung von Services nach [Kra04]

In der Praxis gibt es, wie schon erläutert, auch bei der agilen Entwicklungsstrategie viele Herausforderungen. Hierzu zählen:

- **Transparenz über die Ist-Situation**
Die Transparenz über die Ist-Situation ist die Grundlage, um die IT-Landschaft schrittweise in Richtung zum Ziel-Zustand weiterentwickeln zu können.
- **Identifikation der für SOA relevanten Anwendungsbereiche** (siehe Abschnitt 3.2 im Buch)
Voraussetzung für die Identifikation von SOA-relevanten Anwendungsbereichen sind bekannte Agilitätsanforderungen, die mit der Unternehmensstrategie einhergehen.
- **Gestaltung der Service-Strategie**
Hierzu müssen die Agilitätsanforderungen verstanden und über ein Business-IT-Alignment (Verknüpfung zwischen Geschäftsarchitektur und Informationssystemlandschaft) in Verbindung zur IT-Landschaft gebracht werden.

Daneben gibt es auch hier folgende allgemeine zusätzliche Herausforderungen:

- **Gewährleistung eines zuverlässigen Geschäftsbetriebs auch während der Migration**
Der laufende Geschäftsbetrieb darf während der Migrationsphase nicht gestört werden. Hierzu müssen Risikovorkehrungen getroffen werden, wie z.B. eine Rückfallstrategie.
- **Architektur- und Management-Skills**
Die Gestaltung des Ziel-Zustands und der Roadmap zur Umsetzung erfordert hohe Architektur-Skills (siehe [Han11]). Die Umsetzung erfordert Nachhaltigkeit und Durchsetzungsvermögen.

Hilfestellungen für diese Herausforderungen finden Sie in Kapitel 4 im Buch, [Kra04], [Eng08] und [Sla11].

Nun betrachten wir die Ableitung von Business-Services aus der Projektperspektive noch etwas näher an.

11.4 Schritt-für-Schritt-Anleitung für die Ableitung von Business-Services in Projekten

Im Folgenden finden Sie eine Schritt-für-Schritt-Anleitung für die Ableitung von Business-Services im Rahmen von Projekten. Hierbei gehen wir von der agilen Entwicklungsmethode aus, die im vorhergehenden Abschnitt beschrieben ist.

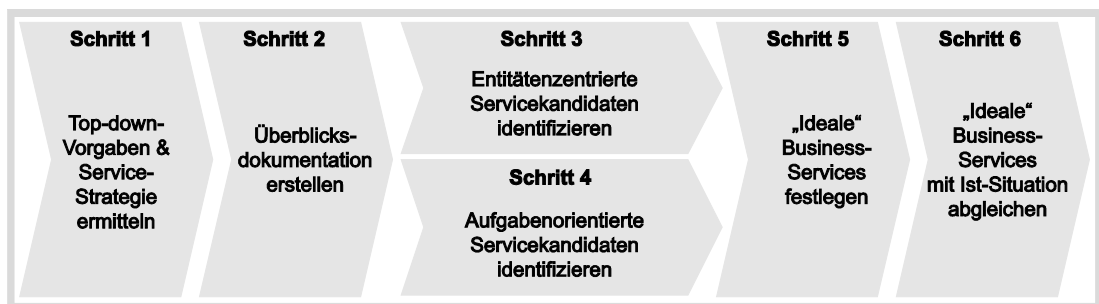


Bild 11.13 Vorgehen bei der Ableitung von Business-Services in Projekten

Die folgenden sechs Schritte zur Ableitung von Business-Services (siehe Bild 11.13) haben sich bewährt. Für Beispiele sei auf [HGG12] verwiesen.

Schritt 1: Ermitteln Sie Ihre Top-down-Vorgaben und Service-Strategie

Nutzen Sie, wenn vorhanden, das funktionale Referenzmodell und fachliche Komponentenmodelle sowie gegebenenfalls die Prozesslandkarte, Swimlane-Diagramme und Prozessablaufdiagramme (siehe Download-Anhang 13 und [HGG12]). Ermitteln Sie die für das Projekt relevanten Ausschnitte dieser Top-down-Modelle.

Liegen solche Modelle nicht vor, ermitteln Sie zumindest die relevanten Geschäftsprozesse auf Wertschöpfungskettenebene und/oder groben fachlichen Komponenten. Diese können gut für eine Strukturierung (fachliche Domänen) der Business-Services herangezogen und im Rahmen des Projekts schrittweise mit Le-

ben gefüllt werden. Damit haben Sie bereits die erste Iteration des funktionalen Referenzmodells.

Wichtig

Nutzen Sie nur vorhandene Top-down-Modelle. Sie können im Rahmen eines Projekts kein umfassendes funktionales Referenzmodell und keine SOA-Referenzarchitektur erstellen. Dies ist aufgrund des Abstimmungsaufwands zeitlich kaum möglich und würde vom Projekt nicht „getragen“ werden.

Sammeln Sie technische Vorgaben und erstellen Sie eine Architekturvision für die Umsetzung von Services, wie z.B. eine SOA-Referenzarchitektur. Ermitteln Sie die Randbedingungen für die Gestaltung von Business-Services, die daraus resultieren.

Wenn Sie noch über keine Service-Strategie verfügen, analysieren Sie Ihre Unternehmens- und IT-Strategie sowie Geschäftsanforderungen und identifizieren Sie Ihre Agilitätsanforderungen und weitere Randbedingungen. Interviewen Sie hierzu die Schlüsselpersonen im Business. Leiten Sie in Abstimmung mit den Schlüsselpersonen eine Service-Strategie ab.

Empfehlung

Wenn die Service-Strategie noch nicht vorliegt oder die Agilitätsanforderungen nicht klar kommuniziert sind, müssen Sie Annahmen darüber treffen, diese dokumentieren und z.B. in einer Folge von Workshops mit den jeweiligen Verantwortlichen abstimmen. Dies sind wichtige Randbedingungen für die Gestaltung Ihrer Business-Services.

Dokumentieren Sie alle Vorgaben, Randbedingungen und Annahmen für die Nachvollziehbarkeit von Entscheidungen.

Schritt 2: Erstellen Sie eine Überblicksdokumentation

Dokumentieren und analysieren Sie die für den Projektkontext relevanten Geschäftsanforderungen grobgranular. Nutzen Sie hierfür die Vorgehenshinweise, Hilfestellungen und Ergebnistypen aus den Download-Anhängen. Erfassen Sie insbesondere auch die Randbedingungen, Annahmen und nichtfunktionalen Anforderungen (siehe [Rup07]). Sofern die Modelle nicht bereits vorliegen, werden diese dabei erstellt.

Folgende iterativ auszuführenden Schritte sind hierfür notwendig:

- **Ermitteln Sie den Kontext.**

Nutzen Sie die für das Projekt relevanten Ausschnitte der Top-down-Modelle. Erweitern, verändern oder verfeinern Sie die Top-down-Modelle, falls notwendig. Stimmen Sie die Veränderungen an den Top-down-Modellen mit den Unternehmensarchitekten ab (siehe Abschnitt 11.3).

Erstellen oder verfeinern Sie insbesondere das funktionale Referenzmodell und das fachliche Komponentenmodell für den relevanten Ausschnitt, so dass die fachlichen Funktionen als Top-down-Ausgangspunkt für die Ableitung der Business-Services herangezogen werden können (erster Wurf). Sammeln Sie hierzu fachliche Funktionen und clustern Sie diese in fachliche Domänen. Als fachliche Domänen werden häufig Geschäftsprozesse auf Wertschöpfungskettenebene verwendet (siehe Abschnitt 3.5.3 im Buch).

Nutzen Sie hierbei, soweit schon vorhanden, Informationen aus der Use-Case- und Geschäftsprozessdokumentation. Hilfestellungen hierzu finden Sie in Download-Anhang 13.

Empfehlung

Im Rahmen der Analyseaktivität im Projekt werden die vorhandenen Top-down-Modelle erweitert, verfeinert oder verändert. Diese Änderungen sollten gesammelt vor der finalen Festlegung der Services mit den jeweiligen verantwortlichen Unternehmensarchitekten abgestimmt werden. Nur so können Sie sicherstellen, dass die strategischen übergreifenden Zielsetzungen auch berücksichtigt werden. Zudem ersparen Sie sich eine Menge Ärger und Korrekturaufwand, falls in Folge von Reviews oder Abnahmen die Service-Konzeption nochmals geändert werden muss.

- **Sammeln und dokumentieren Sie die Geschäftsobjekte.**

Identifizieren Sie die relevanten Geschäftsobjekte und erstellen Sie grobgranular fachliche Klassenmodelle. Dokumentieren Sie darüber hinaus Geschäftsregeln (Constraints, Einschränkungen und Berechnungen) zumindest in Prosa.

In den fachlichen Klassenmodellen werden die Geschäftsobjekte und ihre Beziehungen dargestellt. In der Dokumentation der Geschäftsregeln werden die Abhängigkeiten, Einschränkungen (Constraints) und Berechnungen beschrieben.

- **Ermitteln Sie die Use-Cases.**

Schaffen Sie, falls noch nicht vorhanden, einen möglichst vollständigen Überblick über die grobgranularen (System) Use-Cases (siehe [HGG12]) mit minimaler Bezeichnung und Beschreibung. Ordnen Sie die Use-Cases den Geschäftsprozessen, genauer genommen den Geschäftsprozessaktivitäten, zu.

Soweit mit geringem Zeitaufwand möglich, dokumentieren Sie zumindest grob wesentliche Use-Case-Schritte, aber vor allen Dingen Variationen, Sonderfälle und Fehlerfälle, wenn hier wichtige Funktionalitäten vermutet werden können. Nutzen Sie auch hier, soweit möglich, fachliche (Teil-)Funktionen aus dem funktionalen Referenzmodell.

Erweitern oder verfeinern Sie gegebenenfalls die Sammlung der fachlichen Funktionen und deren Teilfunktionen und dokumentieren Sie diese als mögliche noch nicht abgestimmte Funktionen im funktionalen Referenzmodell und/oder fachlichen Komponentenmodell.

Empfehlung

Für die Ableitung der aufgabenorientierten Services sind auf jeden Fall entweder die Use-Case-Sammlung oder aber gegebenenfalls Swimlane- oder Prozessablaufdiagramme erforderlich. Use-Cases sind jedoch vorzuziehen, da sie bereits eine Systemperspektive haben. Sie können also gegebenenfalls auf detaillierte Geschäftsprozessdokumentationen verzichten.

Wenn sowohl Use-Cases als auch Geschäftsprozesse dokumentiert werden, ordnen Sie Use-Cases den Aktivitäten der Geschäftsprozesse zu. Ein Use-Case beschreibt in sich abgegrenzte Funktionen eines IT-Systems, welche Akteure bei der Durchführung von einer oder mehreren Aktivitäten in Geschäftsprozessen unterstützen (siehe [HGG12]).

So haben Sie ein adäquates Mittel zur Konsistenzprüfung. Ein Geschäftsprozess stellt eine Zusammenfassung von fachlich zusammenhängenden Aktivitäten dar, die durchgeführt werden, um das Kundenbedürfnis End-to-end zu befriedigen. Die Use-Cases unterstützen bei der Ermittlung der Aktivitäten.

- **Detaillieren Sie die Geschäftsprozesse**

Ermitteln Sie Teilprozesse sowie wesentliche Aktivitäten und modellieren Sie diese in einem Swimlane-Diagramm (siehe Download-Anhang 13). Soweit für das Verständnis oder die Abstimmung notwendig, erstellen Sie auch detaillierte Prozessablauf-Diagramme (siehe Download-Anhang 13 und [HGG12]). Verwenden Sie als Aktivitäten, soweit möglich, fachliche Funktionen oder Teilfunktionen.

Wichtig

Bestehende Geschäftsprozessdokumentationen entsprechen häufig nicht den Modellierungsempfehlungen. Die Teilprozesse und Aktivitäten innerhalb eines Geschäfts-

prozesses liegen häufig in unterschiedlicher Granularität vor und verwenden keine abgestimmten fachlichen Funktionen. Analysieren Sie und berücksichtigen Sie dies bei der Ableitung von Services.

Erweitern oder verfeinern Sie gegebenenfalls die Sammlung der fachlichen Funktionen und deren Teilfunktionen und dokumentieren Sie diese als mögliche noch nicht abgestimmte Funktionen im funktionalen Referenzmodell und im fachlichen Komponentenmodell.

Wichtig

Beschränken Sie sich bei der Erstellung der Überblicksdokumentation auf das Wesentliche und Wichtige. Der Aufwand muss in einem angemessenen Verhältnis zum Nutzen stehen und vor allen Dingen in kurzer Zeit leistbar sein. Wichtig ist hier die richtige Granularität der Modelle. Dies ist bei den einzelnen Modellen oben ausgeführt.

Auf der Basis der Überblicksdokumentation kann die Ableitung von Servicekandidaten starten. Das Vorgehen für die Ableitung von aufgabenorientierten Services unterscheidet sich von dem für entitätenzentrierte Services. Daher beschreiben wir dies im Folgenden getrennt voneinander. Die Reihenfolge ist, wie in Bild 11.13 dargestellt, gleichgültig. Häufig wird mit entitätenzentrierten Servicekandidaten gestartet, da dies einfacher erscheint.

Schritt 3: Identifizieren Sie Kandidaten für entitätenzentrierte Services

Sammeln Sie durch Analyse und gegebenenfalls Verfeinerung der fachlichen Klassenmodelle und Geschäftsregeln mögliche Servicekandidaten. Folgende iterativ auszuführenden Schritte sind hierfür notwendig:

1. Identifizieren Sie die Kernentitäten.

Kernentitäten sind die Geschäftsobjekte von Bedeutung, die unabhängig von anderen existieren. Beispiele für Kernentitäten sind „Auftrag“ oder „Kunde“. Ermitteln Sie die Kernentitäten, die für den Projektkontext relevant sind, möglichst vollständig.

Analysieren Sie die vorliegenden Modelle und verfeinern Sie diese gegebenenfalls. Dokumentieren Sie die Ergebnisse im fachlichen Klassenmodell.

Die wesentliche Quelle sind Interviews mit den fachlichen Experten. Aber auch die anderen Modelle sind hilfreich. So können Sie z.B. aus einem detaillierten Prozessablaufdiagramm ableiten, welche Geschäftsobjekte für welchen Prozessschritt relevant sind. Jede Aktivität muss sich auf konkrete Entitäten beziehen. So bezieht sich z.B. der Prozess "Produkt beauftragen" auf die Geschäftsobjekte

"Auftrag", "Kunde" und "Produkt". Analog können Sie auch aus Use-Cases Anhaltspunkte ableiten. Falls diese detaillierten Beschreibungen nicht vorliegen oder sich nicht mit vertretbarem Aufwand erstellen und abstimmen lassen, können Sie die Geschäftsobjekte aus Schnittstellenbeschreibungen heraus ableiten. Nutzen Sie auch die Hilfestellungen in [HGG12] und [Rup07].

Ordnen Sie die gesammelten Kernentitäten den fachlichen Domänen des funktionalen Referenzmodells beziehungsweise den Komponenten des fachlichen Komponentenmodells zu. Falls nicht zuordenbar, sind es Kandidaten für eine neue fachliche Domäne. Stimmen Sie dies mit den verantwortlichen Unternehmensarchitekten ab.

2. Legen Sie mögliche fachliche Komponenten fest.

Die identifizierten Kernentitäten sind Kandidaten für fachliche Komponenten. Ordnen Sie die eng mit diesen Kernentitäten verbundenen Entitäten den Kernentitäten zu. So gehört z.B. das Geschäftsobjekt „Auftragstyp“ zur Kernentität „Auftrag“. Kriterium für die Zuordnung ist die enge Kopplung.

3. Analysieren Sie die Beziehungen zwischen den identifizierten fachlichen Komponenten.

Um die Abhängigkeiten zwischen den fachlichen Komponenten zu ermitteln und um zyklische Abhängigkeiten zu vermeiden, müssen alle Assoziationen zwischen den Geschäftsobjekten der verschiedenen fachlichen Komponenten analysiert werden. Durch Festlegung von Navigationsbeziehungen kann die Richtung sichtbar gemacht werden. Durch die Auflösung der Abhängigkeiten zwischen den so identifizierten fachlichen Komponenten wird das Gesamtmodell konsolidiert.

Stellen Sie sicher, dass fachlich eng zusammenhängende Services in fachlichen Komponenten im fachlichen Komponentenmodell zusammengefasst werden und wenige Abhängigkeiten zwischen den fachlichen Komponenten bestehen.

4. Legen Sie die Kandidaten für die entitätenzentrierten Services fest.

Neben den Verwaltungs-Services ("CRUD"), wie „Auftrag anlegen“, sind die verschiedenen Geschäftsregeln mögliche Kandidaten für diese Art von Business-Services.

5. Überarbeiten Sie das funktionale Referenzmodell und das fachliche Komponentenmodell.

Erweitern oder verändern Sie das funktionale Referenzmodell und/oder das fachliche Komponentenmodell um die Servicekandidaten (als fachliche Funktionen) und die fachlichen Komponenten, wenn diese noch nicht enthalten sind.

Hinweis

Das fachliche Komponentenmodell folgt der Strukturierung in fachliche Domänen des funktionalen Referenzmodells. Die fachlichen Domänen sind grobgranulare fachliche Komponenten. Im fachlichen Komponentenmodell werden die fachlichen Domänen gegebenenfalls weiter detailliert.

In der Projektarbeit wird in der Regel am fachlichen Komponentenmodell gearbeitet. Die Ergebnisse werden dann ins funktionale Referenzmodell konsolidiert. Beide Modelle können aber auch gleichzeitig gepflegt werden.

Beispiel EasyHouse

In Bild 11.14 finden Sie ein fachliches Klassenmodell aus dem EasyHouse-Beispiel aus [HGG12]. In diesem sind die Kernentitäten Partner, Smart-Home-Auftrag und Geräte hervorgehoben. Die "CRUD"-Verwaltungs-Services sind gleichzeitig Kandidaten für entitätenzentrierte Services für alle Kernentitäten.

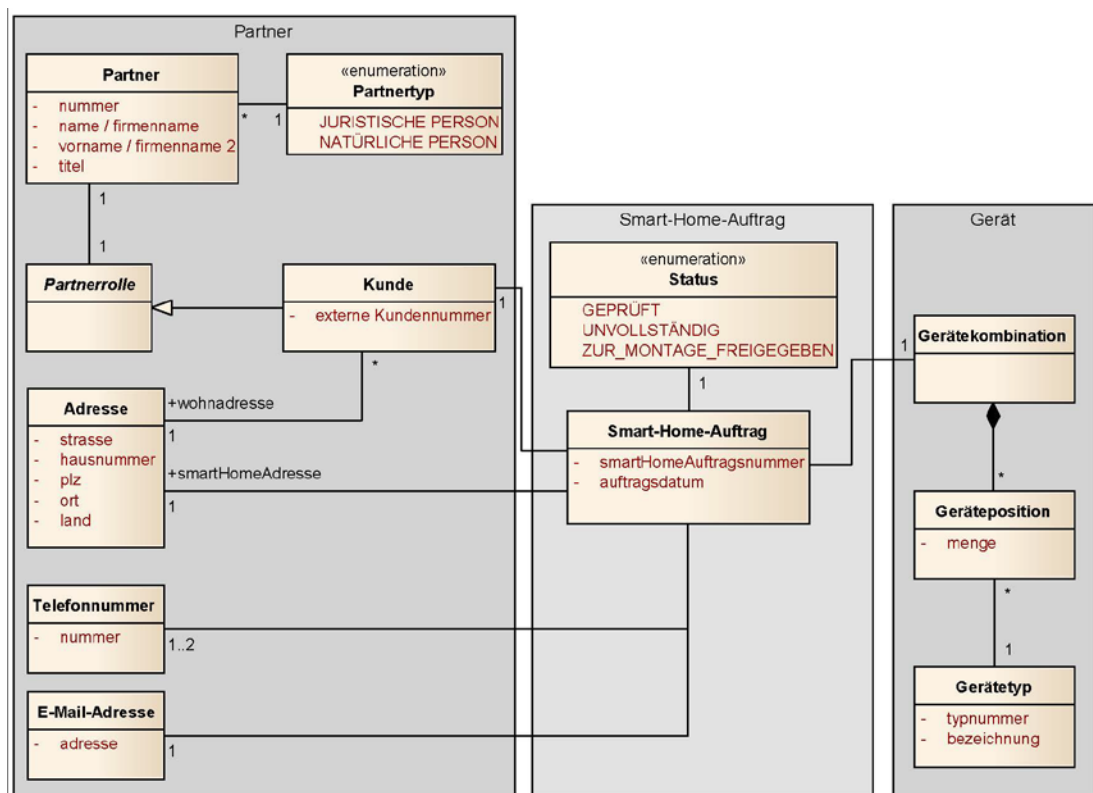


Bild 11.14 Ausschnitt aus einem fachlichen Klassenmodell vom
EasyHouse

Schritt 4: Identifizieren Sie die Kandidaten für aufgabenorientierte Services

Aus den ablaforientierten Modellen (Dokumentation der Use-Cases und/oder Geschäftsprozesse) werden Kandidaten für aufgabenorientierte Services abgeleitet. Die Überblicksdokumentation wird im Rahmen der Ableitung verfeinert und gegebenenfalls geändert.

Empfehlung

Erstellen Sie, wenn vom Aufwand her darstellbar, einen GUI-Prototyp. GUI-Prototypen dienen als Diskussionsgrundlage für die Anwender und stellen sicher, dass die Anforderungen „richtig“ verstanden wurden.

Folgende iterativ auszuführende Schritte sind hierfür notwendig:

1. Analysieren Sie die Aktivitäten.

Analysieren Sie die Use-Cases bzw. Dialoge von GUI-Prototypen, sofern vorhanden, oder Teilprozesse beziehungsweise Aktivitäten im Prozessablaufdiagramm oder Swimlane-Diagramm. Erstellen oder verfeinern Sie diese Modelle, wenn diese noch nicht in der erforderlichen Granularität vorhanden sind. Eine Dokumentation der nichtfunktionalen Anforderungen und zu erwartenden Fehler- und Ausnahmefälle ist zudem hilfreich.

Jeder nicht manuell ausgeführte Use-Case oder ggf. jede Aktivität/jeder Teilprozess ist ein Servicekandidat (und auch ein Kandidat für eine fachliche Funktion).

2. Überprüfen Sie den Projektkontext.

Filtern Sie die Servicekandidaten heraus, die nicht im Projektkontext liegen. Diese werden nicht weiter betrachtet. Ein typischer Fall hierfür ist die Nutzung von Funktionalitäten einer Standardsoftware, deren Anpassung kein Bestandteil des Projekts ist.

3. Filtern Sie entsprechend der Servicekategorien.

Filtern Sie die Servicekandidaten heraus, die Geschäftsregeln, Bedingungslogik oder Ablauflogik enthalten.

Die Ablauflogik wird über Orchestration Services umgesetzt, um eine höhere Flexibilität zu erzielen. Die Geschäftsregeln und die Bedingungslogik werden

über entitätenzentrierte Services abgebildet, um Daten und Geschäftsregeln zu bündeln. Hierzu kommt häufig auch eine Rule-Engine zum Einsatz.

Gleichen Sie die bereits ermittelten entitätenzentrierten Services mit den hier ermittelten ab und verändern oder ergänzen Sie die Sammlung von entitätenzentrierten Services.

4. Fassen Sie zusammen oder zerlegen Sie die aufgabenorientierten Servicekandidaten.

Um eine möglichst einheitliche Granularität zu erreichen, müssen die Services ggf. zusammengelegt oder weiter zerlegt werden. Die aufgabenorientierten Servicekandidaten sollten sich funktional möglichst nicht überschneiden.

Berücksichtigen Sie hier insbesondere auch die in der Service-Strategie geforderte Granularität.

5. Überarbeiten Sie das funktionale Referenzmodell und das fachliche Komponentenmodell.

Ordnen Sie die identifizierten Servicekandidaten in das fachliche Komponentenmodell ein. Erweitern oder verändern Sie das fachliche Komponentenmodell um die fachlichen Servicekandidaten und die fachlichen Komponenten, wenn diese noch nicht enthalten sind.

Nach Fertigstellung des fachlichen Komponentenmodells gleichen Sie dieses mit dem funktionalen Referenzmodell ab, erstellen Sie Änderungsvorschläge und stimmen Sie diese mit den Unternehmensarchitekten ab.

Empfehlung

Schnittstellen zwischen den fachlichen (Teil-)Komponenten im fachlichen Komponentenmodell lassen sich gut aus den Abhängigkeiten zwischen Use-Cases ableiten.

Wichtig

Bei der Ableitung der aufgabenorientierten Services können Sie im Gegensatz zur Ableitung der Entitätenzentrierten Services iterativ vorgehen. Die Verfeinerung der Use-Cases kann entsprechend der ausgewählten Realisierungsanforderungen (siehe [HGG12]) erfolgen.

Die identifizierten Business-Services sind in diesem Beispiel gleichzeitig die "idealen" Business-Services und lassen sich auch so in der Realität abbilden.

Schritt 5: Festlegung der "idealen" Business-Services

Durch die Analyse der Komponenteninteraktion werden die "idealen" (= gewünschten) Business-Services festgelegt. Auf Grundlage der ablaforientierten

Modelle (übergreifende Use-Case-Szenarien oder Geschäftsprozessabhängigkeiten) wird das Zusammenspiel der fachlichen Komponenten ermittelt. Hierzu verwendet man das fachliche Komponentenmodell auf der Überblicksebene und das UML-Kommunikationsdiagramm (siehe [Rup07]) für Detailsichten.

1. **Analysieren Sie alle Use-Cases (beziehungswise Geschäftsprozesse) übergreifend mittels Kommunikationsdiagrammen (Szenarien).**
Erstellen Sie Szenarien, in denen Sie die Abhängigkeiten zwischen Use-Cases als Ausstiegs- /Rücksprungsschritte zwischen Szenarien unterschiedlicher Use-Cases modellieren. Ordnen Sie soweit möglich Geschäftsobjekte zu. Verfeinern Sie die Service-Schnittstellen (u.a. Methoden und deren Signaturen). Leiten Sie Schnittstellen zwischen den fachlichen Komponenten im fachlichen Komponentenmodell aus den Verbindungen im Kommunikationsdiagramm ab.
2. **Verfeinern Sie die Modelle und legen Sie die Business-Services und deren Schnittstellen und Kontrakte im Detail fest** (siehe [Sta09]).
3. **Gleichen Sie die resultierenden Business-Services mit der Service-Strategie ab und verschmelzen Sie gegebenenfalls Business-Services.**
Durch die Service-Strategie werden neben Implementierungsvorgaben insbesondere die Arten von Services und deren Granularität festgelegt.
4. **Stimmen Sie die Modelle mit den verschiedenen Verantwortlichen ab und führen Sie eine Entscheidung über den Serviceschnitt herbei.**
Dies ist die Grundlage für das detaillierte Servicedesign und die Serviceentwicklung.

Stimmen Sie insbesondere auch die fachlichen Funktionen und deren Einordnung im funktionalen Referenzmodell mit den verantwortlichen Unternehmensarchitekten ab.

Durch den hier beschriebenen Best-Practice-Ansatz finden Sie eine geeignete Granularität für die "idealen" Business-Services, die die „Units of work“ zur Umsetzung der Projektanforderungen kapseln.

Schritt 6: Abgleich der "idealen" Business-Services mit der Ist-Situation

Wie in Abschnitt 11.2 ausgeführt, ist der Abgleich mit der Realität Grundlage, um das Projekt wirtschaftlich und im Zeitrahmen durchzuführen. Hierzu müssen die bestehende IT-Landschaft analysiert und die vorhandenen fachlichen und technischen Möglichkeiten der Legacy-Anwendungen festgestellt werden. Grundlage für den Abgleich sind die "idealen" Business-Services. Hieraus können Anforderungen an die Applikations-Services für die Anbindung der Legacy-Anwendungen abgeleitet werden.

Folgende Schritte sind hierfür notwendig:

1. Identifikation der zur Unterstützung von Business-Services in Frage kommenden Applikations-Services anhand der Geschäftsobjekte und der Funktionalitäten.
2. Analyse der Applikations-Services in Bezug auf die Abdeckung der geforderten Funktionalitäten und benötigten Daten. Sie müssen sich Fragen stellen wie z.B.:
 - Können bestehende Applikations-Services wiederverwendet werden?
 - Müssen Applikations-Services erweitert, kombiniert oder verändert werden?
 - Müssen neue Applikations-Services bereitgestellt werden?

Damit erlangen Sie ein Verständnis davon, ob der Business-Service überhaupt und gegebenenfalls mit welchem Aufwand in dieser Form mit den relevanten Anwendungsfunktionen und Daten beliefert werden kann. Auf dieser Basis kann eine Entscheidung darüber getroffen werden, ob die Applikations-Services entsprechend bereitgestellt (d.h. ggf. verändert oder neu erstellt werden) oder aber der "ideale" Business-Service so nicht umgesetzt wird.

3. Festlegung der Business-Services und Applikations-Services.
Dokumentation der resultierenden Business-Services und Applikations-Services.
Dies schließt gegebenenfalls eine Veränderung der "idealen" Business-Services entsprechend der Entscheidung aufgrund des Analyseergebnisses mit ein.

Gegebenenfalls wird eine Komponentisierung der Legacy-Anwendungen erforderlich. Hier gibt es einige bewährte Aktivitäten:

- a) Identifizieren der noch relevanten Funktionsblöcke
- b) Schneiden von grobkörnigen Komponenten (Services) mit klaren Verantwortlichkeiten
- c) Entkopplung von Interface und Implementierung
- d) Entkopplung von Geschäftsprozesssteuerung, Fachkomponenten und I/O-Logik
- e) Herauslösung kontext-, mandanten- und länderspezifischer Teile in Konfigurationsanteile
- f) Kapselung von atomaren Transaktionen soweit möglich

Nach der Festlegung der Business-Services und Applikations-Services müssen diese umgesetzt werden. Dies kann sowohl agil als auch wasserfallartig erfolgen. Wir empfehlen eine gesteuerte evolutionäre Umsetzung ("Managed Evolution"), um frühzeitig und regelmäßig Feedback zu erhalten (siehe [Kra04]). Wie bei allen anderen Vorgehensweisen gibt es auch dabei Risiken, wie in Abschnitt 11.2 bereits ausgeführt.

Die resultierenden Business-Services sind bereits in Bild 11.8 beschrieben. In Bild 11.15 finden Sie in einer Grafik nochmals die Zusammenhänge zwischen den wesentlichen Modellen auf einen Blick zusammengestellt.

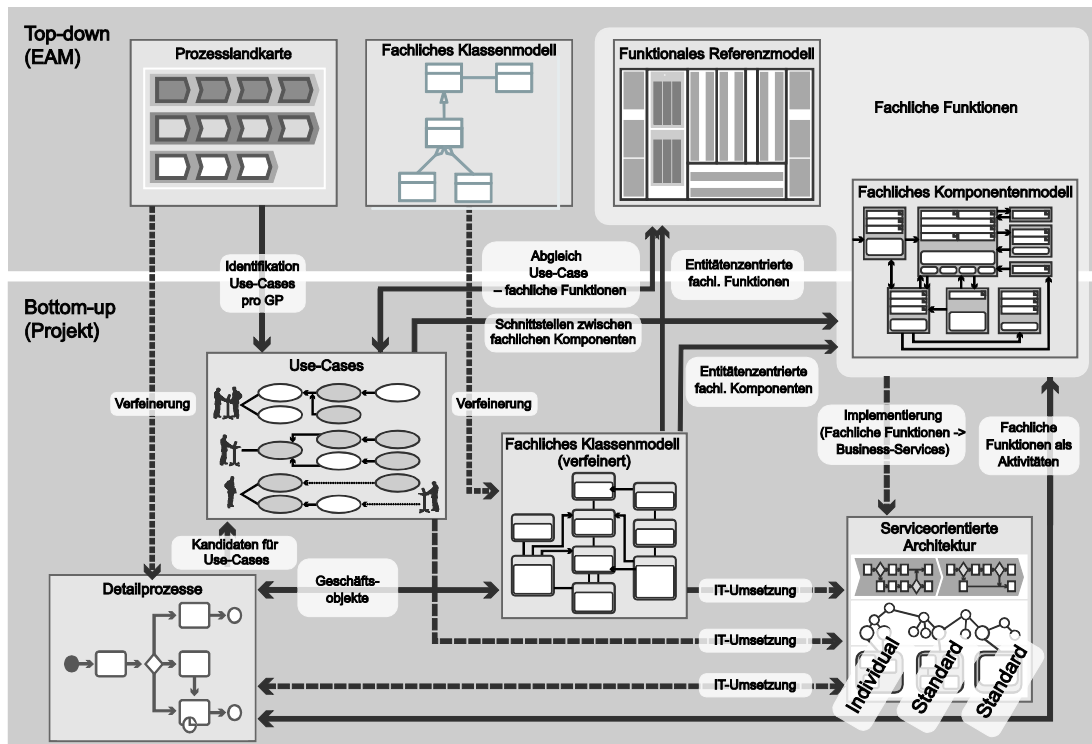


Bild 11.15 Beispiel für das Zusammenspiel der verschiedenen Modelle

Das Wesentliche zur Ableitung von Business-Services:

Serviceorientierte Architekturen versprechen Business-Agilität – die Fähigkeit zur flexiblen und effektiven Umsetzung sich immer schneller ändernder Geschäftsanforderungen. Voraussetzung dafür ist Flexibilität in der IT. Dazu müssen die Geschäftsre-

geln und Business-Funktionen von der Prozesslogik getrennt und in Business-Services gekapselt werden. Die bestehende IT-Landschaft muss in Komponenten mit Serviceschnittstellen "zerschlagen" und eine Integrationsarchitektur eingezogen werden, die es erlaubt, die geschäftlichen Abläufe transparent und änderungsfreundlich zu unterstützen.

Bei der Ableitung der Business-Services auf Basis der realen Geschäftsanforderungen werden die Top-down vorgegebenen Strukturen bottom-up mit Leben gefüllt. Ein funktionales Referenzmodell und ein fachliches Komponentenmodell mit fachlichen Domänen und fachlichen Komponenten geben neben gegebenenfalls einer Prozesslandkarte den übergeordneten Bauplan vor, der für die SOA-Governance genutzt werden kann. Business-Services stehen als wiederverwendbarer Katalog zur Orchestrierung zur Verfügung. Dem Wunschbild einer flexiblen IT, die sich an verändernde Geschäftsprozesse anpasst, kommen Sie damit in jedem Fall einen Schritt näher. Nutzen Sie die Hilfestellungen aus diesem Abschnitt für die Ableitung Ihrer Business-Services.

Die Migration zu einer serviceorientierten Landschaft ist sowohl fachlich und technisch als auch organisatorisch sehr komplex und erfordert einen langen Atem. Der Ausgangspunkt ist die Festlegung der Service-Strategie. Doch nicht immer ist SOA der „richtige“ Lösungsansatz; Flexibilität wird nicht in allen Anwendungsbereichen benötigt. Die Entscheidung für SOA in einem Anwendungsbereich erfordert ein nachhaltiges Management-Commitment und ein Evangelisieren der verschiedenen Stakeholder-Gruppen (siehe [Kra04]).