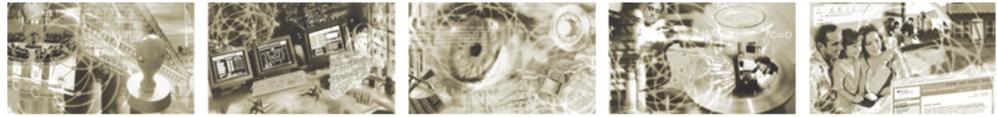




Bundesamt
für Sicherheit in der
Informationstechnik



Band B, Kapitel 7: IT-Dienste

Bundesamt für Sicherheit in der Informationstechnik
Postfach 20 03 63
53133 Bonn

Tel.: +49 22899 9582-0

E-Mail: hochverfuegbarkeit@bsi.bund.de

Internet: <https://www.bsi.bund.de>

© Bundesamt für Sicherheit in der Informationstechnik 2013

Inhaltsverzeichnis

1	Einleitung	5
2	Dienstabhängigkeiten	6
2.1	Horizontale Abhängigkeiten	6
2.2	Vertikale Abhängigkeiten	6
2.3	Abhängigkeitsmodell/Verfügbarkeitsgraf	7
2.3.1	Abhängigkeitsmodell für die technische Dienstrealisierung	8
2.3.2	Abhängigkeiten der Dienstleistungen in den Bereichen IT-Organisation, IT-Personal und IT-Infrastruktur	9
3	Dienstorientierte Architekturen	11
3.1	Interoperabilität von Diensten	11
3.2	Lose Kopplung von Diensten	11
3.3	Dienst-Autonomie	12
3.4	Netzwerk-Einbindung	12
3.5	Öffentliche Schnittstelle	12
3.6	Kapselung von Implementationsdetails	12
3.7	Plattformunabhängigkeit	12
3.8	Wiederverwendbarkeit	13
3.9	Service-Repository	13
3.10	Enterprise-Service-Bus (ESB)	13
4	Dienst-Management	15
4.1	Dienstpriorisierung	16
4.2	Dienstredundanz	17
4.2.1	Strukturelle Redundanz	17
4.2.2	Geografische Redundanz	17
4.3	Präventive Maßnahmen für kritische Geschäftsprozesse	17
4.3.1	Risikoanalyse	18
4.3.2	Monitoring	18
4.3.3	Wartung	18
4.3.4	Kapazitätsplanung	18
4.3.5	Notfallplanung	18
5	Dienstfindung in dynamischen Netzen	19
5.1	Universal Plug and Play (UPnP)	19
5.2	Service Location Protocol (SLP)	19
5.3	Java Intelligent Network Interface (Jini)	19
6	Attribute und Parameter für die Dienstbewertung	22
6.1	Dienstgüte/Quality of Service	22
6.2	Grundlagen für Service Level Agreements	23
6.2.1	Dienstgüteparameter	23
6.2.2	Parameter für das Monitoring	23
	Anhang: Verzeichnisse	24
	Abkürzungsverzeichnis	24
	Glossar	24

Literaturverzeichnis.....24

Abbildungsverzeichnis

Abbildung 1: Horizontale Abhängigkeiten.....6
Abbildung 2: Vertikale Abhängigkeiten.....7
Abbildung 3: Technisches Abhängigkeitsmodell des E-Mail Dienstes.....8
Abbildung 4: Abhängigkeiten der Dienstleistungen.....9
Abbildung 5: Bereitstellung und Aufruf eines Dienstes in einer SOA.....14
Abbildung 6: Aufgaben des Dienst-Managements.....16
Abbildung 7: Dienstfindung mit Jini.....20

Tabellenverzeichnis

1 Einleitung

Der Wandel von der Produktions- zur Dienstleistungsgesellschaft stellt die IT-Verantwortlichen in Unternehmen und Behörden nicht nur stetig wachsenden Ansprüchen gegenüber, sondern ermöglicht auch eine neue Sicht auf das eigene IT-Umfeld. Traditionell stand eher die Entwicklung und der Betrieb von Informationstechnik im Mittelpunkt, was oftmals nur zu einer lokalen Sicht auf einzelne IT-Ressourcen (z.B. Netzwerk- oder Infrastrukturkomponenten) führte. In der Dienstleistungsgesellschaft steht der Dienstabnehmer und seine Qualitätsanforderungen im Mittelpunkt. Die Erbringung (auch) von IT-Dienstleistungen muss bedarfsgerecht erfolgen und der Bedarf kann nicht durch Leistungsmerkmale einzelner Komponenten definiert werden. Die Entwicklung hin zur Dienstleistungsorientierung in der Informationstechnik hat auch zur Folge, dass Anforderungen an Geschäftsprozessen, welche von IT-Dienstleistungen gestützt werden, ausgerichtet sind. Neben der Dienstleistungsorientierung kommt es somit auch zu einer Prozessorientierung, die im 3-Ebenen-Modell in diesem Kompodium eingeführt wurden. In diesem Modell kann die Komplexität der IT durch Kapselung in Subservices (Assistent Services) reduziert werden, ohne dass die bestehenden Abhängigkeiten der Ressourcen untereinander vernachlässigt werden. Zusätzlich zur traditionellen Sicht auf die IT-Komponenten wird dadurch eine neue Betrachtungsebene für die Bewertung und das Design von Services geschaffen. Beides, die Bewertung erbrachter Dienstleistungen als auch das Design von Services, welches sich nach Best Practices richtet, sind grundlegende Voraussetzungen für eine bedarfsgerechte Erbringung von Dienstleistungen.

Weiterführende Informationen zur Bewertung von Dienstleistungen sind in den Beiträgen

- Phase I
- 3- Ebenen-Service-Modell
- HV-Assessment und Benchmarking

enthalten.

Anhaltspunkte zum bedarfsgerechten Service-Design können für die Ebene der Ressourcen den weiteren Beiträgen im Band B

- Netzwerk
- Server
- Speichertechnologien
- Software
- Überwachung
- Infrastruktur
- Organisation und Personal

entnommen werden. In diesem Beitrag werden übergreifende Aspekte wie Dienstabhängigkeiten, dienstorientierte Architekturen, Dienstmanagement, Dienstfindung und Attribute für die Dienstbewertung thematisiert.

2 Dienstabhängigkeiten

Insbesondere für die Bewertung und Optimierung von HV-Umgebungen sind die Aspekte der Dienstabhängigkeit und -modellierung von Interesse. Die Modellierung beschreibt, mit welchen IT-Ressourcen eine IT-Dienstleistung erbracht wird. Sämtliche Aspekte der Modellierung werden ausführlich im HV-Kompodium im Rahmen der HV-Methodik in der Phase M betrachtet.

In der Regel stützt sich die betrachtete IT-Dienstleistung auf IT-Dienste, die wiederum in einer gewissen Abhängigkeit zu weiteren Sub- oder Teildiensten stehen. Dienstabhängigkeiten beschreiben also prozessuale Abhängigkeiten zwischen IT-Diensten, während mit der Modellierung die konkrete systemtechnische Realisierung einer IT-Dienstleistung verbunden ist.

Für den Anbieter einer IT-Dienstleistung ist die Kenntnis von Dienstabhängigkeiten wichtig, weil er dadurch erforderliche Informationen für Analyse, Konzeption und Betrieb der IT-Ressourcen erhält. Dabei können, wie nachfolgend erläutert, zwei Arten von Dienstabhängigkeiten (horizontale und vertikale) unterschieden werden.

2.1 Horizontale Abhängigkeiten

Horizontale Dienstabhängigkeiten bestehen zwischen Diensten auf derselben Funktionsebene. Dies führt zu einer Modellierung von Diensten in Dienstketten (Supply Chains). Die Abbildung 1 verdeutlicht anhand der schwarzen Beziehungspfeile die bestehenden Abhängigkeiten.

Im vorliegenden Beispiel ist Dienst A für die Erbringung der auf gleicher Ebene befindlichen Dienste B und C erforderlich. Diese sind damit von Dienst A funktional abhängig, d. h., ein Ausfall dieses Dienstes führt unmittelbar zu einer Nichtverfügbarkeit oder Beeinträchtigung der abhängigen Dienste B und C.

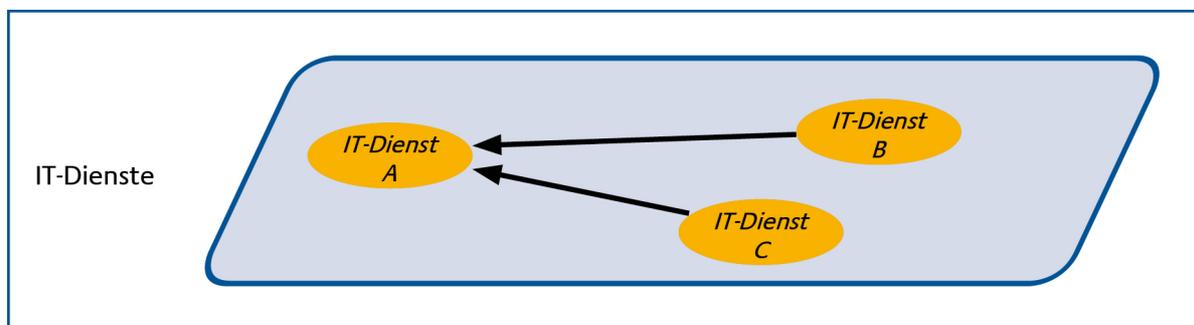


Abbildung 1: Horizontale Abhängigkeiten

2.2 Vertikale Abhängigkeiten

Vertikale Dienstabhängigkeiten existieren zwischen Diensten auf unterschiedlichen Funktionsebenen. Es entstehen hierdurch Diensthierarchien. Im vorliegenden Beispiel (siehe Abbildung 2) werden die auf einer untergeordneten Funktionsebene liegenden Dienste B und C für die Erbringung von Dienst A benötigt. Somit besteht eine Ebenen übergreifende funktionale Abhängigkeit des Dienstes A von den anderen beiden Diensten. Eine Störung von Dienst B oder C führt damit zu einer Nichtverfügbarkeit des abhängigen Dienstes A.

Hierbei gilt, dass ein Dienst einerseits als sogenannter Antecedent für einen anderen Dienst notwendig sein kann, andererseits aber auch selbst als Dependent in einem Abhängigkeitsverhältnis zu der Leistungserbringung eines weiteren Dienstes stehen kann.

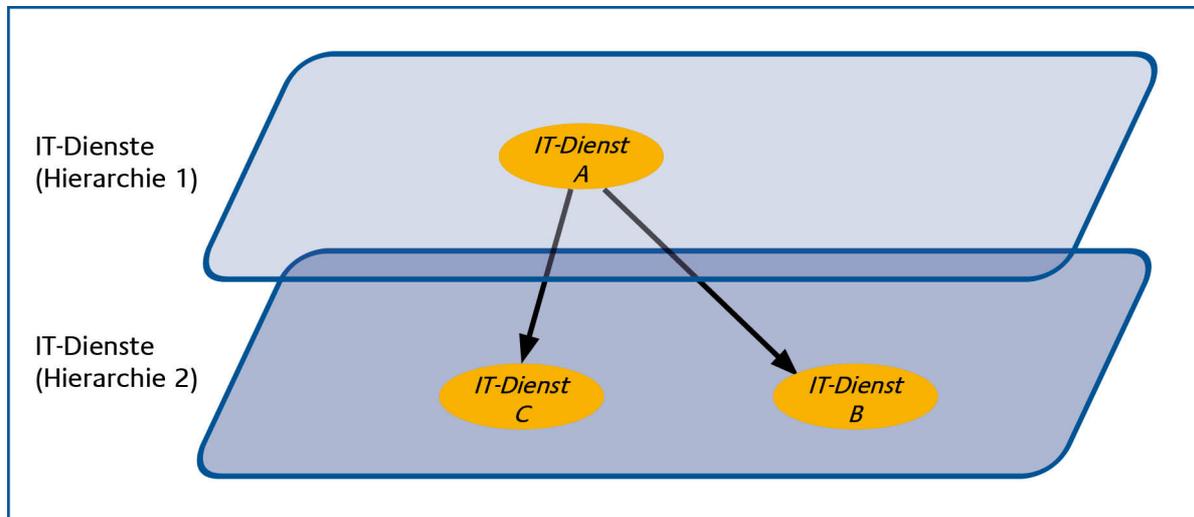


Abbildung 2: Vertikale Abhängigkeiten

2.3 Abhängigkeitsmodell/Verfügbarkeitsgraf

In der Praxis liegen häufig Mischformen aus horizontalen und vertikalen Abhängigkeiten vor. Applikations-Dienste sind zum Beispiel in der Regel aus mehreren anderen IT-Diensten zusammengesetzt, von deren Verfügbarkeit sie funktional abhängen. Diese können aber auch untereinander in einem Abhängigkeitsverhältnis stehen. Die IT-Dienste selbst werden wiederum mit Hilfe von IT-Komponenten realisiert. Auch auf der Ebene dieser Komponenten bestehen im Allgemeinen wechselseitige Abhängigkeitsbeziehungen.

Nachfolgend wird dieser Sachverhalt zunächst für die **technische Dienstrealisierung** mittels einer vereinfachten Darstellung des E-Mail-Dienstes anhand eines Abhängigkeitsmodells (siehe Abbildung 3) veranschaulicht. Die vier möglichen Szenarien für bestehende technische Abhängigkeiten werden dann am Beispiel erläutert. In einem weiteren Schritt werden die Dienstleistungen aus den Bereichen IT-Personal, IT-Organisation und IT-Infrastruktur in die Betrachtung der Abhängigkeiten mit einbezogen.

2.3.1 Abhängigkeitsmodell für die technische Dienstrealisierung

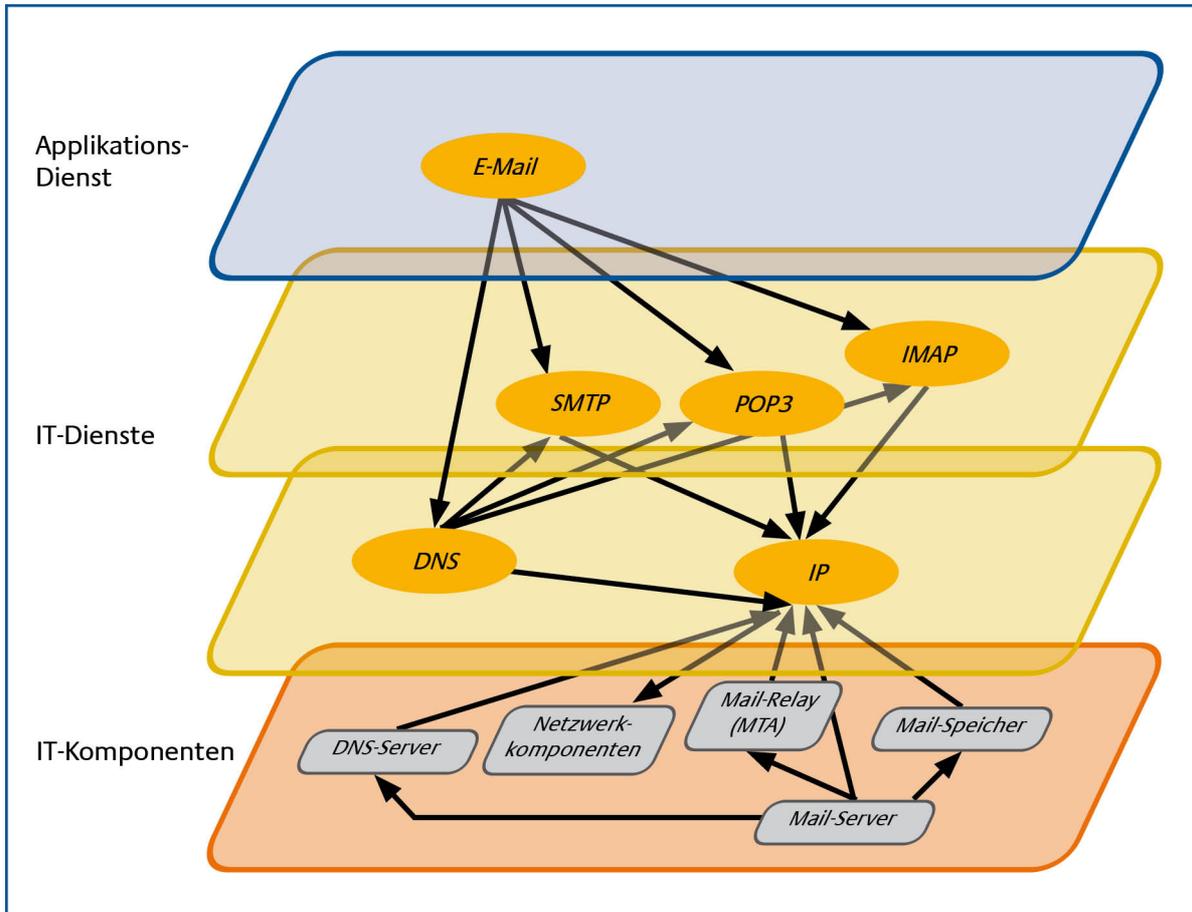


Abbildung 3: Technisches Abhängigkeitsmodell des E-Mail Dienstes

Die Abbildung 3 stellt in einer vereinfachten Form die zur Erbringung des Applikations-Dienstes E-Mail erforderlichen IT-Dienste und IT-Komponenten dar. Die vier möglichen Szenarien der technischen Abhängigkeiten werden nachfolgend erläutert:

1. Abhängigkeiten zwischen Applikations-Dienst und untergeordneten Diensten

Wie in der Abbildung 3 ersichtlich, setzt sich der E-Mail-Dienst aus verschiedenen untergeordneten Diensten (DNS, SMTP, POP3 und IMAP) zusammen. Aufgrund dieser bestehenden vertikalen Abhängigkeiten ist ein funktionsfähiger E-Mail-Dienst nur möglich, wenn alle diese Dienste zur Verfügung stehen. Ein Ausfall des SMTP-Dienstes würde beispielsweise den E-Mail-Austausch verhindern. Somit sind DNS, SMTP, POP3 und IMAP Antecedenten hinsichtlich der Leistungserbringung des Applikationsdienstes.

2. Abhängigkeit zwischen untergeordneten Diensten

SMTP, POP3 und IMAP stehen in funktional vertikaler Abhängigkeit zum IP-Dienst, da dieser von ihnen zur Kommunikation genutzt wird. Somit sind sie hinsichtlich dieses Basis-Dienstes Dependents, da sie ihre Funktionalität im Falle eines Ausfalls der Netzwerkkommunikation nicht aufrechterhalten können. Gleiches gilt für den auf gleicher Ebene wie der IP-Dienst befindlichen DNS-Dienst, sodass in diesem Fall gleichzeitig auch horizontale Abhängigkeiten bestehen. Der DNS-Dienst ist damit nicht nur, wie oben beschrieben, Antecedent, sondern hinsichtlich des IP-Dienstes auch Dependent.

3. Abhängigkeit zwischen untergeordneten Diensten und IT-Komponenten

Der IP-Dienst wird mittels der Netzwerkkomponenten realisiert. Eine Nichtverfügbarkeit dieser IT-Komponenten ist somit gleichbedeutend mit einem Ausfall dieses Dienstes und damit der gesamten Netzwerkkommunikation. Dies führt schließlich auch dazu, dass der Dienst E-Mail nicht mehr zur Verfügung steht.

4. Abhängigkeiten zwischen IT-Komponenten

Im vorliegenden Beispiel werden E-Mails von einem Mailserver mittels des Mailrelays verschickt. Diese Funktionalität ist bei einer Nichtverfügbarkeit des Mailrelays nicht mehr gegeben und bewirkt letztlich auch einen Ausfall des gesamten E-Mail-Dienstes.

2.3.2 Abhängigkeiten der Dienstleistungen in den Bereichen IT-Organisation, IT-Personal und IT-Infrastruktur

Abhängigkeiten bestehen ebenso auf der Ebene der Dienstleistungen. Diese werden in der nachfolgenden Abbildung 4 verdeutlicht.

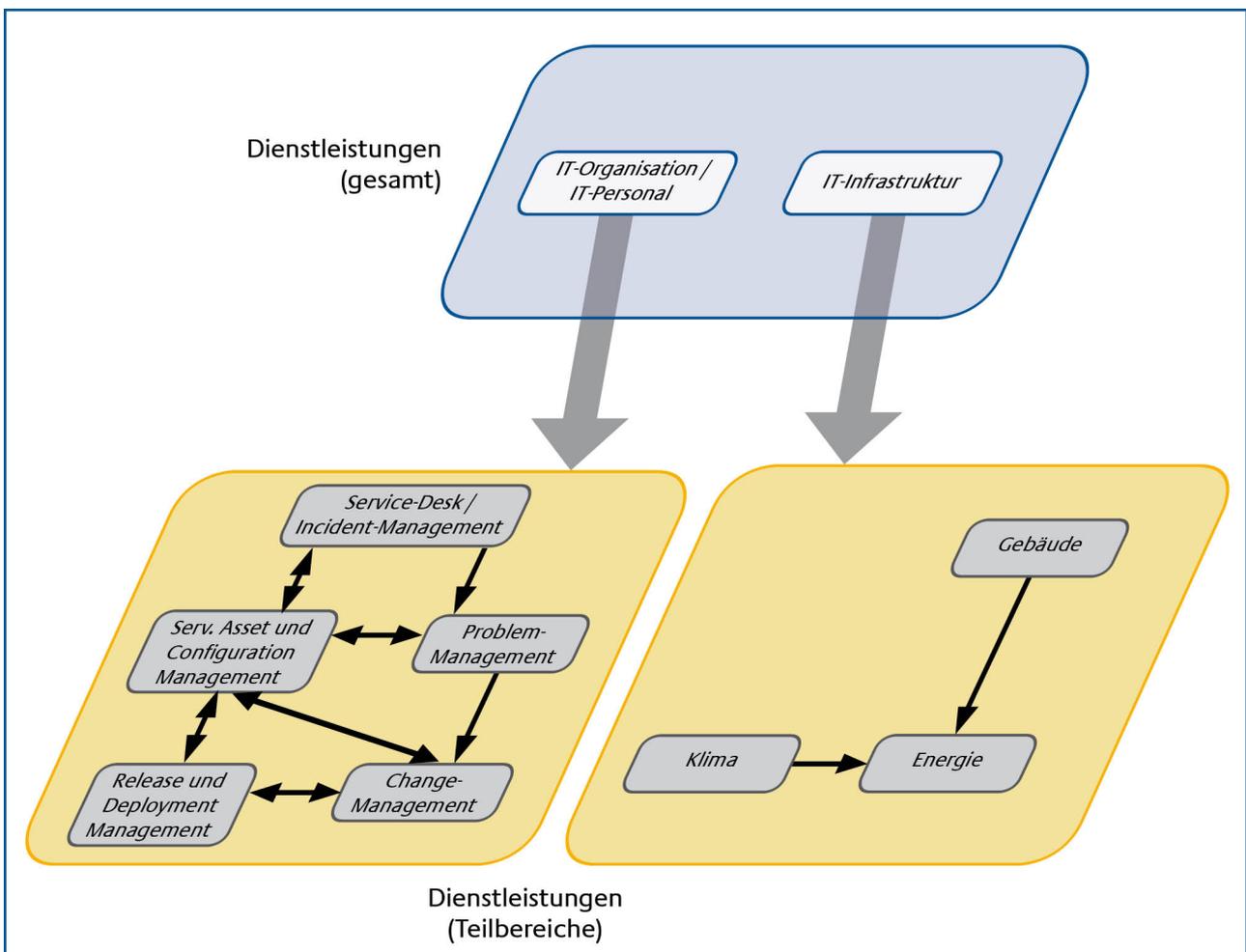


Abbildung 4: Abhängigkeiten der Dienstleistungen

Für die Dienstleistungen aus dem Bereich IT-Infrastruktur sind hierbei exemplarisch die Abhängigkeiten zwischen Gebäude, Klima und Energie herangezogen worden. Offensichtlicherweise kann die Klimatisierung, insbesondere wichtig für Server-Räume mit hoher Wärmeentwicklung, nicht ohne Energie funktionieren. Das Gebäude ist ebenfalls auf die ausreichende Bereitstellung von Energie angewiesen, da ohne diese beispielsweise elektronische

Zutrittskontrollen, Gefahrmeldungen, Beleuchtung etc. nicht verfügbar sind. In dem Beitrag „Infrastruktur“ des HV-Kompendiums wird detailliert auf die Zusammenhänge bzgl. der Infrastruktur eingegangen.

3 Dienstorientierte Architekturen

Der Einsatz von dienstorientierten Architekturen (SOA) hat in den letzten Jahren stark an Bedeutung gewonnen. Dienstorientierten Architekturen stellen flexible, anpassbare IT-Architekturen dar, die eine verteilte Datenverarbeitung unterstützen und die Abbildung von Geschäftsprozessen auf IT-Systeme vereinfachen. Einzelne Dienste werden dabei bedarfsgerecht, mittels der sogenannten Orchestrierung, für einen Geschäftsprozess zusammengesetzt.

Somit stehen bei dienstorientierten Architekturen nicht mehr einzelne Hard- und Softwarekomponenten im Vordergrund, sondern der mittels der modellierten Dienste vom Dienstanbieter zu realisierende Geschäftsprozess.

Auch wenn sich in der Literatur noch keine allgemeingültige Definition von dienstorientierten Architekturen durchgesetzt hat, existieren die nachfolgend angeführten anerkannten Merkmale, die eine SOA genauer bestimmen. Diese Eigenschaften ermöglichen die Umsetzung vieler Prinzipien der Verfügbarkeit in Form von technischen Maßnahmen. Detaillierte Information zu SOA können dem SOA-Security Kompendium entnommen werden.

3.1 Interoperabilität von Diensten

Durch eine Minimierung der bestehenden Abhängigkeiten zwischen den Diensten untereinander wird die Interoperabilität als Grundlage für Redundanz und Skalierbarkeit gewährleistet. Die Interoperabilität ermöglicht es beispielsweise den Dienstonutzern im Bedarfsfall auf vergleichbare redundante Dienste (z. B. eines anderen Dienstansbieters) schnell und unkompliziert zu wechseln. Dies ist in besonderem Maße in einem Hochverfügbarkeitsumfeld von Interesse, da er nicht mehr von der Verfügbarkeit eines einzelnen Dienstansbieters abhängig ist. Ist beispielsweise der von einem Dienstanbieter bereitzustellende DNS-Dienst nicht verfügbar, kann der auf die Dienstleistung angewiesene Nutzer einen von einem anderen Dienstanbieter angebotenen DNS-Dienst einbinden.

Dienstansbiestern hingegen ermöglicht die Interoperabilität, schnell und zuverlässig auf geänderte Anforderungen an die Geschäftsprozesse reagieren zu können, indem sie die Dienste neu miteinander kombinieren. Aufgrund der reduzierten Abhängigkeiten in der Architektur muss es daher nicht zu einer Beeinträchtigung des bestehenden IT-Systems oder laufender Geschäftsprozesse kommen. Bei einem bis dato nur mittels POP3 und SMTP realisierten E-Mail-Dienst könnte beispielsweise im Bedarfsfall dem Wunsch nach einem Zugriff und der Verwaltung von E-Mails mit IMAP entsprochen werden, indem dieser Dienst zusätzlich integriert wird.

3.2 Lose Kopplung von Diensten

Die einzelnen Dienste können lose gekoppelt für einen Geschäftsprozess zusammengesetzt werden. Hierzu sind die zwischen den einzelnen Diensten bestehenden Implementierungs-Abhängigkeiten auf ein Minimum reduziert. Dies ermöglicht eine exakte Abbildung des Geschäftsprozesses. Prozesse mit hohen Verfügbarkeitsanforderungen können somit von denen mit geringeren Anforderungen separiert und mögliche Störeinflüsse vermieden werden. Darüber hinaus kann beim Ausfall eines Dienstes, wie oben angeführt, relativ einfach zu einem kompatiblen Dienst gewechselt werden.

3.3 Dienst-Autonomie

Für den Anbieter von Diensten ist es wichtig, dass er autonome Dienste anbietet. Ein Dienst sollte ein in sich abgeschlossenes Konstrukt und damit in der Lage sein, seine Funktionalität unabhängig von anderen Diensten erbringen zu können. Um dies umzusetzen, muss er die gesamte benötigte Logik, Ressourcen und Umgebung selbst verwalten können. Somit findet mit der **Autonomie** ein weiteres wichtiges Prinzip der Verfügbarkeit bei dienstorientierten Architekturen Berücksichtigung.

Obwohl, wie dargestellt, SOA-Dienste so entworfen werden, dass sie möglichst autonom sind, steht in der Regel kein SOA-Dienst für sich alleine, da es sich bei einer dienstorientierten Architektur um ein komplexes Gebilde handelt, das aus einer Vielzahl von Sub-Diensten besteht. Somit können trotz der Autonomiebestrebungen Abhängigkeiten zu Sub-Diensten (vertikale Abhängigkeiten) bestehen.

3.4 Netzwerk-Einbindung

Jeder Dienst wird in einem Netzwerk zur Verfügung gestellt. Im HV-Umfeld muss das Netzwerk zur Nutzung dieser Dienste selbstverständlich auch hochverfügbar realisiert werden (siehe hierzu den Beitrag „Netzwerk“ im HV-Kompendium).

3.5 Öffentliche Schnittstelle

Jeder Dienst in einer dienstorientierten Architektur verfügt über eine von außen zugängliche, eindeutige und maschinenlesbare Schnittstelle, die offene Standards benutzt. Dies ist eine notwendige Voraussetzung zur Realisierung der losen Kopplung.

3.6 Kapselung von Implementationsdetails

Über die öffentliche Schnittstelle werden die Implementationsdetails eines Dienstes dem Dienstanutzer in gekapselter Form zur Verfügung gestellt. Dies bildet die Grundlage für Modellierung, **Virtualisierung**, **Skalierbarkeit** und **Autonomie**, da die bei monolithischen Systemlandschaften sonst übliche statische Zuordnung von Diensten und Hardware-Komponenten durch die Herauslösung der Dienste aufgehoben wird. Hierdurch wird eine sehr große Flexibilität des gesamten IT-Systems ermöglicht, da jeder Dienst auf jedem System lauffähig ist. Somit können beispielsweise Dienste bei einem Fehler der zugrundeliegenden Hardware-Plattform jederzeit auf ein anderes System verschoben und die Verfügbarkeit sichergestellt werden.

3.7 Plattformunabhängigkeit

Die Kommunikation innerhalb einer dienstorientierten Architektur erfolgt plattformunabhängig über Web-Services, wodurch ebenfalls eine Grundlage für die Virtualisierung geschaffen wird, da ein flexibler Einsatz der Dienste auf unterschiedlichsten Systemen ermöglicht wird. In Bezug auf die Anforderungen, die sich im HV-Umfeld ergeben, ist die Plattformunabhängigkeit darüber hinaus ein wesentlicher Aspekt der Redundanzausprägung Diversität.

3.8 Wiederverwendbarkeit

Einmal erstellte zentrale Dienste können in dienstorientierten Architekturen auch in einem anderen Umfeld kostengünstig immer wieder verwendet werden. Diese Wiederverwendbarkeit führt durch den Einsatz bereits erprobter und optimierter Dienstfunktionalitäten zu einer größeren Zuverlässigkeit und damit zu einer Erhöhung der Verfügbarkeit des Gesamtsystems.

3.9 Service-Repository

Um dem Dienstanbieter das Auffinden der Dienste zu ermöglichen, registriert der Dienstanbieter die von ihm angebotenen Dienste in einem Service-Repository. Hierbei erfolgt eine Kategorisierung der Dienste. Im Service-Repository werden Informationen über die verwendeten Schnittstellen und die Implementierung der Dienste hinterlegt. Somit bildet das Service-Repository eine Grundlage für Redundanz, indem es dem Dienstanbieter die Abfrage mehrerer gleichwertiger Dienste, die dabei auch von unterschiedlichen Dienstanbietern stammen können, ermöglicht. Zudem leistet das Service-Repository einen wichtigen Beitrag zur **Skalierbarkeit**, da in diesem die einzelnen Dienste gesammelt bereitgestellt werden, die durch die Orchestrierung zu neuen Diensten zusammengesetzt werden können.

In einer dienstorientierten Architektur ist das Auffinden der Dienste nur über das Service-Repository möglich. Somit stellt dieses einen Single Point of Failure dar. Zur Sicherstellung der Verfügbarkeit bietet sich damit insbesondere beim Service-Repository eine redundante Implementation an.

3.10 Enterprise-Service-Bus (ESB)

Der Enterprise-Service-Bus beschreibt eine Kommunikationsinfrastruktur mit besonderen Routing-Fähigkeiten, welche in dienstorientierten Architekturen zum nachrichtenbasierten Informationsaustausch eingesetzt werden kann. Zentrale Aufgabe eines ESB ist hierbei der Austausch von Daten in Form von Dienstaufrufen zwischen IT-Systemen oder Teilen von IT-Systemen. Zusätzlich übernimmt der Enterprise-Service-Bus Aufgaben wie Orchestrierung, Kommunikation, Verschlüsselung, Load-Balancing und weitere Querschnittsdienste.

Gerade im Zusammenhang hoher Anforderungen an die Verfügbarkeit leistet ein ESB wertvolle Dienste. Statt synchron über Prozeduraufrufe zu kommunizieren, was die gleichzeitige Verfügbarkeit von Dienstanbieter und Anbieter erfordert, erfolgt die Kommunikation asynchron. Hierzu muss sich der Dienstanbieter an dem Enterprise-Service-Bus registrieren. Anschließend können die Dienstanbieter ihre Dienstanforderungen auf dem Bus publizieren. Innerhalb dieser Dienstanforderung wird spezifiziert, wie das durch den Dienst erbrachte Ergebnis dem Dienstanbieter zugestellt werden kann. Der Enterprise-Service-Bus übernimmt die Weiterleitung der Nachrichten vom Dienstanbieter zum Dienstleister, wandelt Nachrichtenformate um und ruft automatisch bestimmte Konnektoren zur Anpassung von Schnittstellen auf.

Zudem kann eine ESB-Architektur auch als verteilte Architektur realisiert werden, um damit Leistungsengpässe zu vermeiden und Ausfallsicherheit zu gewährleisten. Weitere Hinweise und Architekturansätze können der spezifischen Fachliteratur entnommen werden [Chap04]. Eine besondere Herausforderung stellt die Unterstützung von Hochverfügbarkeit für zustandsbehaftete Dienste dar. Damit die Zustandsinformationen der Dienste auch bei Fail-Over-Verfahren nicht

verloren gehen, sind Mechanismen, wie Software-Lastverteiler mit einem „Share-State“-Ansatz, erforderlich [IONA08].

In der nachfolgenden Abbildung 5 wird der typische Ablauf von Bereitstellung und Abruf eines Dienstes in einer dienstorientierten Architektur dargestellt. Zunächst registriert der Dienstanbieter den von ihm bereitgestellten Dienst im Service-Repository. Dieses antwortet auf die Suchanfrage eines potenziellen Dienstanwenders mit einem Verweis auf diesen Dienst. Mit dieser Information kann der Dienstanwender die entsprechende Dienstbeschreibung beim Anbieter abrufen und anschließend den bereitgestellten Dienst aufrufen.

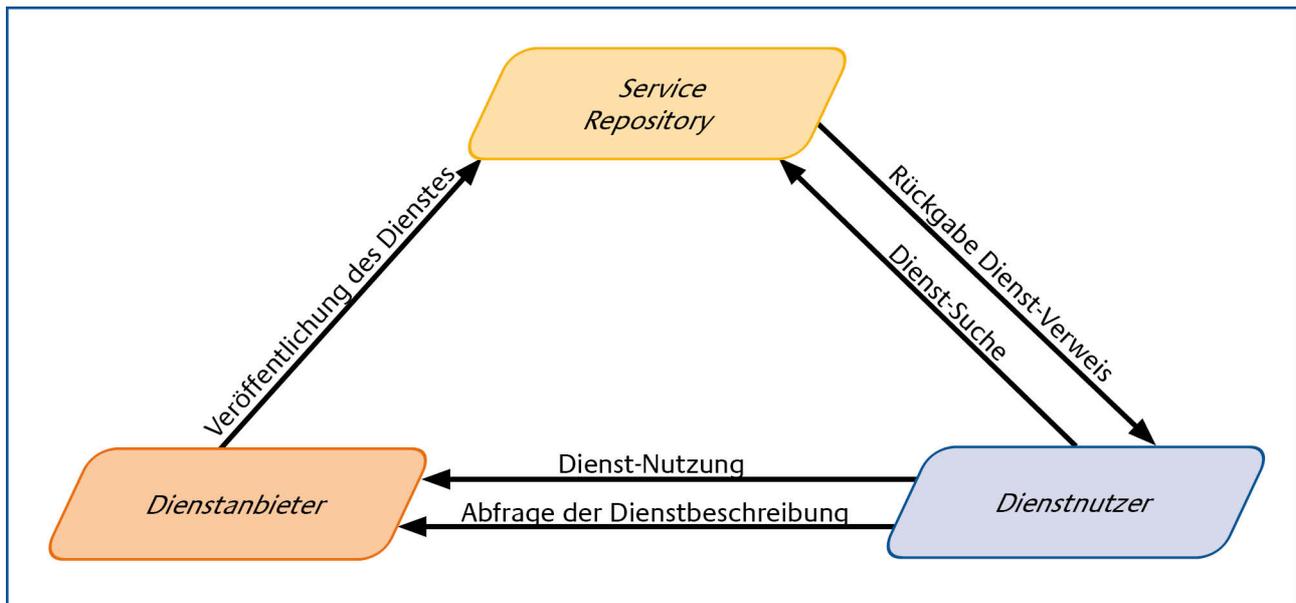


Abbildung 5: Bereitstellung und Aufruf eines Dienstes in einer SOA

Die vorangegangene Untersuchung zeigt, dass sich in dienstorientierten Architekturen mit Redundanz, Skalierbarkeit, Autonomie, Zuverlässigkeit und Virtualisierung viele wichtige Prinzipien zur Sicherstellung der Verfügbarkeit wieder finden. Somit sind Service-orientierte Architekturen in besonderem Maße für einen Einsatz in einem Umfeld mit hohen Anforderungen an die Verfügbarkeit geeignet.

4 Dienst-Management

Im Rahmen des Dienst-Managements werden all jene Aspekte analysiert, die für die Implementation und Aufrechterhaltung von IT-Diensten von besonderer Relevanz sind. Hierzu gehören insbesondere die Dienstbeschreibung, die Definition von Dienstgüteparametern, die Erstellung von Dienstvereinbarungen und Dienstkatalogen sowie die Ermittlung von Dienstabhängigkeiten. Darauf basierend lassen sich Maßnahmen auf der Dienstebene umsetzen, welche die Verfügbarkeit der Dienste sicherstellen.

Auf die einzelnen Aufgaben des Dienst-Managements wird nachfolgend näher eingegangen.

Dienstbeschreibung

Eine Grundvoraussetzung für ein effizientes Dienst-Management bildet die exakte Beschreibung des zu realisierenden Dienstes. Hierzu gehören:

- **Die Darstellung der Funktionalität des Dienstes**, d. h. zum einen die Festlegung der durch den Dienst zu erbringenden Leistung (Nutzungsfunktionalität), zum anderen die zur Konfiguration oder Überwachung des Dienstes erforderlichen Funktionen (Management-Funktionalität).
- **Die Beschreibung des Dienstzugangspunktes**, auch Service Access Point genannt, an dem der Dienst dem Dienstanutzer durch den Dienstanbieter zur Verfügung gestellt wird.
- **Die Definition der Schnittstelle**, über die dem Dienstanutzer die Management-Funktionalität ermöglicht wird (Management-Dienstzugangspunkt).

Definition von Dienstgüteparametern

Mit Hilfe der Dienstgüteparameter werden die qualitativen Eigenschaften eines Dienstes beschrieben. Sie geben die Anforderungen vor, die der Anbieter eines Dienstes ihm Rahmen seiner Leistungserbringung zu erfüllen hat.

Erstellung von Dienstvereinbarungen (Service Level Agreements)

Mittels vertraglicher Vereinbarungen zwischen Dienstanutzer und Dienstanbieter werden die Rahmenbedingungen zur Dienstleistung festgelegt. Hierzu gehören z. B. die Dienstfunktionalitäten und die Dienstgüteparameter. Auf SLAs wird im Abschnitt 6 näher eingegangen.

Ermittlung von Dienstabhängigkeiten

Wie schon in Abschnitt 2 dargestellt, ist die Kenntnis der bestehenden horizontalen und vertikalen Abhängigkeiten für den Dienstanbieter von großer Bedeutung, um geeignete Maßnahmen zur Sicherstellung der Verfügbarkeit des zu erbringenden Dienstes ergreifen zu können. Die Ermittlung von Dienstabhängigkeiten bildet somit eine wesentliche Zielrichtung eines effizienten Dienstmanagements.

Nachfolgend werden in der Abbildung 6 die zentralen Aufgaben des Dienst-Managements in einem Schaubild verdeutlicht.

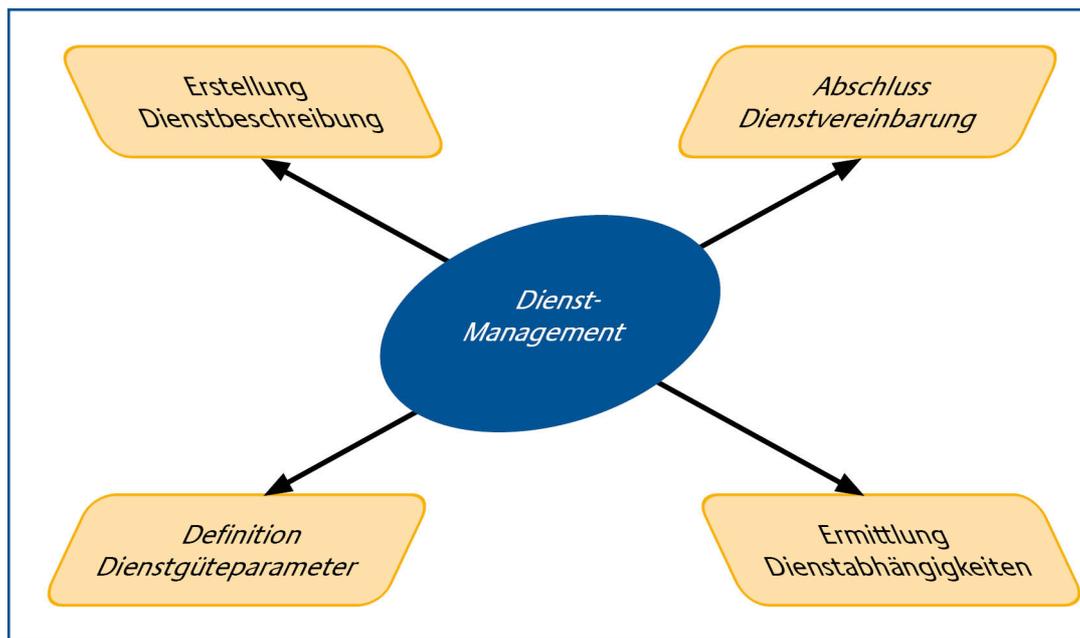


Abbildung 6: Aufgaben des Dienst-Managements

Einen weiteren bedeutenden Aufgabenbereich des Dienst-Managements in einem Hochverfügbarkeitsumfeld bildet zudem die Implementation von Maßnahmen, welche die Verfügbarkeit von Diensten sicherstellen. Hierzu gehören Dienstpriorisierung, Dienstredundanz und die präventiven Maßnahmen für kritische Geschäftsprozesse, auf die nachfolgend näher eingegangen wird.

4.1 Dienstpriorisierung

Eine Management-Maßnahme mit Auswirkungen auf die Verfügbarkeit von Diensten ist die Dienstpriorisierung. Diese ermöglicht es, zeitkritischen Diensten den Vorzug gegenüber anderen Diensten zu geben.

Hierbei handelt es sich um eine sehr effektive und kostengünstige Maßnahme, die es dem Dienstanbieter ermöglicht, den Anforderungen hinsichtlich der vereinbarten Dienstgüteparameter gegenüber dem Dienstnehmer gerecht zu werden. Im Fall von Lastschwankungen oder im Notbetrieb werden hierbei den Diensten mit höherer Priorität mehr Ressourcen als den anderen Diensten zur Verfügung gestellt, um zumindest den Betrieb der wichtigeren Dienste aufrecht-erhalten zu können. Damit dies bei den niedriger priorisierten Diensten nicht zu einer völligen Einstellung ihrer Funktionalität führt, können zusätzlich Verkehrs-Management-Maßnahmen, wie z. B. die Anpassung eines Wartespeichers an das zu erwartende Datenverkehrsaufkommen oder der Einsatz von Load-Balancern zur Lastverteilung, ergriffen werden. Hierdurch wird diesen Diensten ein Mindestmaß an Ressourcen zusichert.

Die Dienstpriorisierung als effektives Mittel zur Sicherstellung der Verfügbarkeit greift ebenso bei einem Systemausfall, indem die Dienste beim Wiederanlaufen entsprechend der zuvor festgelegten Prioritäten gestartet werden.

Eine Voraussetzung für die Priorisierung von Diensten im Rahmen von Planung, Umsetzung und Betrieb in einem HV-Umfeld ist die quantitative Bewertung aller Dienste anhand einheitlicher Kriterien, wie z. B. der Schadenshöhe, wenn der Dienst nicht zur Verfügung steht. Im Anschluss muss eine Gruppierung im Hinblick auf Bedeutung für die abhängigen Geschäftsprozesse erfolgen. Zudem ist die Kenntnis über die bestehenden Dienstabhängigkeiten (siehe Abschnitt Fehler:

Referenz nicht gefunden) von besonderer Bedeutung, um diese bei einem notwendigen Wiederanlaufen nach Systemausfall berücksichtigen zu können.

4.2 Dienstredundanz

Bei den Redundanzverfahren handelt es sich um ein weiteres adäquates Mittel zur Sicherstellung der Verfügbarkeit. Speziell auf der Dienstebene können Verfahren zur strukturellen und geografischen Redundanz eingesetzt werden.

4.2.1 Strukturelle Redundanz

Unter einer strukturellen Redundanz auf der Dienstebene ist das mehrfache Vorhandensein von funktionsgleichen Diensten zu verstehen, die der gleichen Aufgabenerfüllung dienen und über das höhere Maß an Fehlertoleranz keinen weiteren Mehrwert bezüglich der Funktionalität bieten. Sie sind somit für den Normalbetrieb entbehrlich und wären bei einer vorausgesetzten Fehlerfreiheit nicht notwendig. Strukturelle Redundanzen lassen sich hinsichtlich ihrer Aktivierungsart weiter unterscheiden.

Bei der dynamischen strukturellen Redundanz werden die Reserve-Dienste erst im Fehlerfall aktiviert. Ein Beispiel hierfür stellt die Bereitstellung funktionsgleicher und noch nicht eingebundener Dienste (E-Mail-Dienste, DNS-Dienste, Druck-Dienste etc.) in einem Service-Repository bei den dienstorientierten Architekturen dar (siehe auch Abschnitt 3).

Im Gegensatz dazu bezeichnet die statische strukturelle Redundanz das Vorhandensein von Betriebsmitteln, die bereits während des regulären Betriebs aktiv sind und zur Erfüllung der geforderten Funktionalität beitragen. In einer dienstorientierten Architektur lässt sich dies mithilfe des Enterprise-Service-Bus (siehe Abschnitt 3.10) realisieren, indem beispielsweise zwei gleichartige E-Mail-Dienste durch dessen Load-Balancing-Funktionalität parallel ihre Dienste erbringen können (siehe auch Abschnitt 3).

4.2.2 Geografische Redundanz

Bei der geografischen Redundanz handelt es sich um eine Form der strukturellen Redundanz, bei der die redundanten Dienste räumlich ausreichend weit voneinander erbracht werden. Hiermit tritt man Gefahren entgegen, die mit extrem geringer Wahrscheinlichkeit an beiden Orten gleichzeitig auftreten. Für Unternehmen mit unterschiedlichen Standorten bietet es sich an, die damit ohnehin vorhandene strukturelle Redundanz durch eine Verteilung der Diensterbringung auf die unterschiedlichen Standorte um die geografische Redundanz zu erweitern.

Die geografische Redundanz eignet sich insbesondere zur Vorsorge gegen eine vom Menschen oder der Natur ausgehende lokale Bedrohung. Eine Herausforderung unter dem Gesichtspunkt der Hochverfügbarkeit stellt hierbei die Aktualität des zur Diensterbringung benötigten Datenmaterials dar, das bei dem Ausfall eines Standortes am anderen Standort benötigt wird. Um dies zu realisieren, muss in möglichst kurzen Abständen ein regelmäßiger Datenabgleich erfolgen.

4.3 Präventive Maßnahmen für kritische Geschäftsprozesse

Durch eine wirksame Prävention lassen sich sowohl im Vorfeld als auch während der Dienstleistungserbringung Risiken für die Verfügbarkeit vermeiden oder zumindest die Auswirkungen (Ausfallzeiten) einer Nichtverfügbarkeit bei kritischen Geschäftsprozessen reduzieren. Bedeutende präventive Maßnahmen sind nachfolgend angeführt.

4.3.1 Risikoanalyse

Ein wichtiges Mittel zur Prävention stellt die Risikoanalyse dar. Durch die Erstellung eines Risikoprofils werden hierbei vorhandene Schwachstellen, die Einfluss auf die Verfügbarkeit haben können, aufgedeckt. Somit wird der Dienstbringer in die Lage versetzt, bereits während der Konzeption der Dienstleistung geeignete Strategien entwickeln zu können, die hinsichtlich der Verfügbarkeit der von ihm zugesicherten Leistung zu einer Risikovermeidung oder zumindest zu einer Verringerung des Risikos beitragen.

4.3.2 Monitoring

Durch eine permanente Überwachung der Dienste können rechtzeitig Gegenmaßnahmen ergriffen werden, falls sich die für die Bewertung der Dienstgüte festgelegten Parameter kritischen Schwellwerten nähern. Details hierzu finden sich in Abschnitt 6.2.2 und im Beitrag „Überwachung“ in diesem Kompendium. Dabei ist zu berücksichtigen, dass auf den unterschiedlichen Ebenen des Service-Modells (siehe Abschnitt 2.3.1 und 2.3.2) jeweils spezifische Kennwerte zu überwachen sind. Die Überwachungsaktivitäten sollten Bestandteil des SLAs sein und neben den QoS-Parametern, die Messverfahren, Messstellen und Messreports umfassen. Bei Kenntnis der real existierenden Service-Architektur ist eine Abbildung von dienstorientierten Problemmeldungen auf komponentenorientierte Events und umgekehrt möglich.

4.3.3 Wartung

Die regelmäßige Wartung der zur Dienstleistungserbringung eingesetzten einzelnen Dienste leistet einen weiteren Beitrag zu einem verringerten Ausfallrisiko. Hierzu zählt beispielsweise das Einspielen von Patches im Fall von bekannt gewordenen Schwachstellen und Sicherheitslücken eines Dienstes. Zudem können Fehler in der Funktionalität eines Dienstes durch Updates beseitigt und die Zuverlässigkeit erhöht werden.

4.3.4 Kapazitätsplanung

Durch eine Anpassung der Kapazitäten kann der Dienstanbieter rechtzeitig auf veränderte Rahmenbedingungen reagieren und Risiken hinsichtlich der Verfügbarkeit vermeiden.

4.3.5 Notfallplanung

Durch eine effektive Notfallplanung wird der Dienstanbieter in die Lage versetzt, im Fall einer Störung bei der Dienstleistungserbringung umgehend reagieren und geeignete Gegenmaßnahmen ergreifen zu können. Dies führt zu einer Reduzierung des durch die Nichtverfügbarkeit zu erwartenden Schadens. Wege für eine adäquate und effiziente Reaktion bei Notfällen und die schnelle Wiederaufnahme wichtiger Geschäftsprozesse werden beispielsweise im BSI-Standard 100-4 „Notfallmanagement“ aufgezeigt.

5 Dienstfindung in dynamischen Netzen

Die Dienstfindung stellt ein wichtiges Element in sich dynamisch verändernden Netzwerkumgebungen dar, welche besonders hohe Anforderungen an Skalierbarkeit und Robustheit stellen. Sie vermittelt die Nachfrager und Anbieter von Middleware-Diensten. Die Dienstfindung ermöglicht das Auffinden von transparent verteilten Middleware-Diensten, auch wenn diese erst zur Laufzeit hinzutreten und nur temporär zur Verfügung stehen.

Realisiert wird die Dienstfindung mithilfe von Dienstfindungsprotokollen. Diese definieren die Regeln, die für die Vermittlung von Diensten zwischen Dienstanbieter und Dienstanutzer gelten. Zu den Dienstfindungsprotokollen zählen Universal Plug and Play, das Service Location Protocol und Jini, die nachfolgend vorgestellt werden. Eine detaillierte Darstellung der Architektur und Vorgehensweise wird hierbei am Beispiel Jini vorgenommen.

5.1 Universal Plug and Play (UPnP)

Universal Plug and Play ermöglicht die Ad-hoc-Kommunikation von Geräten und Diensten, indem es Peer-To-Peer-Mechanismen zur automatischen Konfiguration bereitstellt. Hierbei existiert keine zentrale Anlaufstelle, bei der die angebotenen Dienste registriert werden. UPnP-Geräte und Dienste können mit anderen UPnP-Geräten und Diensten innerhalb eines IP-basierten Netzwerkes in Verbindung treten. Der Informationsaustausch erfolgt über die Protokolle IP, TCP, UDP und HTTP. Universal Plug and Play ist unabhängig von einer Programmiersprache. Die Beschreibung der Dienste wird mittels XML vorgenommen.

5.2 Service Location Protocol (SLP)

Das Service Location Protocol ist ein auf UDP und TCP basierendes Dienstfindungsprotokoll für IP-basierte lokale Netze. Es handelt sich hierbei um ein Standard-Protokoll der Internet Engineering Task Force (IETF). Charakteristisch für das Service Location Protocol ist die Klassifizierung verschiedener Agententypen.

Der Service-Agent bietet hierbei seine Dienste in einem Netzwerk an. Um diese eindeutig zuordnen zu können, werden die Dienste dabei durch einen Service URL repräsentiert. Unter dem Agententyp User Agent versteht man aus SLP-Sicht den Nutzer eines oder mehrerer Dienste. In größeren Netzwerken wird zudem ein Directory-Agent eingesetzt, der alle im Netzwerk verfügbaren Dienste in einer Datenbank zusammenfasst. Meldet sich ein neuer Service-Agent im Netzwerk an, registriert dieser seine Dienste beim Directory-Agent. Ein User-Agent wiederum stellt seine Anfrage bezüglich eines Dienstes dann an den ihm bekannten Directory-Agent, um von diesem die gewünschten Dienstinformationen zu erhalten.

5.3 Java Intelligent Network Interface (Jini)

Jini stellt eine Erweiterung der Programmiersprache Java dar. Es handelt sich um einen zentralisierten Dienstfindungsmechanismus. Hierbei sollen sich Geräte selbstständig in einem Netz anmelden und ihre Dienste automatisch ohne Installationsaufwand dem potenziellen Nutzer zur Verfügung stellen.

In einer Jini-Architektur werden zur Kommunikation drei Komponenten benötigt. Zum einen ist dies der Dienstanbieter, der anderen Teilnehmern im Netz seine Dienste zur Verfügung stellt. Zum anderen der Dienstnutzer, der die angebotenen Dienste in dem entsprechenden Netzwerk sucht und verwenden will. Um dies realisieren zu können, werden die Teilnehmer mittels des sogenannten Lookup Service (LUS), der auf einem Lookup-Server installiert ist, miteinander bekannt gemacht. Der LUS fungiert hierbei als zentraler Aufbewahrungsort für die Dienste. Die Einträge im Lookup Service sind Java Objekte. Sie bestehen jeweils aus einem Dienst-Proxy, der zur Implementierung eines öffentlichen Interfaces dient und einigen zusätzlichen Attributen.

In der nachfolgenden Abbildung 7 wird die Jini-Architektur beispielhaft dargestellt und anschließend der Dienstfindungsmechanismus näher erläutert.

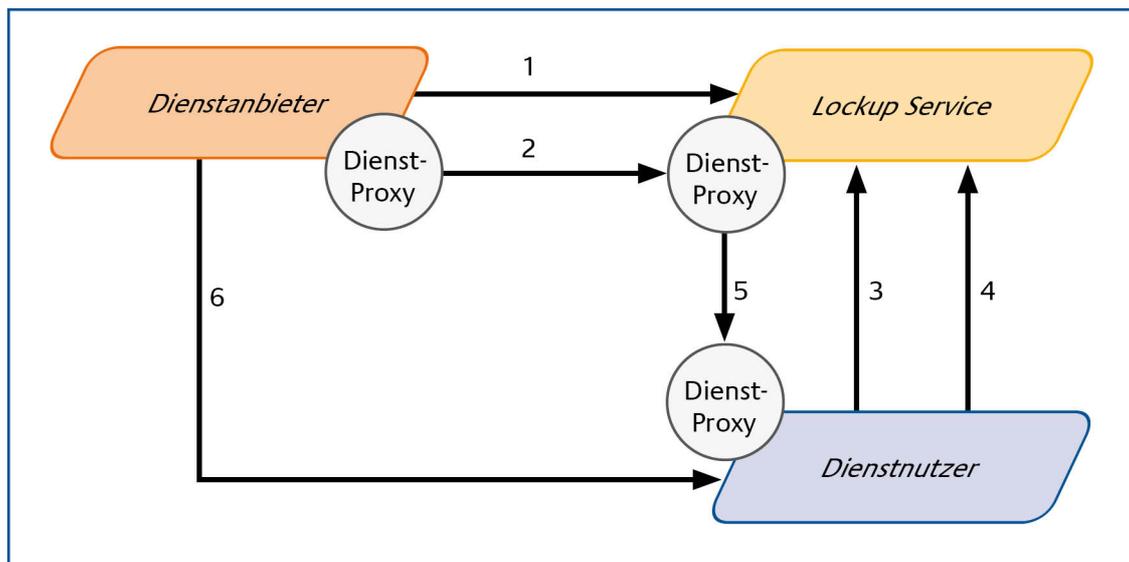


Abbildung 7: Dienstfindung mit Jini

- Um seinen Dienst anderen Teilnehmern zur Verfügung stellen zu können, meldet sich der Dienstanbieter mittels des „Discovery“ genannten Protokolls durch Aussenden von Anfragen bei einem oder mehreren Lookup Services an. Hierzu kann er drei verschiedene Methoden nutzen: zum einen ist es möglich, das Multicast-Request-Protokoll einzusetzen, mit dem er Lookup Services ansprechen kann, ohne dass ihre IP-Adresse bekannt sein muss. Kennt er die IP-Adresse des gesuchten LUS, bietet sich der Einsatz des Unicast-Discovery-Protokolls an. Eine weitere Methode ist das Multicast-Announcement-Protokoll. Dies ermöglicht es einem Lookup Service, sich aktiv bei Dienstanbietern bekannt zu machen, die sich anschließend beim LUS anmelden können.
- Nach erfolgreicher Anmeldung des Dienstanbieters erfolgt der so genannte „Join“. Hierbei wird der angebotene Dienst in den Lookup Service eingetragen. Als Informationen erhält der LUS eine ServiceID, weitere Attribute zur Dienstbeschreibung sowie den Dienst-Proxy zur Implementierung eines öffentlichen Interfaces, der von einem potenziellen Dienstnutzer später heruntergeladen werden kann.
- Auf Seiten des Dienstnutzers wird ebenfalls mithilfe des „Discovery“ nach verfügbaren Lookup Services gesucht.
- Mittels des „Lookup“ sucht der Dienstnutzer auf dem gefundenen LUS nach dem von ihm gewünschten Dienst

- Wenn der Dienstanutzer seine Suche erfolgreich abgeschlossen hat, lädt er den Dienst-Proxy des von ihm gesuchten Dienstes herunter. Dieser fungiert nachfolgend als eine Art Treiber.
- Nach der Installation des Dienst-Proxies wird der LUS zur weiteren Kommunikation nicht mehr benötigt. Der Dienstanutzer kann jetzt direkt auf den Dienst des Diensteanbieters zugreifen und diesen benutzen.

Gerade in verteilten Systemen muss mit einer eingeschränkten Verfügbarkeit der Netzwerkverbindungen oder einzelner Dienste gerechnet werden. Aufgrund dessen ist es für den Dienstanutzer besonders wichtig zu wissen, welcher Dienst ihm momentan tatsächlich zur Verfügung steht. Dies wird bei Jini mittels eines so genannten „Lease“ realisiert, welches jede Referenz zu einer Netzressource zeitlich begrenzt. So muss beispielsweise die Bereitschaft zur Dienstleistung oder das Nutzungsinteresse an einem Dienst periodisch erneuert werden. Geschieht dies nicht, wird der entsprechende Eintrag im LUS gelöscht.

6 Attribute und Parameter für die Dienstbewertung

Neben der reinen Verfügbarkeit eines Dienstes als zeitlicher Anteil eines vereinbarten Servicezeitraums, in dem der Dienst dem Nutzer zur Verfügung stehen soll, spielt die auch die bei der Erbringung erzielte Qualität des Dienstes eine entscheidende Rolle. Unter der Verfügbarkeit eines Dienstes ist somit immer die Verfügbarkeit mit vorgegebener Qualität zu verstehen.

6.1 Dienstgüte/Quality of Service

Ein wesentliches Merkmal eines Dienstes ist die Dienstgüte, auch Quality of Service (QoS) genannt. Mit ihr werden die Eigenschaften hinsichtlich der Qualität eines durch den Dienstanbieter zu realisierenden Dienstes quantitativ bestimmt. Neben der zeitlichen Verfügbarkeit eines Dienstes existieren weitere Attribute, die eine Bewertung und Quantifizierung der Dienstgüte im Sinne einer Verfügbarkeit in einem anhand des Service Level Agreements definierten Leistungsumfang ermöglichen:

Verfügbarkeit

Mit der Verfügbarkeit wird das Maß angegeben, in dem der Dienst dem Dienstanutzer innerhalb eines vereinbarten Zeitraums tatsächlich zur Verfügung steht. In der Regel wird die Verfügbarkeit in Prozent angegeben. Im Zusammenhang mit der Verfügbarkeit stehen auch die Angaben zur Reaktivierungs- und Bereitstellungszeit:

- **Reaktivierungszeit**

Die Reaktivierungszeit gibt die Dauer an, die der Dienstanbieter benötigt, um nach einem Ausfall den von ihm zu erbringenden Dienst wieder in der erforderlichen Qualität anbieten zu können.

- **Bereitstellungszeit**

Die Bereitstellungszeit gibt die Dauer an, die ein Dienstleister benötigt, um für den Dienstnehmer einen neuen Dienst in die vorhandene Infrastruktur zu integrieren.

Darüber hinaus können beispielsweise für den Netzwerk-Bereich weitere Attribute aufgenommen werden:

Bandbreite (Datenübertragungsrate)

Die Bandbreite gibt die Datenmenge an, die innerhalb eines bestimmten Zeitintervalls übertragen werden kann. Diese wird in Bit pro Sekunden angegeben. Jeder Dienst benötigt hierbei eine definierte Bandbreite, wenn dieser Informationen in Echtzeit übertragen soll.

Die Bandbreite einer Netzwerkverbindung kann zum Beispiel durch das Senden von ICMP-ECHO-Paketen und den zurück gelieferten ICMP-REPLY-Paketen berechnet werden. Für die Messung der Bandbreite existieren zudem viele Softwarelösungen.

Latenzzeit

Die Latenzzeit gibt die zeitliche Verzögerung an, die ein Dienst benötigt, um auf eine Anforderung mit einer spezifizierten Reaktion zu antworten bzw. die Summe aller Verzögerungen während einer Übertragung in einem Netzwerk.

Die Netzwerklatenz lässt sich unkompliziert durch die Absendung des PING-Befehls messen. Dieser liefert als Rückgabewert die gesuchte Latenzzeit in Millisekunden.

Zur Messung der Latenzzeit eines Dienstes könnte für diesen im Vorfeld im Rahmen eines Service Level Agreements eine Referenzanfrage definiert werden. Gemessen wird dann die Zeit, die der Dienst benötigt, um auf diese definierte Anfrage zu reagieren. Um die Vergleichbarkeit der Messergebnisse weiter zu erhöhen, könnte zusätzlich noch die Anzahl gleichzeitiger Nutzer des Dienstes während der Messung vorgeschrieben werden.

Fehlerrate

Die Fehlerrate gibt das Verhältnis fehlerhaft oder nicht übertragener Daten im Verhältnis zur Anzahl der insgesamt übertragenen Daten an. Je nach Dateneinheit spricht man von Bit-, Byte- oder Block-Fehlerrate. Bei der Fehlerrate handelt es sich um die Angabe einer relativen Häufigkeit, die innerhalb einer bestimmten endlichen Messzeit ermittelt wurde.

Für eine Messung der Fehlerrate kann im Rahmen einer Ende-zu-Ende-Messung ein Sender eingesetzt werden, der ein definiertes Prüfbitmuster erzeugt. Dieses wird an einen Empfänger geschickt, der das erhaltene Muster auswertet sowie das Ergebnis anzeigt. Alternativ kann auch eine Schleife implementiert werden, mit deren Hilfe das Prüfbitmuster wieder an den Sender zurückgeschickt wird. Sender und Empfänger sind in diesem Fall eine Einheit.

6.2 Grundlagen für Service Level Agreements

Wie bereits im Abschnitt 6.1 angeführt, werden in einem Service Level Agreement zwischen Dienstnehmer und Dienstanbieter konkrete vertragliche Vereinbarungen über die Art, den Umfang und die dabei zu gewährleistende Güte der Dienstleistung getroffen. Auf die wichtigsten Bestandteile eines SLAs unter dem Aspekt der Verfügbarkeit wird nachfolgend näher eingegangen.

6.2.1 Dienstgüteparameter

Ein wesentliches Merkmal des Service Level Agreements stellt die vertragliche Festlegung der Qualität des zu erbringenden Dienstes dar. Hierzu gehen die in Abschnitt 6.1 dargestellten Attribute der Dienstgüte als QoS-Parameter in die Vereinbarung zwischen Dienstanbieter und Dienstnehmer ein, indem für diese einzuhaltende Mindestwerte definiert werden.

Um die Qualität der Dienstleistung und der Dienstleistungsvereinbarung während der Nutzung durch den Dienstnehmer bewerten zu können, müssen die tatsächlich vom Dienstanbieter erzielten IST-Werte dieser Dienstgüteparameter im Rahmen des Monitorings erfasst werden können. Hierzu sollte in einem Service Level Agreement eine geeignete Messmetrik vereinbart werden.

6.2.2 Parameter für das Monitoring

Zusätzlich werden als weiterer Bestandteil von Service Level Agreements zwischen Dienstnehmer und Dienstleister Schwellwerte für die Dienstgüteparameter vereinbart, die im Rahmen des Monitorings überwacht werden müssen. Hierzu zählen beispielsweise die Verfügbarkeit, Latenzzeiten, Bandbreiten etc. (siehe Abschnitt 6.1). Im Fall von Über- oder Unterschreitung dieser Parameter werden vertraglich definierte Eskalationsmechanismen und Gegenmaßnahmen initiiert. Ziel ist es, die Ursache dieser Störung schnellstmöglich zu beseitigen und die Güte der Dienstleistung wieder auf das zugesicherte Niveau zu heben. Bei einer schuldhaften Verletzung der Schwellwerte durch den Dienstanbieter kann dieser zu einer Vertragsstrafe herangezogen werden.

Im Beitrag „Überwachung“ des HV-Kompendiums finden sich viele zum Monitoring weiterführende Informationen (z. B. hinsichtlich der Messung und Überwachung der Dienstgüte).

Anhang: Verzeichnisse

Abkürzungsverzeichnis

Ein komplettes Verzeichnis hierzu findet sich in Band AH, Kapitel 5

Glossar

Ein komplettes Verzeichnis hierzu findet sich in Band AH, Kapitel 6

Literaturverzeichnis

Ein komplettes Verzeichnis hierzu findet sich in Band AH, Kapitel 7