

The background of the page is a light cream color with a decorative pattern of thin, wavy teal lines that create a sense of movement and depth, particularly on the right and bottom sides.

**software** AG

# Glossar

Alfabet 10.15

---

Dokumentationsversion Alfabet 10.15.1

Urheberrechtlich geschützt © 2013 - 23 Software AG, Darmstadt, Deutschland und/oder Software AG USA Inc., Reston VA, USA und/oder ihre Tochtergesellschaften und/oder ihre Lizenzgeber.

Der Name Software AG und die Namen der Software AG Produkte sind Marken der Software AG und/oder Software AG USA Inc., einer ihrer Tochtergesellschaften oder ihrer Lizenzgeber. Namen anderer Gesellschaften oder Produkte können Marken ihrer jeweiligen Schutzrechtsinhaber sein. Genaue Informationen über die geschützten Marken und Patente der Software AG und ihrer Tochtergesellschaften sind veröffentlicht unter <http://softwareag.com/licenses>.

Die Nutzung dieser Software unterliegt den Lizenzbedingungen der Software AG. Diese Bedingungen sind Bestandteil der Produktdokumentation und befinden sich unter <http://softwareag.com/licenses> und/oder im Wurzelverzeichnis des lizenzierten Produkts.

Diese Software kann Teile von Software-Produkten Dritter enthalten. Urheberrechtshinweise, Lizenzbestimmungen sowie zusätzliche Rechte und Einschränkungen dieser Drittprodukte können dem Abschnitt "License Texts, Copyright Notices and Disclaimers of Third Party Products" entnommen werden. Diese Dokumente enthalten den von den betreffenden Lizenzgebern oder den Lizenzen wörtlich vorgegebenen Wortlaut und werden daher in der jeweiligen Ursprungssprache wiedergegeben. Für einzelne, spezifische Lizenzbeschränkungen von Drittprodukten siehe PART E der Legal Notices abrufbar unter dem Abschnitt „License Terms and Conditions for Use of Software AG Products / Copyrights and Trademark Notices of Software AG Products“. Diese Dokumente sind Teil der Produktdokumentation, die unter <http://softwareag.com/licenses> oder im Verzeichnis der lizenzierten Produkte zu finden ist.

Die Produkte der Software AG stellen Funktionalität zur Verfügung, die für die Verarbeitung persönlicher Daten entsprechend der EU-Datenschutz-Grundverordnung (DSGVO) genutzt werden kann. Die Beschreibungen zur Nutzung dieser Funktionalität finden Sie in der Administrationsdokumentation des jeweiligen Produkts.

Dieses Glossar definiert Begriffe, die in der Onlinehilfe oder beim Arbeiten mit Alfabet öfter verwendet werden. Wenn es sich bei dem Begriff um den Namen einer Alfabet -Objektklasse handelt, dann wird zusätzlich das Symbol angezeigt, dass die Objektklasse auf der Benutzeroberfläche, zum Beispiel in Explorern, repräsentiert.

Die Begriffe sind alphabetisch geordnet.

## A

### Abfrage

Siehe [Alfabet-Abfrage](#).

### Ad-hoc-Meilenstein

Ein Ad-hoc-Meilenstein dient der Planung und Verfolgung des Fortschritts und der Fertigstellung von [Projekten](#) oder [Enterprise-Releases](#). Ad-hoc-Meilensteine schaffen umfassende Flexibilität beim Erzeugen von [Meilensteinen](#) und lassen sich unabhängig von einer konfigurierten [Meilensteinvorlage](#) erzeugen. Anders als ein Meilenstein, der auf einer Meilensteinvorlage basiert, wird ein Ad-hoc-Meilenstein für ein bestimmtes Projekt oder Enterprise-Release erzeugt und kann nicht wiederverwendet werden.

### Aktivität

Eine Aktivität ist ein Element in einem [Service-Diagramm](#), das eine Aufgabe darstellt, die einen [Business-Prozess](#) auf granulärer Ebene repräsentiert. Eine Aktivität kann direkt mit einem Agenten verbunden werden, der die Aufgabe ausführt. Das kann zum Beispiel eine [Rolle](#) sein, die den Business-Prozess steuert oder eine [Applikation](#), die den angeforderten [Business-Service](#) zur Unterstützung der Aufgabe bereitstellt.

Eine Aktivität hat üblicherweise eine Vorgänger- und Nachfolge-Aktivität, die mit ihr über [Sequenzflüsse](#) verbunden sind. Komplexere Verbindungen zwischen Aktivitäten sind mittels [Gateways](#) möglich.

Eine Aktivität kann auch mit einem [Ereignis](#) des Typs **Zwischenereignis** verbunden sein, das stattfindet, während die Aktivität ausgeführt wird. Ein Ereignis des Typs **Startereignis** löst üblicherweise eine Aktivität aus. Eine Aktivität kann mit einem Ereignis des Typs **Endereignis** abschließen.

### Aktivitätsmonitor

Ein Aktivitätsmonitor ist ein [Monitor](#), der die in einem Verteiler eingetragenen Anwender über Änderungen, die an [Objekten](#) einer bestimmten [Objektklasse](#) vorgenommen wurden, informiert. Der Eigentümer des Monitors definiert eine Reihe von [Eigenschaften](#), die überwacht werden sollen, sowie die Anwender, die benachrichtigt werden sollen, sobald der Monitor ausgelöst wird. Diese Anwender sowie der Monitoreigentümer werden per E-Mail informiert, sobald eine der festgelegten Eigenschaften für ein beliebiges Objekt in der Klasse geändert wurde. Die Überwachung erfolgt über einen bestimmten Zeitraum in regelmäßigen Intervallen.

## Alfabet-Abfrage

Eine Alfabet-Abfrage dient der effizienten Suche nach und dem Abrufen von Informationen in Alfabet auf Basis von Objektattributen und Beziehungen zwischen Objekten. Alfabet-Abfragen basieren auf einer klassenorientierten Abfragesprache und können in diversen Zusammenhängen in Alfabet konfiguriert werden, wie zum Beispiel Volltextsuchen, konfigurierte Berichte, [Berechnungsregeln](#), [Benachrichtigungsmonitore](#) und [Konsistenzmonitore](#). Normalerweise wird eine Alfabet-Abfrage für eine [Basisklasse](#) definiert. Sie kann jedoch auch über mehrere [Objektclassen](#) hinweg definiert sein.

Für viele konfigurierbare Alfabet -Funktionalitäten können alternativ zur Alfabet-Abfragesprache auch native SQL-Abfragen spezifiziert werden.

## allgemeine Referenzdaten

Allgemeine Referenzdaten sind Objekte, die eine Klassifizierung darstellen, und sind eine Alternative für [Aufzählungen](#), um benutzerdefinierte Daten zu erfassen. Typische Klassifikationen, die bereits in Alfabet vorhanden sind, sind die Referenzdatendefinitionstypen [Verbindungstypen](#), [Verbindungsmethoden](#), [Verbindungsfrequenzen](#) und [Verbindungsdatenformate](#).

## Analyse-Dashboard

Ein Analyse-Dashboard ist eine Ad-hoc-Visualisierung von Daten mit hohem Informationsgehalt, die von einem Anwender auf der Grundlage des integrierten Drittanbieter-DevExpress®-Dashboards entwickelt worden ist. Ein oder mehrere Dashboard-Elemente wie zum Beispiel Diagramme, Punktdiagramme, Raster, Karten, Messdiagramme, Drehpunkte, Bereichszuordnungen, Baumstrukturen usw. können dem Analyse-Dashboard hinzugefügt werden, und Filteroptionen wie Kombinationslistenfelder, Listenfelder und Strukturansichten können genutzt werden. Für das Analyse-Dashboard ist ein konfigurierter Datenanbieter angegeben, um die Daten abzurufen, die im Analyse-Dashboard angezeigt werden sollen. Analyse-Dashboards können für andere Anwender freigegeben werden.

## Anbieter

Ein Anbieter stellt [Anbieterprodukte](#) bereit. Ein Anbieterprodukt wird von einem einzigen Anbieter vermarktet und verkauft. Abhängig von der Konfiguration Ihrer Alfabet -Lösung können Anbieter entweder direkt in Alfabet erzeugt werden oder aus dem Technopedia®-Repository importiert werden. Der Import erfolgt während des Imports von Technopedia-Produkten entweder als Anbieterprodukt oder als [Komponente](#) in Alfabet. Für die Objektklasse **Anbieter** können [Objektklassenstereotype](#) konfiguriert werden.

## Anbietergruppe

Eine Anbietergruppe ist ein Container zur logischen Strukturierung von [Anbietern](#). Es gibt verschiedene Möglichkeiten, Anbieter logisch zu strukturieren. Daher kann jeder Anbieter mit mehreren Anbietergruppen verbunden werden.

## Anbieterprodukt

Ein Anbieterprodukt ist eine Ware oder ein Service, die/der von einem [Anbieter](#) für das Unternehmen bereitgestellt wird. Ein Anbieterprodukt kann einer oder mehreren technischen [Komponenten](#) zugeordnet werden. Das Anbieterprodukt "Oracle RDBMS" kann z.B. in die Komponenten "Oracle 11i Server", "Oracle 11i Client" und "Oracle 11i OCL" unterteilt werden. Ferner kann eine Komponente auf ein Anbieterprodukt verweisen, wodurch angegeben wird, dass dieses aus dem Anbieterprodukt abgeleitet wurde.

Abhängig von der Konfiguration Ihrer Alfabet -Lösung können Anbieterprodukte entweder direkt in Alfabet erzeugt werden oder aus dem Technopedia®-Repository importiert werden. Anbieterprodukte können nur aus Technopedia®-Produktkategorien importiert werden, die sich auf der untersten Hierarchieebene der Produktkategorien befinden.

Ein Anbieterprodukt wird von einem einzigen Anbieter vermarktet und verkauft. Ein Anbieterprodukt kann nur einer [Anbieterproduktkategorie](#) zugeordnet werden.

Für die Objektklasse **Anbieterprodukt** können [Objektklassenstereotype](#) konfiguriert werden.

## Anbieterproduktgruppe

Eine Anbieterproduktgruppe ist ein Container zur logischen Strukturierung von Anbieterprodukten. Es gibt verschiedene Möglichkeiten, Anbieterprodukte logisch zu strukturieren. Daher kann jedes Anbieterprodukt mit mehreren Anbieterproduktgruppen verknüpft werden.

## Anbieterproduktkategorie

In einer Anbieterproduktkategorie werden [Anbieterprodukte](#) zusammengefasst und inhaltlich klassifiziert. In Anbieterproduktkategorien können Sie die Anbieterprodukte in Ihrem Unternehmen auch hierarchisch strukturieren. Ein Anbieterprodukt kann nur mit einer Anbieterproduktkategorie verbunden werden.

## Anforderung

Eine Anforderung ist eine Änderungsanfrage für die [IT-Landschaft](#) des Unternehmens. In Alfabet sind Anforderungen üblicherweise strategischer Natur oder betreffen die IT-Landschaft und müssen daher beim Projektplanungsprozess des Unternehmens berücksichtigt werden.

Üblicherweise wird eine Anforderung einer Reihe von [Architekturelementen](#), die zur Erfüllung der Anforderung geändert werden müssen, zugeordnet. Diese Informationen können verwendet werden, um den Architekturplanungsprozess zu initiieren, und um Synergien und Redundanzen zwischen den Anforderungen zu identifizieren. Sobald eine Anforderung aufgenommen wurde, kann diese einem [Projekt](#) zugeordnet und darüber umgesetzt werden. Eine Anforderung kann nur zu einem Projekt zugeordnet werden.

## Anforderungsgruppe

Eine Anforderungsgruppe ist ein Container zur logischen Strukturierung von Anforderungen. Es gibt verschiedene Möglichkeiten, Anforderungen logisch zu strukturieren. Daher kann jede Anforderung mehreren Anforderungsgruppen zugeordnet werden.

## Anwender

Ein Anwender ist eine Person, die dem Alfabet -System bekannt ist. Anwender haben Zugriffsberechtigungen und sind einem oder mehreren [Anwenderprofilen](#) zugeordnet, die den Zugriff auf die Funktionalitäten regeln. Ein Anwender kann für ein Objekt ein [autorisierter Anwender](#) sein. Authentifizierung und Autorisierung werden ebenfalls mittels der Anwenderdefinition gesteuert.

## Anwendergruppe

In einer Anwendergruppe wird eine Reihe von Anwendern zusammengefasst. Wenn eine Anwendergruppe einem [Objekt](#) zugeordnet ist, wird sie als die [autorisierte Anwendergruppe](#) des Objekts bezeichnet.

## Anwenderprofil

Anwenderprofile sind die Basis für die Anwenderverwaltung in Alfabet und dient als Einstiegspunkt beim Zugriff auf Alfabet. Alle Anwender müssen sich mit einem Anwenderprofil anmelden, das ihnen von einem Anwenderadministrator zugeordnet werden muss. Allen Anwendern, die auf Alfabet zugreifen, muss daher mindestens ein Anwenderprofil zugeordnet sein. Die Anwender können jedoch entsprechend Ihrer Verantwortlichkeiten in der Anwendergemeinschaft und im Unternehmen als Ganzes über mehrere Anwenderprofile verfügen. Ein Anwender kann während einer Sitzung jederzeit zu einem anderen verfügbaren Anwenderprofil wechseln.

In einem Anwenderprofil werden die für einen Anwender verfügbaren Alfabet -Funktionalitäten, die Sichtbarkeit und die Bearbeitbarkeit von [Objektklassen](#) und [Objektklassenattributen](#) sowie die Verfügbarkeit von zugehörigen Funktionalitäten wie [Wizards](#) und [Workflows](#) angegeben.

## Applikation

Eine Applikation ist ein voll funktionsfähiges, integriertes IT-Produkt, das Funktionalitäten für Endanwender und/oder für andere Applikationen bietet. Eine Applikation unterstützt das Unternehmen bei der Zielerreichung. Applikationen werden auf einer [Plattform](#) betrieben, die aus den für die Ausführung der Applikation erforderlichen Hardware- und Software-[Komponenten](#) besteht.

Der Begriff "Applikation" wird mit dem Begriff "Applikationsversion" synonym verwendet. Eine Applikation verfügt über einen definierten [Lebenszyklus](#) und kann auch Vorgänger- und Nachfolgeversionen haben. Dadurch werden Informationen über die Entwicklung einer bestimmten Art von [Business-Support](#) oder [Business-Service](#) bereitgestellt.

Applikationen können über [Applikationsvarianten](#) verfügen, um den Bedarf nach der Lokalisierung bestimmter Aspekte der Applikation zu berücksichtigen, wie etwa [Informationsflüsse](#), die erforderlich sind, um die Applikation in eine lokale Architekturumgebung zu integrieren. Aus Sicht der strategischen Planung sind nur die logische Struktur der Applikationen sowie die bereitgestellten Business-Services und Business-Supports von Bedeutung.

## Applikationsarchitektur

Die Applikationsarchitektur ist eine detaillierte Beschreibung einer bestimmten [Applikation](#). Die Applikationsarchitektur umfasst Folgendes:

- Die Business-Architektur der Applikation, die [Business-Supports](#) und [Business-Services](#) umfasst, die die Applikation bereitstellt.
- Die Informationsarchitektur der Applikation, die die Interaktion der Applikation mit anderen Applikationen in den [IT-Landschaften](#) des Unternehmens sowie die [Businessdaten](#), die verarbeitet oder ausgetauscht werden, umfasst.
- Die technische Architektur der Applikation, durch die die technische Plattform einschließlich deren [Plattform-Tiers](#) und [Plattform-Layers](#) definiert wird, sowie die technischen [Komponenten](#), die für die Applikation erforderlich sind.
- Die Installationsarchitektur der Applikation, durch die die technischen Installationen der Applikation in der physikalischen IT-Landschaft des Unternehmens identifiziert werden.

## Applikationsarchitektur des Unternehmens

Die Applikationsarchitektur des Unternehmens definiert die Applikationsansicht auf die Unternehmensarchitektur und wird im Hinblick auf die IT-[Applikationen](#) des Unternehmens beschrieben.

## Applikationsgruppe

Eine Applikationsgruppe dient der logischen Strukturierung von [Applikationen](#), um die IT-Architektur des Unternehmens anzuzeigen, zu analysieren und zu kommunizieren. Applikationen können auf viele verschiedene Arten logisch strukturiert werden. Daher kann jede beliebige Applikation mit mehreren Applikationsgruppen verbunden sein. Zu den üblichen Arten der Gruppierung von Applikationen zählen:

- Die [Business-Prozesse](#) auf der obersten Ebene, die durch Applikationen unterstützt werden.
- Die [Organisationen](#), die durch Applikationen unterstützt werden.
- Die Organisationen, die für das Betreiben und die Pflege der Applikationen verantwortlich sind.
- Die Technologie, die die Applikationen nutzen.
- Die Ad-hoc-Bewertungen von Segmenten der [IT-Landschaft](#).

## Applikationsvariante

Eine Applikationsvariante wird aus einer Basis[applikation](#) abgeleitet. Durch eine Applikationsvariante wird die Definition von lokalen Variationen der Applikation ermöglicht, z.B. bezüglich der [Schnittstellensysteme](#) der [Informationsflüsse](#), der [Businessdaten](#) oder der technischen [Komponenten](#). Eine Applikationsvariante ist ihrer Basisapplikation untergeordnet und wird im Explorer-Baum unterhalb der Basisapplikation angezeigt. Eine Applikationsvariante kann immer dann aktualisiert werden, wenn die Definition der Basisapplikation geändert wird.

## Applikationsversion

siehe [Applikation](#)

## Architekturelement

Ein Architekturelement ist ein Objekt im [Inventory](#), das für die Erfassung der [Ist-Architektur](#) definiert ist.

## Archivobjekt

Ein Archivobjekt ist eine Momentaufnahme eines gelöschten Alfabet -[Objekts](#). Wenn ein Alfabet -Objekt archiviert wird, wird eine ZIP-Datei erzeugt, die HTML-Dateien enthält, in denen das [Objektprofil](#) des archivierten Objekts sowie die Objektprofile der abhängigen Objekte angezeigt werden. In jedem archivierten Objektprofil wird eine Reihe vorkonfigurierter Ansichtsseiten angezeigt, wobei die Sichtbarkeit dieser Ansichten von der für die [Objektklasse](#) konfigurierten Klasseneinstellungen abhängig ist. Wenn auf einer Ansichtseite abhängige Objekte angezeigt werden, können die Anwender das abhängige Objekt in der HTML-Ansicht anklicken, um eine andere HTML-Datei zu öffnen, in der das archivierte Objektprofil des ausgewählten abhängigen Objekts angezeigt wird.

Alfabet -Objekte werden üblicherweise vom Lösungsadministrator archiviert. Die ZIP-Datei kann auf einem lokalen Datenträger gespeichert und entpackt werden. Die entsprechende HTML-Datei kann in einem Browser-Fenster angezeigt werden. Ein Archivobjekt wird für alle [Landeseinstellungen](#) erzeugt, die in Ihrem Unternehmen unterstützt werden.

## Artefakt

siehe [Objekt](#)

## Aspekt-Bewertung

Eine Aspektbewertung ist eine qualitative Beurteilung der Leistung von [Applikationen](#) oder [Komponenten](#), die einer oder mehreren Applikationsgruppen bzw. Komponentengruppen zugeordnet sind. Da zum Beispiel eine Applikation mehreren Applikationsgruppen zugeordnet sein kann, kann für verschiedene Applikationsgruppe gemäß den selben Bewertungskriterien eine unterschiedliche Beurteilung derselben Applikation erfolgen.



Beispielsweise kann die Komponente "TradeNet" für unterschiedliche Zwecke in der Unternehmensarchitektur relevant sein. Die Komponente kann in verschiedenem Maß sowohl als eine Business Intelligence-Lösung als auch als eine OLAP-Datenbank geeignet sein. Mittels einer Aspektbewertung könnten Sie Komponenten in einer Komponentengruppe auf Basis spezifischer Aspektkennzahlensystemen, wie etwa der Bedeutung für das Business oder der Nutzung, zu untersuchen. Wie herkömmliche Kennzahlensysteme umfasst auch das Aspektkennzahlensystem einen oder mehrere Kennzahltypen, die für die Bewertung von Objekten verwendet werden.

Eine Aspekt-Bewertung basiert auf einem oder mehreren [Kennzahlensystemen](#), die den Objektklassen Applikation oder Komponente als Aspektkennzahlensysteme zugeordnet sind. Wie herkömmliche Kennzahlensysteme fassen Aspektkennzahlensysteme eine oder mehrere [Kennzahltypen](#) zusammen, die für die Bewertung von Objekten verwendet werden.

Aspekt-Beertungen sind derzeit für die Klassen Applikation und Komponente verfügbar.

## Aspektportfolio

Ein Aspekt-Priorisierungsschema ist ein [Portfolio](#) in dem entweder eine dreidimensionale Grafik oder ein BCG-Quadrant-Layout angezeigt werden, in dem qualitative Informationen gemäß der [Aspektbewertung](#) und den [Aspektpriorisierungsschemata](#), die für das Aspektportfolio definiert wurden, über [Applikationen](#) oder [Komponenten](#), die einer oder mehreren bestimmten Applikations- oder Komponentengruppen zugeordnet sind, angegeben werden.

## Aspekt-Priorisierungsschema

Ein Aspekt-Priorisierungsschema ist ein [Priorisierungsschema](#), in dem [Kennzahlensysteme](#) in einer [Aspektbewertung](#) zusammengefasst sind, um die Leistung von [Applikationen](#) oder [Komponenten](#) zu bewerten, die einer oder mehreren Applikationsgruppen bzw. Komponentengruppen zugeordnet sind. Ein Aspekt-Priorisierungsschema kann ebenfalls als Achse oder Dimension implementiert werden, die in einem [Aspektportfolio](#) verwendet wird. Aspekt-Priorisierungsschemata sind derzeit für die Klassen Applikation und Komponente verfügbar.

## Aufgabe

Aufgaben werden für ein ausgewähltes [Objekt](#) definiert und einem bestimmten Anwender zugewiesen. Der Empfänger muss die für das Objekt erforderlichen Eingaben bis zu einem bestimmten Fälligkeitsdatum bereitstellen. Aufgaben können als optional oder obligatorisch definiert werden. Im Kontext der Funktionalität "Aufgaben" kann das Versenden von E-Mail-Benachrichtigungen konfiguriert werden.

## Autorisierte Anwendergruppe

Eine autorisierte Anwendergruppe ist eine [Anwendergruppe](#), die einem [Objekt](#) zugeordnet wurde. Wie der [autorisierte Anwender](#) des Objekts haben alle Anwender in der autorisierten Anwendergruppe Schreib- und Lesezugriffsberechtigungen für das Objekt. Eine Anwendergruppe kann über eine unbegrenzte Anzahl von untergeordneten Anwendergruppen verfügen.

Ein Lösungsentwickler kann die Vererbung und/oder Ausbreitung von Zugriffsberechtigungen auf autorisierte Anwendergruppen konfigurieren. Wenn die Vererbung von Zugriffsberechtigungen spezifiziert wurde, sollten alle Anwendergruppen in der Anwendergruppenhierarchie, die einer Anwendergruppe untergeordnet sind, über dieselben Zugriffsberechtigungen für dieses Objekt verfügen. Wenn die Ausbreitung von Zugriffsberechtigungen spezifiziert wurde, sollten alle Anwendergruppen in der Anwendergruppenhierarchie, die einer ausgewählten Anwendergruppe übergeordnet sind, über dieselben Zugriffsberechtigungen für dieses Objekt verfügen.

## Autorisierter Anwender

Der autorisierte Anwender ist der Alfabet -Anwender, der die Hauptverantwortung für ein [Objekt](#) trägt. Der autorisierte Anwender hat Schreib- und Lesezugriffsberechtigungen für die Objekte, deren autorisierter Anwender er ist. Der autorisierte Anwender wird häufig auch als der Eigentümer des Objekts bezeichnet und die Objekte, die zu seinem Eigentum zählen, werden als [Eigene Objekte](#) des Anwenders bezeichnet.

Der autorisierte Anwender kann einen anderen Anwender als [Stellvertreter](#) für seine eigenen Objekte definieren. Jeder Anwender, der als Mitglied einer [autorisierten Anwendergruppe](#) definiert ist, erhält automatisch eine Autorisierung für die Objekte, die der autorisierten Anwendergruppe zugeordnet sind.

Zusätzlich zu autorisierten Anwendern können Anwender und [Organisationen](#), die eine fachliche Beziehung zu einem Objekt aufweisen, über das Konzept der [Rollen](#) definiert werden.

## Autorisiertes Anwenderobjekt

Ein autorisiertes Anwenderobjekt ist ein beliebiges [Objekt](#), dem der Anwender als [autorisierter Anwender](#) zugeordnet ist.

## B

### Basis-Gefährdungspotential

Das Basis-Gefährdungspotential ist eine Bewertung von Objekten entsprechend ihrer potentiellen [Risiken](#). Alle [Risiko-Objekte](#), die einer [Risikomanagementgruppe](#) zugeordnet sind, werden mithilfe eines Risikorelevanz-Fragebogens auf Basis einer konfigurierten [Kennzahl-Lookup-Tabelle](#) hinsichtlich ihres Basis-Gefährdungspotentials bewertet. Sobald das Basis-Gefährdungspotential eines Objekts bewertet wurde, werden für dieses Objekt Risikorelevanzwerte berechnet, um zu bestimmen, ob das Basis-Gefährdungspotential eine bestimmte Schwelle überschreitet. Alle Objekte, die einen Wert oberhalb dieser Schwelle aufweisen, sollten hinsichtlich der potentiellen Schadensgefahr und der Risikowahrscheinlichkeit genauer analysiert werden.

### Basisklasse

Die Basisklasse ist die [Objektklasse](#), für die ein Anwender nach Ergebnissen sucht. Beispielsweise ist eine Basisklasse für eine [Alfabet-Abfrage](#), eine native SQL-Abfrage oder eine [Berechnungsregel](#) relevant.

## Basisobjekt

Das Basisobjekt ist das [Objekt im Inventory](#), auf dem ein [Lösungsobjekt](#) im Kontext der Lösungsplanung basiert. Beispielsweise kann eine [Lösungsapplikation](#) auf einer [Applikation](#) basieren, die Teil der [Ist-Architektur](#) eines [Projekts](#) ist. Das Basisobjekt dient als Ausgangspunkt für die Definition der Lösungsapplikation. Alle für die Basisapplikation definierten [Eigenschaften](#) und Beziehungen werden in die neue Lösungsapplikation kopiert. Die Definition der Lösungsapplikation kann anschließend im Kontext der [Lösung](#) präzisiert werden.

## Bebauungsplan

Mittels eines Bebauungsplans wird die mittelfristige [Soll-Architektur](#) beschrieben, die zur Erfüllung der langfristigen [IT-Strategie](#) des Unternehmens erforderlich ist. Ein Bebauungsplan stellt die Gesamtheit aller [Business-Supports](#) dar, die vom Unternehmen oder einer unabhängigen Organisationseinheit des Unternehmens geplant sind. Ein Bebauungsplan kann über eine oder mehrere [Bebauungsplan-Matrizen](#) verfügen, in denen die zukünftigen [taktischen Business-Supports](#) speziell für den Bebauungsplan definiert sind. Ein Bebauungsplan kann als [Blueprint](#) angegeben werden, der als Orientierungshilfe bei der Bebauungsplanung dient.

## Bebauungsplan-Matrix

Eine Bebauungsplan-Matrix ist Bestandteil eines [Bebauungsplans](#), der eine Aufteilung der [Bebauungsplanung](#) nach Zuständigkeit, Affinität oder Ähnlichkeiten in Bereiche ermöglicht. Ein Bebauungsplan kann über mehrere Bebauungsplan-Matrizen verfügen.

Eine Bebauungsplan-Matrix setzt sich aus einer Reihe von [Organisationen](#) oder [Marktprodukten](#) und einer Reihe von [Business-Prozessen](#) oder Domänen zusammen und ermöglicht eine Analyse der Ausrichtung des geplanten Business-Supports. Eine Bebauungsplan-Matrix kann eine Reihe bevorzugter Architekturelemente umfassen, die bei der Definition des [taktischen Business-Supports](#) in der [Business-Support-Matrix](#) verwendet werden sollen.

Eine [Matrixansicht](#) ermöglicht es den Anwendern, nur einen Bereich des Business-Supports anzuzeigen, der in der Business-Support-Matrix der Bebauungsplan-Matrix geplant wurde. Bebauungsplan-Matrizen können in Bebauungsplan-Ordnern organisiert werden.

## Bebauungsplanung

Bebauungsplanung ist der Prozess des Definierens von [Bebauungsplänen](#), die an der [IT-Strategie](#) ausgerichtet sind, die in [strategischen Bebauungsplan-Matrizen](#) definiert ist. Bebauungsplanung beinhaltet die Definition von [taktischem Business-Support](#) in [Bebauungsplan-Matrizen](#). Die taktischen Business-Supports, die in solchen Matrizen definiert sind, stellen den mittelfristig geplanten Support von [Business-Prozessen](#) oder Domänen dar, der von [Applikationen](#) und [ICT-Objekten](#) bereitgestellt wird und in unterschiedlichen [Organisationen](#) oder von Marktprodukten genutzt wird. Die Bebauungsplanung umfasst außerdem die Analyse und die Ausrichtung von taktischen und [operativen Business-Supports](#) im Hinblick auf den Zeitrahmen, auf die bereitgestellten [Business-Services](#) sowie in Bezug auf das Business und die technische und strategische Ausrichtung.


## Bedrohung

Eine Bedrohung verweist auf die Quelle einer bestimmten Art von [Risiko](#) für die IT des Unternehmens. Eine Bedrohung kann mit einem bestimmten [Risiko-Objekt](#) in der IT-Landschaft des Unternehmens verbunden sind, was bedeutet, dass das Risiko für das Objekt aus der Bedrohung resultiert. Zudem können [Risikominderungs-vorlagen](#) für die Bedrohung definiert werden, um ein oder mehrere [Risikominderungen](#) zu formulieren, die darauf abzielen, das Risiko, dass von der potentiellen Bedrohung ausgeht, zu vermeiden, zu reduzieren oder einzudämmen.

## Benachrichtigungsmonitor

Ein Benachrichtigungsmonitor ist ein [Monitor](#) mit dem E-Mail-Benachrichtigungen automatisch basierend auf konfigurierten [Alfabet-Abfragen](#) oder nativen SQL-Abfragen ausgelöst werden können. Die Abfragen geben die angestrebten [Objekte](#) und ihre [Objektklasseneigenschaften](#) sowie die Anwender an, die über die mittels der Abfragen gefundenen Objekte benachrichtigt werden sollen.

## Benutzerdefinierte Eigenschaft

Eine benutzerdefinierte Eigenschaft  ist eine [Objektklasseneigenschaft](#), die konfiguriert wurde, um unternehmensspezifische Daten für eine bestimmte [Objektklasse](#) zu erfassen.

Pro Objektklasse können mehrere benutzerdefinierte Eigenschaften definiert werden. Wie private Eigenschaften, können benutzerdefinierte Eigenschaften (auch als öffentliche Eigenschaften bezeichnet) aus einer Klasseneinstellung ausgeschlossen werden und somit für Anwender, die Alfabet mit den zugehörigen [Anwenderprofilen](#) aufrufen, ausgeblendet werden. Ferner können benutzerdefinierte Eigenschaften auch in die Definition eines Klassenschlüssels integriert werden. Hierzu müssen die Anwender während der Definition von Objekten in der Objektklasse für eine Reihe von Standard- und benutzerdefinierten Eigenschaften eindeutige Werte eingeben.

Damit die Anwender in der Anwendergemeinschaft Werte für die benutzerdefinierten Eigenschaften definieren können, müssen die Eigenschaften mit Dateneingabefeldern in benutzerdefinierten Editoren verbunden werden. Benutzerdefinierte Eigenschaften können angezeigt werden in: [Objektprofilen](#) und [Objekt-Cockpits](#). Ferner können benutzerdefinierte Eigenschaften genutzt werden, um nach Objekten in den Alfabet -Suchfunktionalitäten zu suchen und sie können in [Alfabet-Abfragen](#) und nativen SQL-Abfragen spezialisiert werden, die z.B. in [konfigurierten Berichten](#) genutzt werden.

## Berechnungsregel

Mittels einer Berechnungsregel werden numerische [Kennzahlen](#) für Objekte einer bestimmten [Objektklasse](#) automatisch berechnet. Zur Definition der Berechnungslogik muss der Anwender eine [Basisklasse](#) und eine zugehörige Eigenschaft des Typs `Integer`, `Real`, `Reference`, oder `ReferenceArray` definieren. Der [Kennzahltyp](#), der berechnet werden soll, muss ebenfalls der Berechnungsregel zugeordnet werden. Ein Skalierungsschema kann bei Bedarf für die Berechnungsregel definiert werden.

Durch eine Berechnungsregel kann folgendes definiert werden::

## Bericht

Ein Bericht ist eine konfigurierte Anzeige von Daten aus der Alfabet-Datenbank. Alfabet stellt eine Reihe von Standardberichten zur Verfügung, es können jedoch auch konfigurierte Berichte durch einen Lösungsentwickler erstellt werden.

## Bewertung

Eine Bewertung ist eine Reihe von Größen, die die Leistung eines [Objekts](#) in Bezug auf eine bestimmte Dimension messen. Eine Bewertung basiert auf einem [Kennzahlensystem](#), das ein Bündel von [Kennzahlen](#) umfasst. Kennzahlen sind die berechneten Ausgabewerte von vorkonfigurierten [Kennzahltypen](#).

Üblicherweise führt der [autorisierte Anwender](#) oder die [autorisierte Anwendergruppe](#) eines Objekts die Bewertung aus.

## Blueprint

Ein Blueprint ist ein [Bebauungsplan](#) oder eine [IT-Strategie](#), die/der als Richtlinie für die Planung des organisationsübergreifenden [Business-Supports](#) im Unternehmen dient. Im Blueprint-Planungsprozess können sich die Bebauungsplaner und strategischen Planer auf einen Blueprint beziehen, um [taktische Business-Supports](#) oder [strategische Business-Supports](#) in [Business-Support-Matrizen](#) zu planen. Der Blueprint-Planungsprozess unterstützt die Standardisierung des Business-Supports und die Effizienz des IT-Support-Roll-Outs im Unternehmen.

## Bucket

Buckets unterstützen wiederkehrende Budgetierungen und Priorisierungen durch jährliche Budgetzuweisungen. Ein Bucket ist ein Container, in dem [Projekte](#) entsprechend Budgets strukturiert werden können, die für ein Kalenderjahr definiert werden. Ein Projekt kann mehreren Buckets zugeordnet werden und bestimmte Prozentsätze der Projektkosten können den jeweiligen Buckets zugeordnet werden. Einem Bucket können keine [Projektszenarien](#) (und keine [Projektlösungen](#)) zugeordnet werden.

## Business-Architektur

Siehe [Business-Architektur des Unternehmens](#).

## Business-Architektur des Unternehmens

Die Business-Architektur des Unternehmens definiert die Business-Ansicht auf die Unternehmensarchitektur und beschreibt die [Business-Prozesse](#), [Business-Funktionen](#), [Marktprodukte](#) und [Organisationen](#) des Unternehmens.

## Business-Bewertung

Mithilfe von Business-Bewertungen können Unternehmen ein erstes Verständnis von der Ausrichtung der unterstützenden IT am Business erhalten. Business-Bewertungen werden im Kontext einer [Business-Support-Matrix](#) definiert. Die Definition von [Business-Supports](#) ist für eine Business-Bewertung nicht erforderlich.

Die Business-Bewertung ist eine [Bewertung](#) des Ist- und Soll-Supports für den [Business-Prozess](#) oder die [Domäne](#) für eine ausgewählte [Organisation](#) oder ein [Marktprodukt](#) bezüglich eines bestimmten [operativen Aspekts](#). Sobald für die [Kennzahltypen](#), die für die Business-Bewertung konfiguriert wurden, Werte definiert sind, können die Lücken zwischen dem Ist- und Soll-Support bewertet werden, sodass das Unternehmen Prioritäten für zukünftige [Projekte](#) oder die [IT-Strategie](#) festlegen kann, um die IT am Business auszurichten.

## Business-Case

Der Business-Case ist für ein [Projekt](#) definiert und bietet einen strukturierten Ansatz für die Einschätzung der Investitions- und Betriebskosten und Leistungen, die mit einem Projekt verbunden sind. Der Business-Case berechnet Standard-Leistungsmaßstäbe, wie etwa die Rendite oder den abgeschriebenen Cash Flow.

## Businessdaten

Businessdaten stellen die konkreten logischen Instanzen von [Business-Objekten](#) dar. Businessdaten werden zwischen [Applikationen](#) und deren technischen [Komponenten](#) über [Informationsflüsse](#) ausgetauscht. Sie werden in den [Business-Services](#) verarbeitet, die von den Applikationen/Komponenten bereitgestellt werden. In einer Standard-[CRUD-Matrix](#) können Anwender Konflikte in der [Businessdatennutzung](#) angezeigt bekommen und identifizieren.

## Businessdatenattribut

Ein Businessdatenattribut gibt die Merkmale für [Businessdaten](#) oder ein [Business-Objekt](#) an. Ein Businessdatenattribut hat eine Typdefinition (CHAR, BLOB, REFERENZ usw.) sowie eine Größenspezifikation. Das für Businessdaten definierte Businessdatenattribut wird von dem Business-Objekt geerbt, dem die Businessdaten untergeordnet sind.

## Businessdatennutzung

Die Businessdatennutzung beschreibt, ob [Businessdaten](#) im Kontext einer [Applikation](#), [Komponente](#), eines [Informationsflusses](#) oder [Business-Services](#) erzeugt, gelesen, aktualisiert oder gelöscht werden, und ob die Businessdaten ein Input und/oder Output eines [Architekturelements](#) sind.

Die Businessdatennutzung ist ein wichtiges Mittel zur Identifizierung von Inkonsistenzen und Redundanzen in der Applikationslandschaft eines Unternehmens. In einer Standard-[CRUD-Matrix](#) können Anwender Konflikte in der Businessdatennutzung anzeigen und identifizieren.

## Business-Frage

Mithilfe von Business-Fragen können die Schwerpunktbereiche von Anwendungsfällen des Business artikuliert werden, sodass das Unternehmen Informationen erfassen und auswerten kann, die einen bestimmten, für das Business relevanten Abfragepfad unterstützen. Business-Fragen werden Anwendern bereitgestellt, denen ein bestimmtes Anwenderprofil zugeordnet ist. Business-Fragen können in [Objekt-Cockpits](#) und [Navigationsansichten](#) implementiert werden.

## Business-Funktion

Eine Business-Funktion gibt eine nicht unterteilbare Geschäftstätigkeit an, die typischerweise im Rahmen eines oder mehrerer [Business-Prozesse](#) ausgeführt wird. Eine Business-Funktion zeichnet sich durch ihre Unabhängigkeit vom Kontext eines Business-Prozesses aus und kann dadurch über verschiedene Business-Prozesse hinweg anwendbar sein. Auf der untersten Ebene kann ein Business-Prozess über entsprechende [Business-Service-Anfragen](#) auf eine oder mehrere Business-Funktionen verweisen. Business-Funktionen können demnach als kleinste Bausteine zum Aufbau von Business-Prozessen angesehen werden.

Das Konzept der Business-Funktionen ist für die Implementierung einer serviceorientierten Architektur von grundlegender Bedeutung. Business-Funktionen standardisieren [Business-Services](#) unter dem Aspekt von Angebot und Nachfrage durch Beschreibung der IT-Erfordernisse des Business in einer gleichartigen, formalisierten und vergleichbaren Weise. Zusätzlich können [Operationen](#) für eine Business-Funktion definiert werden, um die tatsächliche Verfügbarkeit der zugeordneten Business-Services, die von Applikationen oder Komponenten bereitgestellt werden, zu beschreiben.

Business-Funktionen können entweder auch einer [Domäne](#) zugeordnet werden, um so die funktionale Struktur der Unternehmensarchitektur zu beschreiben, oder sie können einem [funktionalen Module](#) zugeordnet werden, um die spezifischen Geschäftsanforderungen aufzunehmen, die üblicherweise von einer Applikation erfüllt werden.

In Unternehmen, die ihre Business-Architektur mittels [Geschäftsfähigkeiten](#) strukturieren, befinden sich Business-Funktionen auf der untersten Ebene der Geschäftsfähigkeits-Hierarchie und stellen konkrete Geschäftsaktivitäten dar.

## Business-Funktion-Operation

Eine Business-Funktion-Operation ist eine detaillierte Beschreibung darüber, wie ein [Business-Service](#) von einer [Applikation](#) bereitgestellt wird, um eine [Business-Service-Anfrage](#) umzusetzen, die zur Durchführung einer [Business-Funktion](#) erforderlich ist. Eine Business-Funktion-Operation wird für die Business-Funktion definiert, deren Realisierung sie unterstützt. Für eine Business-Funktion definierte Business-Funktion-Operationen werden den Business-Services zugeordnet, die die Business-Funktion ausführen. Für eine Business-Funktion können mehrere Business-Funktion-Operationen definiert sein.

Die für eine Business-Funktion-Operation definierten Parameter und Rückgabewerte können für eine Operation definiert werden, um die Schnittstelle eines Serviceaufrufs zu beschreiben und nützliche Informationen für das Voranschreiten der Entwicklungstätigkeiten bereitzustellen.

## Business-Funktionskategorie

In einer Business-Funktionskategorie werden inhaltsspezifische [Business-Funktionen](#) zusammengefasst und klassifiziert. Die Kategorien werden üblicherweise aus [Business-Prozessen](#) abgeleitet.

In Business-Funktionskategorien können Sie die Business-Funktionen in Ihrem Unternehmen hierarchisch strukturieren. Jede Business-Funktion kann nur mit einer einzigen Business-Funktionskategorie verbunden sein.

## Business-Objekt

Ein Business-Objekt ist die Darstellung einer Einheit, die für die Business-Domäne des Unternehmens relevant ist. Ein Business-Objekt kann beispielsweise einen Kunden, eine Rechnung oder eine Bestellung darstellen. Business-Objekte können anhand einer Reihe von Objekteigenschaften beschrieben werden. Das Business-Objekt "Kunde" kann beispielsweise die Businessdatenattribute "Name" und "Telefonnummer" haben.

Business-Objekte werden von und zwischen [Business-Prozessen](#) oder [Business-Funktionen](#) verarbeitet und ausgetauscht. [Businessdaten](#) sind die konkrete, logische Instanziierung eines Business-Objekts.

## Business-Objektkategorie

In einer Business-Objektkategorie werden inhaltsspezifische [Business-Objekte](#) zusammengefasst und klassifiziert. In Business-Objektkategorien können Sie die Business-Objekte in Ihrem Unternehmen hierarchisch strukturieren. Jedes Business-Objekt kann nur mit einer Business-Objektkategorie verbunden sein.

## Business-Prozess

Ein Business-Prozess stellt eine Reihe von Aktivitäten dar, die Arbeiten angeben, die für das Erreichen eines Business-Ziels ausgeführt werden müssen. Typische Beispiele für Business-Prozesse sind Marketing-Services, der Verkauf von Produkten, die Bereitstellung von Services, der Vertrieb von Produkten, die Fakturierung von Dienstleistungen und das Buchen eingegangener Zahlungen. Ein Business-Prozess wird selten allein ausgeführt. Business-Prozesse stehen in einem gegenseitigen Abhängigkeitsverhältnis.

Business-Prozesse sind in einem [Business-Prozessmodell](#) hierarchisch strukturiert. Ein Business-Prozess kann, je nach Bedarf, über eine unbegrenzte Anzahl untergeordneter Prozesse verfügen. Auf der untersten Ebene kann ein Business-Prozess über entsprechende [Business-Service-Anfragen](#) auf eine oder mehrere [Business-Funktionen](#) verweisen. Die Ausrichtung des Business und der IT wird normalerweise auf einer bestimmten Ebene der Business-Prozesshierarchie ausgeführt (zum Beispiel auf der dritten Ebene in der Business-Prozesshierarchie).



## Business-Prozess-Gruppe



Eine Business-Prozess-Gruppe ist ein Container zur logischen Strukturierung von [Business-Prozessen](#). Es gibt verschiedene Möglichkeiten, Business-Prozesse logisch zu strukturieren. Daher kann jeder Business-Prozess mit mehreren Business-Prozess-Gruppen verbunden werden.

## Business-Prozess-Informationsfluss



Wie bei herkömmlichen [Informationsflüssen](#) stellt ein Business-Prozess-Informationsfluss die Übertragung von Informationen zwischen einer Quelle und einem Ziel dar. Ein Business-Prozess-Informationsfluss beschreibt den Austausch von [Business-Objekten](#) zwischen [Business-Prozessen](#). Der Business-Prozess-Informationsfluss erstellt auch eine Reihenfolge zwischen den Business-Prozessen, die er miteinander verbindet. Daher verbinden Business-Prozess-Informationsflüsse üblicherweise Business-Prozesse, die sich auf der gleichen Ebene der Business-Prozesshierarchie befinden.

## Business-Prozessmodell



Ein Business-Prozessmodell beschreibt eine Hierarchie von [Business-Prozessen](#) im Unternehmen. Es können mehrere Business-Prozessmodelle definiert werden, um Variationen im Verlauf der Zeit oder für Organisationseinheiten des Unternehmens zu ermöglichen.

## Business-Prozessmodellversion

Die Business-Prozessmodellversion ist eine Version eines bestehenden [Business-Prozessmodells](#) und dient als ein Container, der eine oder mehrere verwandte Versionsänderungen am [Business-Prozess](#) in der Business-Prozesshierarchie enthält. Normalerweise erfassen [Business-Prozessversionen](#) zeitgleich vorgenommene Änderungen am Business-Prozess in der Organisation. Da es aufgrund von Unterschieden im Aufbau der Organisation sowie in den Produktlinien, Kanälen, Kundensegmenten, Märkten oder Marken möglicherweise erforderlich ist, verschiedene Versionen desselben Business-Prozesses gleichzeitig zu verwenden, können mehrere Versionen desselben Business-Prozesses organisationsübergreifend implementiert werden. Business-Prozessversionen werden zusammen mit ihren [Business-Supports](#) geplant, die die zeitliche Perspektive des Roll-outs und die Verwendung der Business-Prozessversionen beschreiben.

## Business-Prozessvariante

Eine Business-Prozessvariante wird aus einem [Business-Prozess](#) abgeleitet und ermöglicht den jeweils Zuständigen, die Ausführung von Business-Prozessen im Detail zu analysieren. Zum Beispiel könnte der hochrangige Business-Prozess "Auftragsmanagement" Business-Prozessvarianten für die spezifische Prozessausführung in den Bereichen Automobile und Motorräder haben. Neben der Definition der [Business-Services](#), die durch die Business-Prozessvariante bereitgestellt werden, und der Modellierung dieser Business-Services in [Service-Diagrammen](#) können Anwender auch Alfabet zu einem spezifischen EPC/BPMN-Diagramm in ARIS überwechseln, oder ein Diagramm in ARIS auswählen und auf einen Business-Prozessvariante in Alfabet verlinken, um die zugrunde liegende IT zu studieren.

## Business-Prozessversion

Eine Business-Prozessversion ist ein bereitgestellter [Business-Prozess](#), der für eine [Business-Prozessmodellversion](#) bestimmt worden ist und eine Aktivität repräsentiert, die ausgeführt wird, um ein Business-Ziel zu erreichen. Normalerweise erfassen Business-Prozessversionen zeitgleich vorgenommene Änderungen am Business-Prozess in der Organisation. Da es aufgrund von Unterschieden im Aufbau der Organisation sowie in den Produktlinien, Kanälen, Kundensegmenten, Märkten oder Marken möglicherweise erforderlich ist, verschiedene Versionen desselben Business-Prozesses gleichzeitig zu verwenden, können mehrere Versionen desselben Business-Prozesses organisationsübergreifend implementiert werden.

## Business-Service

Ein Business-Service ist ein IT-Service, der von einer [Applikation](#), [Komponente](#), [lokalen Komponente](#), [Organisation](#), einem [Marktprodukt](#), [Business-Prozess](#), [ICT-Objekt](#) oder [Lösungsbaustein](#) bereitgestellt werden kann, um eine bestimmte [Business-Funktion](#) umzusetzen. Abhängig von der Lösungskonfiguration können Applikationen und Komponenten mehrere Business-Services für dieselbe Business-Funktion bereitstellen. Ein Business-Service erfüllt daher eine bestimmte [Business-Service-Anfrage](#), die von einem [Business-Prozess](#) für diese spezielle Business-Funktion gestellt wurde.

Business-Services verarbeiten [Businessdaten](#) mittels der Applikationen, Komponenten, Organisationen, Marktprodukte oder Business-Prozesse, die den Service bereitstellen. Business-Services können auch [Business-Objekte](#) mittels der ICT-Objekte oder Lösungsbausteine, die den Service bereitstellen, verarbeiten. Eine [Operation](#) kann definiert werden, um zu beschreiben, wie ein Business-Service von dem bereitstellenden Objekt bereitgestellt werden soll.

Ein Business-Service erbt den [autorisierten Anwender](#) und/oder die [autorisierten Anwendergruppen](#) vom Objekt, das ihn bereitstellt.

## Business-Service-Anfrage

Eine Business-Service-Anfrage formuliert den Bedarf an IT-Unterstützung, den ein [Business-Prozess](#) oder ein [Marktprodukt](#) hat, in Form eines [Business-Services](#), der auf einer [Business-Funktion](#) basiert. Auf der Grundlage dieser Informationen können die Anfragen konkreten Business-Services zugeordnet werden, die von [Applikationen](#) oder [Komponenten](#) bereitgestellt werden.

Der Business-Service erbt den autorisierten Anwender und/oder die autorisierten Anwendergruppen von der Applikation oder der Komponente, die ihn bereitstellt.

## Business-Support

Ein Business-Support definiert ein Objekt, wie beispielsweise eine [Applikation](#), das das Unternehmen bei dessen Geschäftstätigkeit unterstützt. Business-Supports unterstützen in der Regel [Business-Prozesse](#), obwohl ein Unternehmen seinen Business-Support auch auf [Domänen](#) beziehen könnte. Darüber hinaus werden Business-Supports typischerweise durch [Organisationen](#) ausgeführt, die für die Business-Prozesse/Domänen verantwortlich sind. Einige Unternehmen geben [Marktprodukte](#) anstelle von Organisationen an. Alfabet unterstützt die Definition und Analyse von [operativen Business-Supports](#), [Lösungs-Business-Supports](#), [strategischen Business-Supports](#) und [taktischen Business-Supports](#).

## Business-Support-Matrix

Eine Business-Support-Matrix ist die grafische Darstellung des [Business-Supports](#), der von angegebenen Objekten bereitgestellt wird, um die Planung kurz-, mittel- und langfristiger [Soll-Architektur](#) zu unterstützen. Die Matrix wird mit einer X-Achse und einer Y-Achse dargestellt. Die Business-Support-Matrix hat eine X-Achse, auf der standardmäßig die unterstützten [Business-Prozesse](#) dargestellt werden, und eine Y-Achse, auf der standardmäßig die unterstützten [Organisationen](#) dargestellt werden. In manchen Industriesegmenten ist es sinnvoller, den Business-Support für [Marktprodukte](#) als für Organisationen zu analysieren. In diesem Fall können Marktprodukte für die Y-Dimension des Business-Supports konfiguriert werden. Zudem können manche Unternehmen Business-Supports so beschreiben, dass Support eher an [Domänen](#) des Business als an Business-Prozesse geleistet werden. In diesem Fall können Domänen für die X-Dimension des Business-Supports konfiguriert werden.

Die Matrixzellen bestehen aus den Business-Supports, die das entsprechende X-Achsen-Objekt und Y-Achsen-Objekt unterstützen. Im Kontext von [Bebauungsplänen](#) oder [IT-Strategien](#) können Business-Support-Matrizen [operative Business-Supports](#), [taktische Business-Supports](#) und [strategische Business-Supports](#) anzeigen. Im Falle einer [Lösungs-Bebauungsplan-Matrix](#) werden [Lösungs-Business-Supports](#) angezeigt.

Eine Business-Support-Matrix kann für eine IT-Strategie oder einen Bebauungsplan angegeben werden, die oder der als [Blueprint](#) spezifiziert ist. Die Definition der Business-Support-Matrix für eine Blueprint-Strategie oder einen Bebauungsplan kann in eine andere Business-Support-Matrix eingebettet sein, um bei der Planung der Business-Architektur des Unternehmens als Richtlinie zur organisationsübergreifenden Planung und Standardisierung des Business-Supports hinweg zu dienen.

Alfabet unterstützt die Konfiguration von benutzerdefinierten Business-Support-Matrizen, um die gleichzeitige Erfassung einer großen Anzahl an Business-Supports für eine bestimmte Kombination von Organisationen/Marktprodukten mit Business-Prozessen/Domänen zu ermöglichen.

## C

### Clustering

Clustering ist eine Verbindung zwischen einem [physischen Gerät](#) und einem [logischen Gerät](#). Mittels Clustering kann ein physisches Gerät zu einem logischen Gerät zugeordnet werden, wobei das logische Gerät als eine Art logisches Cluster für das physische Gerät dient. Eine virtuelle Maschine könnte beispielsweise als logisches Cluster von physischen Geräten bezeichnet werden.

Ein logisches Gerät kann eine Reihe von physischen Geräten in Cluster verpacken und ein physisches Gerät kann von einer Reihe von logischen Geräten in Cluster verpackt werden.

### Compliance-Control

Eine Compliance-Control ist ein kritischer Abfrageprozess, der beantwortet werden muss, um die Compliance bestimmter Objekte mit geltenden Standards zu bewerten. Compliance-Controls werden in einem [Compliance-Katalog](#) gebündelt und sind hierarchisch strukturiert. Es gibt eine Compliance-Control der obersten Ebene und eine unbegrenzte Anzahl von Ebenen mit untergeordneten Compliance-Controls. Die Hierarchie der Compliance-Controls stellt eine Reihe von Compliance-Controls dar, die ein Objekt

durchlaufen muss, um die Compliance-Control auf der untersten Ebene zu erreichen. Die Compliance-Control auf Blattebene in der Hierarchie der Compliance-Controls bedingt die Frage, die im [Compliance-Projekt](#) gestellt wird.

Mindestens eine der für den übergeordneten Compliance-Katalog definierten [Compliance-Richtlinien](#) kann der Compliance-Control zugeordnet werden. Sobald ein Compliance-Projekt ausgeführt wird, werden die Compliance-Controls als [Compliance-Fragen](#) im Kontext des Compliance-Projekts instanziiert.

## Compliance-Domäne

Eine Compliance-Domäne stellt den Bereich des Unternehmens dar, für den ein [Compliance-Projekt](#) durchgeführt werden soll. Eine Compliance-Domäne könnte beispielsweise ein geografisches Gebiet oder ein bestimmtes Thema sein.

## Compliance-Frage

Eine Compliance-Frage stellt die kritische Frage einer Abfrage dar, die in dem [Compliance-Projekt](#) beantwortet werden muss. Compliance-Fragen werden im [Compliance-Katalog](#) als [Compliance-Controls](#) definiert und werden instanziiert, sobald der [Release-Status](#) des Compliance-Projekts auf **Aktiv** gesetzt wird. Die Compliance-Frage auf der Blattebene der Compliance-Fragenhierarchie bedingt die Fragen bezüglich der Zielobjekte in dem Compliance-Projekt.

## Compliance-Katalog

Ein Compliance-Katalog dient als Entwurf, der für mehrere [Compliance-Projekte](#) genutzt werden kann. Üblicherweise stellt ein Compliance-Katalog einen geltenden Compliance-Standard dar und ist als hierarchischer Fragenkatalog (wie z. B. COBIT oder SOX) strukturiert. Außerdem ist er für die Bewertung bestimmter Objekte in einem Unternehmen definiert.

In einem Compliance-Katalog werden [Compliance-Controls](#) gruppiert, die hierarchisch mit einer Control der obersten Ebene und einer unbegrenzten Anzahl von Ebenen für untergeordnete Controls strukturiert sind. Eine Compliance-Control auf der untersten Ebene eines Blattes in der Hierarchie bedingt eine Frage, die bezüglich eines Zielobjekts gestellt wird.

Es können eine oder mehrere Compliance-Richtlinien für einen Compliance-Katalog definiert werden. Eine [Compliance-Richtlinie](#) definiert die Regeln zur Suche nach Zielobjekten sowie zur Suche nach Personen, die für die Beantwortung der Fragen bezüglich der Zielobjekte verantwortlich sind.

Durch den Compliance-Katalog wird außerdem festgelegt, welcher [Kennzahltyp](#) als Standard-Metrik bei der Bewertung der Zielobjekte verwendet werden soll. Mit anderen Worten: Der Kennzahltyp umfasst die Werte, die von Anwendern ausgewählt werden können, um die zu den Zielobjekten gestellte Frage zu beantworten. Nur ein Kennzahltyp darf für den Compliance-Katalog definiert werden.

## Compliance-Projekt

Ein Compliance-Projekt ist eine Abfrage, die in einem Unternehmen zwecks einer Bewertung bestimmter Objektauswahl eingesetzt wird. Ein Compliance-Projekt basiert auf einem [Compliance-Katalog](#) und einer [Compliance-Domäne](#). Der Compliance-Katalog definiert die Zielobjekte des Compliance-Projekts, die zu beantwortenden Fragen sowie die für die Antwort zuständigen Personen.

Ein Compliance-Projekt dient als Instanziierung des Compliance-Katalogs für einen bestimmten Zeitraum und Gültigkeitsbereich, die von der Compliance-Domäne bestimmt werden. Ein Compliance-Projekt, das beispielsweise auf einem Compliance-Katalog basiert, der SOX darstellt, kann die SOX-Bewertung in Q1/2007 für ein regionales Tochterunternehmen des Unternehmens sein. Ein Compliance-Projekt kann nur auf einem Compliance-Katalog basieren, wenn der [Release-Status](#) des Compliance-Katalogs als [Compliance-Domäne](#) definiert ist.

Ein Compliance-Katalog kann über eine unbegrenzte Anzahl von Compliance-Projekten verfügen, die verschiedene zeit- oder regionsübergreifende Abfragen darstellen. Jedoch kann nur ein Compliance-Projekt für eine ausgewählte [Compliance-Domäne](#) und ein definiertes Datum aktiv sein. In das Compliance-Projekt werden alle [Compliance-Richtlinien](#) übernommen, die für den Compliance-Katalog definiert sind, auf dem das Projekt basiert.

Ein aktives Compliance-Projekt zeigt an, dass der Release-Status des Compliance-Projekts auf **aktiv** gesetzt wurde und mit dem Startdatum beginnt, das für das Compliance-Projekt definiert ist. Sobald das Startdatum erreicht wurde, gilt das Compliance-Projekt als aktiv.

## Compliance-Richtlinie

Für einen [Compliance-Katalog](#) wird eine Compliance-Richtlinie erzeugt. Sie definiert, welche Zielobjekte im Kontext eines [Compliance-Projekts](#) bewertet werden und welche Personen die Bewertung ausführen sollen. Dies wird durch Definition von Regeln zur Suche nach Objekten, die Ziel des Compliance-Projekts sind, sowie zur Suche nach den für die Beantwortung der Fragen bezüglich der Architekturelemente verantwortlichen Anwender erreicht. Die für die Beantwortung der Fragen identifizierten Anwender könnten die autorisierten Anwender der Zielobjekte sein, Anwender, für die Rollen für die Zielobjekte definiert sind, oder Anwender, die nicht direkt mit den Zielobjekten verbunden sind. Die Zielobjekte sowie die verantwortlichen Anwender, die nicht direkt den Zielobjekten zugehörig sind, sind mithilfe von [Abfragen](#) zu finden.

Für jede Objektklasse, auf die ein Compliance-Objekt abzielt, muss eine andere Compliance-Richtlinie definiert werden. Ein Compliance-Katalog kann über eine unbegrenzte Anzahl von Compliance-Richtlinien verfügen. Sobald alle Compliance-Richtlinien für den Compliance-Katalog erzeugt sind, müssen die Compliance-Richtlinien der [Compliance Control](#) zugeordnet werden, für die sie relevant sind. Die Zuordnung der Compliance-Richtlinie zur Compliance-Control bestimmt, welche Objekte für die von der Compliance-Control gestellten Frage bewertet werden müssen.

## CRUD-Matrix

Eine CRUD-Matrix ist ein bewährtes Mittel zur Darstellung der [Businessdatennutzung](#) von [Applikationen](#), [Komponenten](#) oder [Business-Services](#). CRUD ist ein Akronym, das sich aus den vier Hauptaktionen (Create (erzeugen), Read (lesen), Update (aktualisieren), Delete (löschen)) zwischen [Architekturelementen](#) und [Businessdaten](#) ergibt. Außerdem geben die Buchstaben I und O (Input und Output (dt.: Ein- und Ausgang)) an, ob die Businessdaten in den eingehenden oder ausgehenden Informationsflüssen der Applikationen verwendet werden. Eine Vielzahl von Analysen, die eine CRUD-Matrix verwenden, unterstützen den Anwender

bei der Ermittlung von Inkonsistenzen und Redundanzen in der Art und Weise, wie Businessdaten in der [IT-Landschaft](#) genutzt werden.

In einer CRUD-Matrix stellen Architekturelemente die eine, und Businessdaten die andere Dimension der Matrix dar. In den entsprechenden Zellen wird eine Kombination der Buchstaben C,R,U, D, I und O angezeigt, wodurch angegeben wird, ob und wie Zusammenhang zwischen dem Architekturelement und den Businessdaten besteht.

## D

### Datenaufbewahrungsrichtlinie

Eine Datenaufbewahrungsrichtlinie unterstützt die Speicherung und das Management von [Businessdaten](#). Datenaufbewahrungsrichtlinien sind ein Mittel zur Dokumentation von Standardinformationen darüber, wie Businessdaten erhalten und gespeichert werden sollen, einschließlich der Information darüber, wie lange Businessdaten aufbewahrt werden sollen, welche Regeln für die Archivierung der Businessdaten gelten, welche Mittel für die Speicherung und den Zugriff auf Businessdaten zulässig sind, sowie den erforderlichen Verschlüsselungsgrad für die Businessdaten.

### Diagrammansicht

Eine Diagrammansicht ist eine Konfiguration, die mit einem Diagramm verbunden ist. Sie ermöglicht dem Anwender, mit diesen Architekturelementen verbundene qualitative Informationen – zum Beispiel aggregierte [Kennzahlen](#) oder Attributwerte – einzublenden. Diagrammansichten können in Diagrammen implementiert werden, die [Applikationen](#), [Business-Prozesse](#), [Geräte](#), [Domänen](#), [Frameworks](#), [Plattformen](#) und [Lösungsbausteine](#) darstellen. Für Objektklassen, für die Diagramme verfügbar sind, können mehrere Diagrammansichten definiert werden.

### Diagrammansichtselement

Ein Diagrammansichtselement ist die Konfiguration, die festlegt, welche Information über die dargestellten [Objekte](#) in einer bestimmten [Diagrammansicht](#) in einem Alfabet -Diagramm angezeigt werden. Das Diagrammansichtselement erfasst die [Attribute](#) und [Kennzahlen](#), die für das Objekt angezeigt werden sollen. Die festgelegten Attribute und Kennzahlen werden dann auf den Objekten im Diagramm angezeigt. Die Objekte sind im Diagramm üblicherweise als Rechteck oder, wenn es sich um Verbindungselemente wie [Informationsflüsse](#), [Rollen](#) oder [Regeln](#) handelt, als Linie dargestellt.

### Diskussionsgruppe

Eine Diskussionsgruppe ist eine vorkonfigurierte Gruppe von Anwendern, die zugelassen sind, um [Objekte](#) bestimmter [Objektklassen](#) zu diskutieren.

Für eine Diskussionsgruppe muss mindestens eine untergeordnete, klassenbasierte Diskussionsgruppe definiert sein, durch die die Objektklasse, die Thema der Diskussion ist, und durch die für die Diskussion

implementierten Status bestimmt werden. Darüber hinaus wird bestimmt, ob die Mitglieder der Diskussionsgruppe für die diskutierten Objekte über Schreib-/Lese- oder nur über Lesezugriffsrechte verfügen, und ob im Kontext von Diskussions-Aktivitäten automatische E-Mail-Benachrichtigungen verschickt werden sollen.

Andere Diskussionsgruppen können zur Teilnahme an einer Diskussion eingeladen werden, die von der Diskussionsgruppe initiiert wurde, die die Diskussion leitet. In diesem Fall muss die einzuladende Diskussionsgruppe Teil der durchführbaren Diskussionen der Diskussionsgruppe sein. Ferner muss für die einzuladende Diskussionsgruppe eine entsprechende klassenbasierte Diskussionsgruppe definiert sein. Wenn z.B. die leitende Diskussionsgruppe eine Applikation diskutiert und eine andere Diskussionsgruppe einlädt, zur Diskussion beizutragen, muss für die eingeladene Diskussionsgruppe für die Objektklasse "Applikation" auch eine klassenbasierte Diskussionsgruppe definiert sein.

## Domäne

Eine Domäne repräsentiert eine funktionale Einheit im [Domänenmodell](#) und ermöglicht es, die Unternehmensarchitektur hierarchisch in voneinander getrennte Segmente zu unterteilen und sie zum Beispiel fachlich oder technologisch zu strukturieren. Eine Domäne basiert auf einem [Objektklassenstereotyp](#), der eine bestimmte Ebene innerhalb der Domänenhierarchie darstellt.

Eines der Ziele auf dem Weg zu einer effektiven Service-orientierten Architektur ist, die [IT-Landschaft](#) so zu strukturieren, dass jedes [Architekturelement](#) ([Business-Funktionen](#), [Business-Objekte](#), [Business-Prozesse](#), [ICT-Objekte](#), [Applikationen](#), [Komponenten](#), [Standardplattformen](#) und [Anbieterprodukte](#)) nur einer einzigen Domäne zugewiesen ist, welche den wesentlichen funktionalen Kontext für das Objekt definiert. Jedes Architekturelement kann nur zu einer Domäne gehören, die den wesentlichen funktionalen Kontext für das Objekt definiert. In einigen Fällen wird eine Domäne zugehörige Objekte haben. Das sind Objekte, die in einem funktionalen Kontext für die Domäne relevant sind, aber dieser nicht zugeordnet sind.

Für Unternehmen, die ihre Unternehmensarchitektur durch Geschäftsfähigkeiten beschreiben, referenziert jede [Geschäftsfähigkeit](#) eine Domäne im Domänenmodell. Im Kontext des Geschäftsfähigkeitsmanagements beschreibt eine Domäne Business-Aktivitäten auf einem groberen Level, während die Business-Funktionen, die sich auf der untersten Ebene der Domänenhierarchie befinden, die konkrete Aktivität beschreiben.

## Domänenglossar

Mittels eines Domänenglossars können die Anwender nach domänenspezifischer Terminologie suchen, die für die [Objekte](#) definiert ist, die einer [Domäne](#) zugeordnet sind. Domänenspezifische Terminologie umfasst Alias und Beschreibungen von Objekten.

## Domänengruppe

Eine Domänengruppe ist ein Container zur logischen Strukturierung von [Domänen](#). Es gibt verschiedene Möglichkeiten, Domänen logisch zu strukturieren. Daher kann jede Domäne mit mehreren Domänengruppen verbunden werden.

## Domänenmodell

Ein Domänenmodell beschreibt eine Hierarchie von [Domänen](#), die auf konfigurierten Domänenstereotypen basieren. Für das Unternehmen können mehrere Domänenmodelle, von denen jedes eine konfigurierte Anzahl von Ebenen in der Domänenhierarchie hat, konfiguriert werden, um im Kontext einer serviceorientierten Architektur verschiedene funktionale Entitäten beschrieben und bewertet werden.

## Domänenstereotyp

Ein Domänenstereotyp ist ein [Objektklassenstereotyp](#), der für die Objektklasse `Domain` erzeugt wird, um die Klassifikation einer Funktionseinheit im [Domänenmodell](#) darzustellen. Jede in Alfabet erzeugte [Domäne](#) basiert auf einem Domänenstereotypen.

## E

### Eigenes Objekt

Ein eigenes Objekt ist ein beliebiges [Objekt](#), das Eigentum des Anwenders ist. Der Anwender ist daher als [autorisierter Anwender](#) des Objekts definiert.

### Eigentümer

Siehe [Autorisierter Anwender](#)

## Enterprise-Release

Ein Enterprise-Release ist der Prozess, in dem Änderungen an der Business-Architektur in einem gesteuerten Prozess geplant und verwaltet werden. Für jedes Enterprise-Release wird eine Reihe von [Meilensteinen](#) spezifiziert, die die Stage-Gates für die Genehmigung und Ausführung des Releases darstellen und daher die Verwaltung des Enterprise-Release-Zyklus ermöglichen.

Für die Objektklasse "Enterprise-Release" kann eine Reihe von [Release-Statuswerten](#) konfiguriert werden, um die Statuswerte des Release-Zyklus zu definieren (z.B. Neu, Geplant, In Betrieb, Geschlossen). Zur Verwaltung der einzelnen Phasen des Release-Zyklus können [Workflows](#) konfiguriert werden.

Das Enterprise-Release besteht aus einem oder mehreren [Enterprise-Release-Elementen](#), die üblicherweise auf [Projekten](#) basieren, die die Architekturänderungen im Unternehmen bewirken. Um die Phasen der Projektregistrierung, -ausführung und -fertigstellung zu spezifizieren kann für die Objektklasse Enterprise-Release-Element auch eine Reihe von Release-Statuswerten konfiguriert werden.



## Enterprise-Release-Element

Ein Enterprise-Release-Element stellt die Leistungen dar, die für ein [Enterprise-Release](#) bereitgestellt werden. Üblicherweise basiert ein Enterprise-Release-Element auf einem [Projekt](#), das definiert wurde, um die Architekturänderung bereitzustellen. Alternativ können Enterprise-Release-Elemente auf [Applikationen](#), [Komponenten](#) oder [Standardplattformen](#) basieren. In diesem Fall stellt das Architekturelement die Leistung für das Enterprise-Release dar.

Um die Phasen der Projektregistrierung, -ausführung und -fertigstellung zu spezifizieren kann für die Objektklasse Enterprise-Release-Element auch eine Reihe von [Release-Statuswerten](#) konfiguriert werden.

## Ereignis

Ein Ereignis tritt entweder vor, während oder nach der Ausführung eines [Business-Prozesses](#) ein. Ein Ereignis kann durch Faktoren wie Meldungen, Regeln oder Fehler ausgelöst werden. Ein Ereignis kann über Verbindungselemente, wie z.B. einen [Sequenzfluss](#), mit einer [Aktivität](#) verbunden werden.

Es gibt drei Arten von Ereignissen:

- Ein Startereignis ist ein Ereignis, das vor der Ausführung eines Business-Prozesses auftritt. Ein Startereignis löst einen Business-Prozess aus und ist mit der ersten Aktivität eines [Service-Diagramms](#) verbunden - üblicherweise über einen Sequenzfluss.
- Ein Zwischenereignis ist ein Ereignis, das während der Ausführung eines Business-Prozesses auftritt. Ein Zwischenereignis befindet sich üblicherweise zwischen Aktivitäten und ist mit diesen über Servicediagramm-Verbindungselemente verbunden.
- Ein Endereignis ist ein Ereignis, das auftritt, nachdem ein Business-Prozesses ausgeführt und das Ziel des Business-Prozesses erreicht wurde. Ein Endereignis ist mit der letzten Aktivität eines Servicediagramms verbunden - üblicherweise über einen Sequenzfluss.

## Express-Ansicht

Eine Express-Ansicht ist ein Link zu einer bestimmten, Daten bereitstellenden Alfabet -Ansicht oder einem Explorer, über den Personen innerhalb oder außerhalb der Alfabet -Anwendergemeinschaft Alfabet -Informationen angezeigt bekommen können. Wenn die Express-Ansicht erzeugt wird, wird automatisch eine E-Mail-Benachrichtigung an den angegebenen Empfänger geschickt. Der Empfänger erhält eine URL, mittels der er die aktuelle Ansichtseite in Alfabet aufrufen kann.

## Externes Lösungssystem

Ein externes Lösungssystem ist ein vorgeschlagenes [externes System](#), das mit einer [Lösungsapplikation](#) über einen [Lösungs-Informationsfluss](#) verbunden ist, der in der [Soll-Architektur](#) spezifiziert wurde, die für ein [Projekt](#) definiert ist. Das externe Lösungssystem ist als ein [Lösungsobjekt](#) außerhalb des [Inventorys](#) vorhanden, bis die für die Soll-Architektur vorgeschlagenen Architekturänderungen in das Inventory eingeecheckt werden. Nach dem Einchecken wird das externe Lösungssystem zu einem realen [Inventory-Objekt](#) und verfügt dann über den [ObjektstatusGeplant](#). Der Objektstatus muss bei Betriebsbeginn manuell auf **Aktiv** geändert werden.

## Externes System

Ein externes System ist ein Element in der [IT-Landschaft](#), das sich außerhalb des Kerns der Alfabet -Lösung befindet. Ein externes System stellt typischerweise eine [Applikation](#) dar, die durch Business-Partner, z.B.: EDI-Gateway oder einen B2B-Marktplatz, verwaltet wird.

Externe Systeme werden in der IT-Landschaft über [Informationsflüsse](#) mit Applikationen verbunden. Anders als bei Applikationen, sind bei externen Systemen deren [lokale Komponenten](#) oder [Business-Services](#) sowie die Details der zugehörigen technischen Plattform für die Alfabet -Lösung von keiner Bedeutung.

## F

### Farbregel

Eine Farbregel basiert auf einer oder mehreren [Alfabe-Abfragen](#) oder nativen SQL-Abfragen, die konfiguriert wurden, um einen Satz an [Objects](#) in Standard- Alfabet -[Business-Support-Matrizen](#) und Diagrammen einzufärben. Wenn die Farbregelfunktionalität für eine Business-Support-Matrize aktiviert ist, dann werden alle aktivierten Farbregeln ausgeführt und die Matrixzellen werden entsprechend eingefärbt. Um Farbregeln auf Diagramme anzuwenden, müssen die Farbregeln zu einer [Diagrammansicht](#) zugeordnet werden. Wenn die Diagrammansicht in einem Diagramm ausgewählt wird, werden alle aktivierten Farbregeln ausgeführt, und die Diagrammobjekte entsprechend gefärbt. Die Farbregeln werden, wenn Sie ausgeführt werden, automatisch der Legende hinzugefügt.

### Föderative Architektur

Eine föderative Architektur ist eine Architektur, die üblicherweise in Silos verwaltet wird. Solch eine Architektur ermöglicht der Holdinggesellschaft, bei der Strukturierung und Kommunikation der Architektur zwischen Elementen zu unterscheiden, die vom gesamten Unternehmen verwendet werden, oder die von einigen, aber nicht allen Organisationen des Unternehmens verwendet werden, sowie denen, die nur für eine einzelne Organisation verwendet werden. Die Zuordnung von [Mandanten](#) zu Alfabet -[Objekten](#) erlaubt es der Holding-Gesellschaft, die Objekte in der [Unternehmensarchitektur](#) zu strukturieren und deren Sichtbarkeit nur auf die gewünschten Anwender zu begrenzen.

### Framework

Ein Framework dient der Definition einer semantischen Struktur für in Alfabet verwaltete [Objekte](#), unabhängig von der [Objektklasse](#), der das Objekt angehört. Ein Framework in Alfabet ist eine hierarchisch strukturierte Sammlung von [Framework-Gruppen](#), in denen Anwender durch die einzelnen Ebenen der Framework-Struktur navigieren können.

## Framework-Gruppe

Eine Framework-Gruppe dient als Container zur Strukturierung von [Objekten](#) aus unterschiedlichen [Objekt-klassen](#), um die Unternehmensarchitektur der Firma anzuzeigen, zu analysieren und zu vermitteln.

## Funktionales Modul

Ein funktionales Modul ist einer Domäne zugeordnet und bündelt [Business-Funktionen](#), um die speziellen Business-Anforderungen zu erfassen, die üblicherweise von einer [Applikation](#) erfüllt werden (oder erfüllt werden können). In diesem Sinne kann ein funktionales Modul als ein Applikationsprototyp betrachtet werden, der funktionale Aspekte erfasst, die für die Unternehmensarchitektur erforderlich sind. Eine Business-Funktion kann mehreren funktionalen Modulen zugeordnet werden.

## G

## Gateway

Ein Gateway ist ein Verbindungselement in einem [Service-Diagramm](#), durch das die Konvergenz und Divergenz der [Sequenzflüsse](#) kontrolliert wird. Ein Gateway wird üblicherweise zwischen zwei Aktivitäten angeordnet, deren Abhängigkeiten nicht durch einfache Verbindungen ausgedrückt werden können. Gateways können auch kombiniert werden, um noch komplexere Abhängigkeiten zwischen Aktivitäten darzustellen.

Es werden drei Arten von Gateways unterstützt:

- AND-Gateways verbinden eine [Aktivität](#) mit einer Reihe von Aktivitäten, die nach der Vorgänger-Aktivität gleichzeitig ausgeführt werden müssen. Sie stellen den logischen AND-Operator für die Nachfolge-Aktivitäten dar.
- OR-Gateways verbinden eine Aktivität mit einer Reihe optionaler Aktivitäten, die nach der Vorgänger-Aktivität ausgeführt werden können. Sie stellen den logischen OR-Operator für die Nachfolge-Aktivitäten dar. Das heißt, dass einige der Nachfolge-Aktivitäten ausgeführt werden. Im Gegensatz zum AND-Gateway werden jedoch nicht unbedingt alle ausgeführt.
- XOR-Gateways verbinden eine Aktivität mit einer Reihe konkurrierender Aktivitäten, von denen eine nach der Vorgänger-Aktivität ausgeführt werden muss. Sie stellen den logischen XOR-Operator für die Nachfolge-Aktivitäten dar. Das heißt, dass nur eine der Nachfolge-Aktivitäten aufgrund einer bestimmten Bedingung ausgeführt wird. Ein XOR-Gateway wird üblicherweise zur Darstellung eines Entscheidungsbaums mit einer Ja/Nein-Antwort verwendet. Die Auswahl der Nachfolge-Aktivität ist von der Antwort auf die Frage abhängig, die durch das XOR-Gateway dargestellt wird.

## generisches Attribut

Ein generisches Attribut erlaubt die Ad-hoc-Erfassung von Inhalten in einer semistrukturierten Form für folgende Elemente: [Applikation](#), [Komponente](#), [Installation](#), [Gerät](#), [Standardplattform](#), [Stack](#),

[Installationselement](#), [Standardplattform-Element](#), [Stackelement](#) oder [Stack-Konfigurationselement](#). Ein generisches Attribut kann verwendet werden, wenn eine oder mehrere Attribute nur zu Informationszwecken erforderlich sind und jedes Attribut nur für einen kleinen Teil der Objekte verwendet wird, anstatt für alle Objekte in einer [Objektklasse](#). Ein generisches Attribut ist eine Alternative zur Konfiguration einer [benutzerdefinierten Eigenschaft](#) für eine Objektklasse.

## Genutzter Business-Service

Ein genutzter Business-Service ist ein [Business-Service](#), der von einer [Applikation](#) bereitgestellt und von einer [Organisation](#) genutzt wird, um eine [Business-Service-Anfrage](#) eines [Business-Prozesses](#) zu erfüllen.

## Gerät

Ein Gerät stellt entweder reelle oder virtuelle Hardware dar, auf der [Applikationen](#) und [Komponenten](#) installiert werden können. Geräte sind mit [Installationselementen](#) verbunden und an [Standorten](#) zu finden. Ein Gerät kann als ein [physisches Gerät](#) oder als ein [logisches Gerät](#) definiert werden und kann die Quelle einer [Netzwerkroute](#) sein.

## Gerätegruppe

Eine Gerätegruppe ist ein Container zur logischen Strukturierung von [Geräten](#), um die physikalische IT-Infrastruktur des Unternehmens anzuzeigen, zu analysieren und zu kommunizieren. Es gibt verschiedene Möglichkeiten, Geräte logisch zu strukturieren. Daher kann jedes Gerät mit mehreren Gerätegruppen verbunden werden.

## Geschäftsdokument

Ein Geschäftsdokument ist ein unstrukturiertes Dokument, das oftmals nicht direkt im Zusammenhang mit der IT steht. Dabei handelt es sich typischerweise um Berichte oder Handlungsanweisungen, die benötigt werden, um die Geschäftsaktivitäten durchzuführen, die mit einer [Business-Funktion](#) verbunden sind. Das Geschäftsdokument kann als entweder als für die Business-Funktion benötigte Eingabe oder als durch die Business-Funktion generierte Ausgabe definiert werden.

## Geschäftsfähigkeit

Eine Geschäftsfähigkeit ist eine abstrakte Beschreibung davon, was in einem Unternehmen zur Erreichung der Geschäftsziele getan wird. "Management der Kundenbindung" ist beispielsweise eine abstrakte Beschreibung einer Geschäftsfähigkeit auf einer hohen Ebene (was), wobei es sich im Fall von "Fax senden" um eine konkrete Beschreibung handelt, die den Einsatz einer bestimmten Technologie angibt (wie).

Bei [Business-Funktionen](#) liegt der Schwerpunkt üblicherweise auf der detaillierten Beschreibung der Business-Aktivitäten. Geschäftsfähigkeiten hingegen betrachten Business-Aktivitäten auf Basis von, und visualisiert durch eine Hierarchie von [Domänen](#) (und Unterdomänen). Die Business-Funktionen bilden dabei die

unterste Ebene der Hierarchie. In Alfabet gibt es keine eindeutige Klasse für Geschäftsfähigkeiten. Domänen werden zur Beschreibung von Business-Aktivitäten mit einem gröberen Granularitätsgrad (Kundenverwaltung) genutzt. Business-Funktionen dienen der Beschreibung konkreter Aktivitäten (Management der Kundenbindung).

Richtlinien zur Bewertung der Geschäftsfähigkeiten eines Unternehmens sind in einer [Geschäftsfähigkeitsmatrix](#) definiert. Die Leistung von Domänen und Business-Funktionen, die einer Geschäftsfähigkeit zugeordnet sind, kann nach bestimmten [Geschäftsfähigkeitsaspekten](#) und [Kennzahlensystemen](#) vorgenommen werden. Es können ein oder mehrere Anwender zur Bewertung der Geschäftsfähigkeit definiert werden. Die Bewertung ermöglicht der Organisation, die aktuelle Leistung zu bestimmen und mit der geplanten Leistung zu vergleichen und somit besser nachzuvollziehen, welche Geschäftsfähigkeiten Kern des Business sind und weiteren IT-Support rechtfertigen. Die Bewertung kann aus verschiedenen Blickwinkeln vorgenommen werden. Dadurch können Geschäftsfähigkeiten auf Basis eines allgemeinen Verständnisses darüber, welche Aktivitätsbereiche miteinander verknüpft sind, und wo Leistungslücken und -mängel auftauchen, konsolidiert werden.

## Geschäftsfähigkeitsaspekt

Ein Geschäftsfähigkeitsaspekt stellt eine bestimmte Perspektive dar, von der ausgehend die Leistung der [Geschäftsfähigkeiten](#) eines Unternehmens bewertet wird.

## Geschäftsfähigkeitsmatrix

In einer Geschäftsfähigkeitsmatrix können Sie Richtlinien für die Bewertung der Geschäftsfähigkeiten ihres Unternehmens definieren. Die Leistung von [Domänen](#) und [Business-Funktionen](#), die einer [Geschäftsfähigkeit](#) zugeordnet sind, kann nach bestimmten [Geschäftsfähigkeitsaspekten](#) und [Kennzahlensystemen](#) durch eine definierte Gruppe von Gutachtern vorgenommen werden. [Mandanten](#) werden im Kontext von Geschäftsfähigkeitsmatrizen nicht unterstützt.

## Geschäftsrolle

Mit Geschäftsrollen können Informationen über funktionale Beziehungen erfasst werden, die für das ausgewählte Objekt relevant sind. Geschäftsrollen beschreiben funktionale Beziehungen, die nicht durch Personen oder Organisationen erfüllt werden. Typische Geschäftsrollen sind Akteure (etwa IT-Sicherheits-Manager, Nutzer privater Netzwerke, Eigentümer von Services), Personen (etwa Beteiligter, Prozesseigentümer, Genehmiger) und Stellenbeschreibungen. Geschäftsrollen dienen nur zu Informationszwecken und wirken sich nicht auf Zugriffsberechtigungen aus. Für die Klasse **Geschäftsrolle** können [Objektclassenstereotypen](#) konfiguriert werden.

## Gesetzliche Sprache

Eine gesetzliche Sprache ist eine Sprache, die es dem Unternehmen ermöglicht, behördliche Anforderungen zu erfüllen, die die Erfassung von Objekten in einer bestimmten Sprache erfordern. Die gesetzliche Sprache kann die [Primärsprache](#) des Unternehmens oder eine [Sekundärsprache](#) sein, die für die Alfabet - Benutzeroberfläche konfiguriert ist.

---

## H

## I

### ICT-Objekt

Ein ICT-Objekt (ICT = Information and Communication Technology; Informations- und Kommunikationstechnologie) ist ein abstraktes Objekt, das entweder ein [Architekturelement](#) (z. B. eine [Applikation](#), ein [Lösungsbaustein](#), eine [Komponente](#), ein [Gerät](#), eine [Standardplattform](#) oder ein [Anbieterprodukt](#)) unabhängig von dessen Versionierung oder mehrere Architekturelemente darstellt, die aus geschäftlichen oder finanziellen Gründen miteinander verbunden sind.

Ein ICT-Objekt ist Eigentum einer [Organisation](#), die üblicherweise für das Budget der Architekturelemente verantwortlich ist, die dem ICT-Objekt zugeordnet sind. Deshalb sind ICT-Objekte der Schlüssel zur Unternehmensstrategie und zur [Bebauungsplanung](#). Die Verwendung von ICT-Objekten ist insofern vorteilhaft, als dass sich der Planer beispielsweise nicht von Beginn an auf eine bestimmte Version einer Applikation festlegen muss. Wenn im weiteren Verlauf die Phase der detaillierten Planung erreicht wird, kann das ICT-Objekt durch eine spezifische, konkrete Version ersetzt werden.

ICT-Objekte können in [ICT-Objektkategorien](#) sowie in [ICT-Objektgruppen](#) strukturiert werden.

Für die Klasse **ICT-Objekt** können [Objektklassenstereotypen](#) konfiguriert werden.

### ICT-Objektanfrage

Eine ICT-Objektanfrage ist ein Objekt, das üblicherweise im Kontext eines [Workflows](#) erzeugt und genutzt wird, in dem eine Anforderung für ein neues [ICT-Objekt](#) definiert wird. Die ICT-Objektanfrage kann den [Applikationen](#), [Komponenten](#), [Geräten](#), [Standardplattformen](#) und [Anbieterprodukten](#) zugeordnet werden, die das ICT-Objekt darstellen würde. Die Definition einer ICT-Objektanfrage dient nur zu Dokumentationszwecken. Eine ICT-Objektanfrage löst nicht automatisch ein ICT-Objekt aus. Die Definition eines neuen ICT-Objekts erfolgt üblicherweise im Kontext einer [Anforderung](#). Für die Klasse **ICT-Objektanfrage** können [Objektklassenstereotype](#) konfiguriert werden.

### ICT-Objektgruppe

Eine ICT-Objektgruppe ist ein Container zur logischen Strukturierung von [ICT-Objekten](#), um so die Hauptbausteine der [IT-Landschaften](#) des Unternehmens anzuzeigen, zu analysieren und zu kommunizieren. Es gibt verschiedene Möglichkeiten, ICT-Objekte logisch zu strukturieren. Ein ICT-Objekt kann mehreren ICT-Objektgruppen zugeordnet werden, wodurch ein und dasselbe ICT-Objekt in verschiedenen semantischen Zusammenhängen betrachtet werden kann.

## ICT-Objektkatalogelement

Ein ICT-Objektkatalogelement ist Bestandteil eines [Komponentenkatalogs](#). Jedes Element ist mit einem [ICT-Objekt](#) im Unternehmen verbunden. Ein ICT-Objektkatalogelement kann mehreren Komponentenkatalogen zugeordnet sein.

Jedes ICT-Objektkatalogelement hat einen [Lebenszyklus](#), der von dem des zugehörigen ICT-Objekts unabhängig ist. Über ICT-Objektkatalogelemente kann ein ICT-Objekt über verschiedene, für einzelne Kataloge definierte Lebenszyklen verfügen. Anhand des Lebenszyklus wird erkennbar, wann das ICT-Objekt entsprechend des Umfangs des ausgewählten Katalogs als Standard definiert ist.

## ICT-Objektkategorie

In einer ICT-Objektkategorie werden inhaltsspezifische [ICT-Objekte](#) zusammengefasst und klassifiziert. In ICT-Objektkategorien können Sie die ICT-Objekte in Ihrem Unternehmen hierarchisch strukturieren. Jedes ICT-Objekt kann nur einer einzigen ICT-Objektkategorie zugeordnet werden.

## Inaktivitätsmonitor

Ein Aktivitätsmonitor ist ein [Monitor](#), der die in einem Verteiler eingetragenen Anwender über das Ausbleiben von Änderungen an [Objekten](#) einer bestimmten [Objektklasse](#) informiert. Der Eigentümer des Monitors definiert eine Reihe von Objekten, die überwacht werden sollen, sowie eine Reihe von Anwender, die benachrichtigt werden sollen, sobald der Monitor ausgelöst wird. Üblicherweise ist der Eigentümer der Anwender, der für die vom Monitor überwachten Objekte verantwortlich ist. Der Monitoreigentümer und die zu benachrichtigenden Anwender werden mittels E-Mail-Benachrichtigung informiert, ein bestimmtes Objekt während eines bestimmten Zeitraums nicht geändert oder überprüft wurde. Wenn das überwachte Objekt nicht geändert werden muss, dann kann der Monitoreigentümer das Objekt als überprüft markieren.

## Informationsarchitektur

Siehe [Informationsarchitektur des Unternehmens](#)

## Informationsarchitektur des Unternehmens

Die Informationsarchitektur des Unternehmens definiert die Informationsansicht auf die Unternehmensarchitektur und beschreibt die [Business-Objekte](#) des Unternehmens und die damit verbundenen Standards.

## Informationsfluss

Durch einen Informationsfluss wird der Austausch von [Businessdaten](#) zwischen Quell- und Ziel-[Applikationen](#), [Komponenten](#), [Geräten](#) oder [externen Systemen](#) beschrieben. Die Schnittstellenlogik, die für den Informationsaustausch benötigt wird, ist ein wesentlicher Bestandteil des Informationsflusses. Informationsflüsse zwischen Applikationen werden durch konkrete [Schnittstellensysteme](#) ausgeführt, die meist in der

---

[Plattform](#) der Applikation als Plattformkomponente eingebettet sind. Innerhalb einer Applikationsplattform kann auch eine [lokale Komponente](#) als Quell- oder Ziel-[Schnittstelle](#) definiert werden, über die die Datenübertragung des Informationsflusses ermöglicht wird. Ein bestimmter Informationsfluss kann über Quell- und Zielschnittstellen sowie auch über Schnittstellensysteme verfügen. Informationsflüsse können durch den [Verbindungstyp](#), die [Verbindungsmethode](#), die [Verbindungsfrequenz](#) des Informationsaustauschs und das [Verbindungsdatenformat](#), die den Datenaustausch beschreiben, klassifiziert werden.

## Installation

Eine Installation ist die logische Gesamtheit der installierten Elemente, aus denen sich eine Instanziierung einer [Applikation](#) oder [Komponente](#) zusammensetzt. Ein [Installationselement](#) bestimmt das [Gerät](#), auf dem das entsprechende [Architekturelement](#) installiert ist, für jedes der installierten Architekturelemente, einschließlich der Applikation oder Komponente selbst und aller Elemente des [Stacks](#) dieser Applikation oder Komponente.

## Installationsarchitektur des Unternehmens

Die Installationsarchitektur definiert die Installationsansicht auf die Unternehmensarchitektur und beschreibt die [Standorte](#), [Geräte](#) und [Installationen](#) des Unternehmens.

## Installationselement

Ein Installationselement ist Bestandteil einer [Installation](#). Jedes Installationselement ist entweder direkt mit einem [Gerät](#) verbunden, verweist durch Wiederverwendung auf ein anderes Installationselement oder verwendet [lokale Komponenten](#) direkt. Außerdem ist jedes Installationselement mit einem [Stack-Element](#) für den von der Installation installierten [Stack](#) verbunden.

## Instanz

Siehe [Objekt](#)

## Inventory

Das Inventory ist eine Sammlung aller in Alfabet definierten Objekte. Diese Objekte werden häufig als [Inventory-Objekte](#) bezeichnet. Das Inventory umfasst [Architekturelemente](#), die aktiv verwendet werden, sowie alle genehmigten Architekturelemente, die zukünftig verwendet werden sollen.

## Inventory-Objekt

Ein Inventory-Objekt ist ein Objekt im Alfabet -[Inventory](#). Die meisten der Inventory-Objekte sind [Architekturelemente](#), die von vorgeschlagenen Architekturänderungen, die für die [Soll-Architektur](#) eines [Projekts](#) definiert sind, betroffen sein könnten.



Im Gegensatz zu einem Inventory-Objekt wird ein [Lösungsobjekt](#) im Kontext einer Architektur**lösung** definiert und ist nicht Teil des Alfabets [-Inventorys](#). Sobald die in der Lösung vorgeschlagenen Architekturänderungen in das Inventory eingecheckt werden, wird das Lösungsobjekt zu einem realen Inventory-Objekt und erhält einen [Objektstatus](#), durch den angegeben wird, dass das Objekt geplant ist. Der Objektstatus muss manuell in einen aktiven Objektstatus geändert werden, wenn die operative Nutzung der Soll-Architekturelemente beginnt (meist bei Fertigstellung der Implementierung des Projekts).

## Ist-Architektur

Die Ist-Architektur umfasst alle aktiv genutzten [Architekturelemente](#) sowie Architekturelemente, für die eine zukünftige Verwendung geplant ist. Die Ist-Architektur dient als Basis für den Planungsprozess und das Vorschlagen neuer Architekturelemente für die [IT-Landschaft](#).

Im Kontext eines [Projekts](#) wird durch die Ist-Architektur der Bereich der IT-Umgebung definiert, auf den das Projekt ausgerichtet ist. Dies sind üblicherweise die Architekturelemente, die Gegenstand der Änderungen sind, oder die von den vorgeschlagenen Änderungen betroffen sein könnten. Dies können Architekturelemente aus der IT-Landschaft, wie z.B. [Applikationen](#) oder [ICT-Objekte](#), sowie Architekturelemente aus der Business-Landschaft, wie z.B. [Business-Prozesse](#) oder [Organisationen](#), sein. Die Ist-Architektur spielt eine zentrale Rolle bei der Bewertung potentieller Konflikte und Redundanzen zwischen Projekten sowie bei der Identifikation möglicher Projektrisiken.

## IT-Fähigkeit

Eine IT-Fähigkeit ist eine Basisfähigkeit, wie z.B. ein Infrastrukturservice, die im Kontext von Rechenzentrumsoperationen bereitgestellt wird. "Mainframe Operations", "Database Management" oder "Backup & Recovery" sind Beispiele für typische IT-Fähigkeiten.

IT-Fähigkeiten können einer [Technologie](#) zugeordnet werden, die die IT-Fähigkeit potentiell unterstützt. Die [Komponenten](#), die die Technologie für die IT-Fähigkeit tatsächlich ausführen, können genauer spezifiziert werden. Die Komponenten "Oracle 10i" und "Oracle 11g" die Technologie "Oracle RDBMS" aktivieren, die die IT-Fähigkeit "Database Management" ausführt. Die Verwendung einer Technologie und deren zugehörigen Komponenten durch eine IT-Fähigkeit an einem bestimmten [Standort](#) wird in einer [IT-Fähigkeitszuordnung](#) definiert.

Eine IT-Fähigkeit kann über eine unbegrenzte Anzahl von untergeordneten IT-Fähigkeiten sowie über eine unbegrenzte Anzahl von untergeordneten Ebenen in der Baumstruktur verfügen. Eine IT-Fähigkeit kann nur einer IT-Fähigkeit zur Zeit untergeordnet sein.

## IT-Fähigkeitszuordnung

Mittels einer IT-Fähigkeitszuordnung werden die [Technologien](#) und deren zugehörige [Komponenten](#) dokumentiert, die eine [IT-Fähigkeit](#) an einem oder mehreren [Standorten](#) ausführen (üblicherweise Rechenzentren).

## IT-Landschaft

Die IT-Landschaft beschreibt die Gesamtheit der [Architekturelemente](#) und deren Beziehungen im Unternehmen, einschließlich der [Applikationsarchitektur](#), der [Business-Architektur](#), der [Informationsarchitektur](#), der technischen Architektur und der Installationsarchitektur.

## IT-Strategie

Eine IT- Strategie beschreibt die [Zielarchitektur](#), die am besten geeignet ist, um die zugrundeliegende Unternehmensstrategie zu unterstützen. Da die IT-Strategie ein langfristiges Ziel eines Unternehmens darstellt, wird sie nicht nur für einen bestimmten Zeitraum definiert. Die IT-Strategie als solche gibt die Richtung für den [Bebauungsplan](#) sowie [Architektur-Lösungen](#) vor, die im Kontext von [Projekten](#) entwickelt werden. Eine IT-Strategie kann als [Blueprint](#) angegeben werden, der als Orientierungshilfe bei der Planung der IT-Strategie und der Bebauungsplanung dient.

Eine IT- Strategie beinhaltet eine oder mehrere [strategische Bebauungsplan-Matrizen](#), die die zukünftigen [strategischen Business-Supports](#) enthalten, die speziell für die IT-Strategie definiert wurden. Da die IT-Strategie Gegenstand regelmäßiger Überprüfungen und Neudefinitionen ist, können mehrere IT-Strategien für ein Unternehmen existieren. Eine IT-Strategie kann beispielsweise für ein geschätztes Marktwachstum von 40% oder 20% definiert sein.

## J

## K

### Kennzahl

Eine Kennzahl ist die qualitative oder quantitative Bewertung der Leistung eines Objekts. Kennzahlen können manuell mittels eines externen, mit Alfabet verbundenen Systems eingegeben werden oder in Alfabet berechnet oder aggregiert werden. Jede Kennzahl basiert auf einem [Kennzahltyp](#), durch den die Dimension und der Rahmen der Bewertung definiert wird. Eine oder mehrere Kennzahltypen werden in einem [Kennzahlensystem](#) gebündelt und können in [Bewertungen](#), [Priorisierungsschemata](#) und Portfolios implementiert werden.

### Kennzahlensystem

Ein Kennzahlensystem fasst ein oder mehrere [Kennzahltypen](#) zusammen, um die [Bewertung](#) einer bestimmten Dimension der Leistung eines Objekts durchzuführen. Die Anwender können eine [Kennzahl](#) für jeden Kennzahltyp, der in der Bewertung enthalten ist, definieren. Typische Kennzahlensysteme sind zum Beispiel "Komplexität" oder "Standardisierungsgrad". Das Kennzahlensystem "Komplexität" könnte zum Beispiel die Kennzahltypen "Anzahl an Schnittstellen", "Anzahl an Modulen" und "Stand der Technik" haben, und das Kennzahlensystem "Standardisierungsgrad" könnte einen einzelnen Kennzahlentyp haben, der ebenfalls "Standardisierungsgrad" heißt.

Ein Kennzahlensystem kann einmal konfiguriert und dann für verschiedene [Objektklassen](#) verwendet werden. Dadurch können alle systemrelevanten Klassen wie zum Beispiel Applikation, Komponente und Standardplattform einheitlich bezüglich zum Beispiel ihrer Komplexität oder ihrem Standardisierungsgrad beurteilt werden.

Kennzahlensysteme können auch in [Priorisierungsschemata](#) gruppiert werden, die dann als Achsen eines [Portfolios](#) implementiert werden können.

Der Zugriff auf die Kennzahlensysteme in Alfabet -Bewertungen kann über [Anwenderprofile](#) kontrolliert werden. Ein Anwender kann ein Kennzahlensystem sehen, das entweder keinem Anwenderprofil, oder dem Anwenderprofil, mit dem der Anwender angemeldet ist, zugeordnet ist.

## Kennzahl-Lookup-Tabelle



Eine Kennzahl-Lookup-Tabelle ist ein Mittel zur Bewertung des [Basis-Gefährdungspotentials](#) einer bestimmten Gruppe von Objekten und zur Bestimmung, ob die Objekte hinsichtlich des [Risikos](#) beurteilt werden sollen. In der Kennzahl-Lookup-Tabelle werden zwei unterschiedliche [Kennzahlensysteme](#) erfasst, die zur Bewertung des Basis-Gefährdungspotentials genutzt werden. In einem Kennzahlensystem werden mehrere [Kennzahltypen](#) gebündelt, die die Fragen darstellen, die beantwortet werden müssen, um zu bestimmen, ob ein Objekt hinsichtlich des Risikos beurteilt werden soll. In einem anderen Kennzahlensystem werden mehrere Kennzahltypen definiert, die die Übertragung aller Fragen in einen Risikorelevanzwert ermöglichen – das ist ein Wert, durch den die Höhe des Risikos für das Objekt bestimmt und festgelegt wird, ob das Objekt in den nächsten Schritt der Risikobeurteilung übergehen soll oder nicht. In der Kennzahl-Lookup-Tabelle wird daher die Zuordnung von Antworten zu den Fragen und der Risikorelevanzwert angegeben. Jeder potenziell für eine Frage definierte Wert verfügt über einen entsprechenden Risikorelevanzwert.

## Kennzahltyp

Ein Kennzahltyp definiert eine Bewertungsdimension für die Leistung eines Objekts in der [IT-Landschaft](#). Die [Kennzahl](#) ist der für den Kennzahltyp im Kontext eines Objekts der IT-Landschaft definierte Wert.

Ein oder mehrere Kennzahltypen können in einem [Kennzahlensystem](#) gebündelt werden, dass zur [Bewertung](#) eines Objekts genutzt wird. Zum Beispiel können die Kennzahltypen "Anzahl an Schnittstellen", "Anzahl an Modulen" und "Stand der Technik" einem Kennzahlensystem "Komplexität" zugeordnet werden, und der Kennzahltyp "Standardisierungsgrad" kann einem Kennzahlensystem zugeordnet werden, dass ebenfalls "Standardisierungsgrad" heißt.

Jedem Kennzahlensystem können ein oder mehrere Kennzahlen zugeordnet sind. Kennzahlen können:

- durch den Anwender manuell eingegeben werden.
- vom Anwender aus vordefinierten Werten ausgesucht werden.
- durch Alfabet nach definierten [Berechnungsregeln](#) berechnet werden.
- auf Grundlage eines von Alfabet bereitgestellten kundenspezifischen Codes berechnet werden.
- über ein Schnittstellensystem wie ADIF importiert werden.

Kennzahltypen, für die Werte vordefiniert sind, können mit einer [Symbolgalerie](#) verbunden werden, über die die Kennzahlen in [Diagrammansichten](#) und benutzerdefinierten [konfigurierten Berichten](#) als Symbol statt als numerischer Wert angezeigt werden können.

## Klasse

Siehe [Objektklasse](#).

## Klassenbasierte Risikomanagementvorlage

In einer klassenbasierten Risikomanagementvorlage können Sie die [Objektklasse](#) definieren, die das Ziel einer [Risiko](#)bewertung ist. Für alle Objektklassen, deren Risiko bewertet wird, kann durch Erfassen der Fragekatalog-Elemente und des Risikorelevanzwerts eine bestimmte [Kennzahl-Lookup-Tabelle](#) definiert werden.

Die für eine übergeordnete [Risikomanagementvorlage](#) definierten Einstellungen, einschließlich des Schadensrisiko-[Kennzahltyps](#), des Risikowahrscheinlichkeits-Kennzahltyps und des Risiko-[Portfolios](#), werden von allen Objekten geerbt, die zu einer Risiko-Objektgruppe gehören, die auf die Risikomanagementvorlage verweist.

## Kollaborationsthema

Ein Kollaborationsthema ist ein formloser Austausch von Informationen und Ideen zu einem bestimmten Objekt oder einer Ansicht in Alfabet zwischen Mitgliedern der Anwendergemeinschaft. Sie können ein Kollaborationsthema für ein [Objekt](#) in einem [Objektprofil/Objekt-Cockpit](#) oder einer bestimmten Ansichtsseite erstellen, und alle eingeladenen Anwender können dem Kollaborationsthema einen Beitrag hinzufügen. Jeder Anwender mit den Zugriffsberechtigungen „schreibgeschützt“ (ReadOnly) und „Lesen/Schreiben“ (ReadWrite) für die Objektansicht oder Ansichtsseite kann ein Kollaborationsthema ins Leben rufen.

## Komponente

Eine Komponente ist ein wiederverwendbarer Funktionalitätsbaustein, der entweder per Hard- oder Software implementiert wird. In Alfabet stellen Komponenten üblicherweise Folgendes dar:

- Eine [Schnittstelle](#) zum Datenaustausch (siehe [Informationsfluss](#))
- Ein gemeinsam genutztes Element, das, wenn installiert, Business-Services für verschiedenen Applikationen (z.B. eine Datenbank) bereitstellen kann
- Ein untergeordnetes System einer Applikation, das als separate Einheit mit einem eigenen, unabhängigen [Lebenszyklus](#) verwaltet wird.
- Eine halb beendete Applikation, wie z.B. eine Applikation, die konfiguriert werden muss, um zu funktionieren, und daher mit verschiedenen Konfigurationen verwendet wird (es kann z.B. eine bestimmte Konfiguration von SAP CO als Basis für verschiedene Applikationen mit unterschiedlichen Konfigurationen genutzt werden)

Üblicherweise bietet eine Komponente anders als ein [Business-Service](#) den Endanwendern keine Funktionalität. Stattdessen kann eine [lokale Komponente](#) in einem bestimmten Applikationskontext zur Unterstützung einer [Applikation](#), die Business-Services für Anwender bereitstellt, eingesetzt werden. Üblicherweise stellt die lokale Komponente im Kontext einer Applikation [technische Services](#) bereit.

Im Gegensatz zu lokalen Komponenten, können auch Standardkomponenten entwickelt, verwaltet und einzeln und unabhängig von den Applikationen, die sie nutzen, verwendet werden. In Alfabet werden Standardkomponenten einfach als Komponenten bezeichnet. Ähnlich wie Applikationen, werden Standardkomponenten auf einer [Plattform](#) betrieben, die aus Hard- und Softwarekomponenten (lokalen Komponenten) besteht, die erforderlich sind, um die Komponente zu betreiben. Für Komponenten des Typs "Service" können [technische Services](#) definiert werden, um technische Anforderungen zu erfüllen, die erforderlich sind, um [Business-Service-Anfragen](#) zu unterstützen.

Ferner kann eine Komponente auf ein [Anbieterprodukt](#) verweisen, wodurch angegeben wird, dass sie aus dem Anbieterprodukt abgeleitet wurde. Komponenten können in einer [Komponentengruppe](#), einer [Komponentenkategorie](#) und/oder in einem [Komponentenkatalog](#) strukturiert sein.

## Komponente, lokal

Siehe [lokale Komponente](#).

## Komponentengruppe

Eine Komponentengruppe ist ein Container zur logischen Strukturierung von [Komponenten](#), um die IT-Standards des Unternehmens anzuzeigen, zu analysieren und zu kommunizieren. Es gibt verschiedene Möglichkeiten, Komponenten logisch zu strukturieren. So kann jede beliebige Komponente mit mehreren Komponentengruppen verbunden werden, wodurch dieselbe Komponente in verschiedenen semantischen Zusammenhängen betrachtet werden kann.

"SAP BW" wird z.B. üblicherweise als Mitglied einer Komponentengruppe "Business Intelligence Tools" betrachtet. Jedoch könnte "SAP BW" auch Mitglied der Gruppen "Reporting Tools" und "OLAP Tools" sein.

## Komponentenkatalog

Ein Komponentenkatalog bietet eine Übersicht über den Grad der Standardisierung von [Komponenten](#) für einen definierten Zeitraum. So kann ein Unternehmen festlegen, ob eine Komponente als Standardkomponente definiert ist, über eingeschränkten Support verfügt oder für eine zukünftige Verwendung nicht genehmigt ist, etc. In Alfabet können sowohl die [Ist-Architektur](#) als auch vorgeschlagene Architekturänderungen analysiert und auf Compliance geprüft werden. Dadurch wird die Entwicklung der technischen Architektur in Ausrichtung an der Unternehmensstrategie unterstützt.

Ein Komponentenkatalog bezieht sich normalerweise auf funktionale, geografische oder organisatorische Teileinheiten des Unternehmens. Eine Komponente kann zu mehr als einem Katalog gehören und über verschiedene [Lebenszyklen](#) in verschiedenen Katalogen verfügen.

Eine Komponente, die einem Komponentenkatalog zugeordnet ist, wird als [Komponentenkatalogelement](#) bezeichnet. Jedes Komponentenkatalogelement hat einen Lebenszyklus, der von der zugehörigen Komponente unabhängig ist. Der Lebenszyklus gibt an, wann die Komponente im Rahmen des Katalogs als

Standard definiert ist. Wenn eine Komponente nicht in einem bestimmten Komponentenkatalog aufgeführt ist, gilt die Komponente während ihres definierten Lebenszyklus für das gesamte Unternehmen als verfügbar.

## Komponentenkatalogelement

Ein Komponentenkatalogelement ist Bestandteil eines [Komponentenkatalogs](#). Jedes Element ist mit einer Standard-[Komponente](#) im Unternehmen verbunden. Ein Komponentenkatalogelement kann mehreren Katalogen zugeordnet sein.

Jedes Komponentenkatalogelement hat einen [Lebenszyklus](#), der von der zugehörigen Komponente unabhängig ist. Über Komponentenkatalogelemente kann eine Komponente über verschiedene, für jeden Katalog definierte Lebenszyklen verfügen. Der Lebenszyklus gibt an, wann die Komponente im Rahmen des Katalogs als Standard definiert ist.

## Komponentenkategorie

In einer Komponentenkategorie werden inhaltsspezifische [Komponenten](#) zusammengefasst und klassifiziert. In Komponentenkategorien können Sie die Komponenten in Ihrem Unternehmen hierarchisch strukturieren. Jede Komponente kann nur einer Komponentenkategorie zugeordnet werden.

## Komponentenmodul

Ein Komponentenmodul ist ein Mittel zur logischen Gruppierung von [lokalen Komponenten](#) für eine ausgewählte [Applikation](#) oder [Standard-Komponente](#). Eine lokale Komponente kann zu mehreren Komponentenmodulen zugeordnet werden. Die Business-, die technische und die Informationsarchitektur des Komponentenmoduls ist eine Aggregation der Business-, der technischen und der Informationsarchitektur der lokalen Komponenten, die im Komponentenmodul gebündelt sind.

## Komponententest

Ein Komponententest ist ein definierter Plan zum Testen eines ausgewählten Objekts und zum Dokumentieren der Testergebnisse nach der Ausführung des Tests. Komponententests können für folgende Objektklassen konfiguriert werden: [Applikation](#), [Komponente](#), [Komponentenmodul](#), [Lokale Komponente](#), [Installation](#), [Gerät](#), [Standardplattform](#) und [Anbieterprodukt](#).

## Konsistenzmonitor

Ein Konsistenzmonitor ist ein [Monitor](#)typ, der die systemweite [Objekt](#)pflege in der Alfabet-Datenbank unterstützt. Ein Konsistenzmonitor wird für die regelmäßige Suche nach Inkonsistenzen zwischen Objekten konfiguriert. Alle Konsistenzmonitore basieren auf einer [Alfabet-Abfrage](#) oder einer nativen SQL-Abfrage, über die die [Objektklassen](#) definiert werden, die Ziel des Monitors sind, sowie die zu ermittelnden inkonsistenten [Eigenschaften](#). Wenn eine Inkonsistenz von der Abfrage gefunden wird, warnt der Monitor den

[autorisierten Anwender](#) des Objekts mittels eine [Aufgabe](#) bezüglich der Inkonsistenz. Der pünktliche Abschluss der Aufgabe, der von einem Konsistenzmonitor ausgelöst wird, kann vom Lösungsadministrator verfolgt werden.

## Kosten

Kosten stellen einen Geldwert dar, der für ein Objekt in der [IT-Landschaft](#) für einen bestimmten Aktivitäts- oder Servicetyp für einen bestimmten Zeitraum erforderlich ist.

## Kostenart

Eine Kostenart ist eine [Kosten](#)-Klassifikation. Kostenarten werden vom Unternehmen definiert, um Vergleichbarkeit bei der Definition und Bewertung von Investitions- und Betriebskosten von [ICT-Objekten](#), [Applikationen](#), [Installationen](#), [Projekten](#) und [Service-Produkten](#) zu gewährleisten. Kostenarten sind entscheidend für die Definition von [Business-Cases](#) sowie für die Kostenplanung für Projekte. Für Kostenarten kann eine unbegrenzte Anzahl an untergeordneten Kostenarten definiert werden, um Bottom-Up-Kostenabschätzung und -analyse durchzuführen.

## Kostenstelle

Eine Kostenstelle dient der zentralen Definition von [Kosten](#) für eine bestimmte Zeitspanne, die dann einer Gruppe von Architekturobjekten ([Applikationen](#), [Installationen](#), [ICT-Objekten](#), [Projekten](#) oder [Service-Produkten](#)) nach einem definierten Zuordnungsschema zugeordnet werden.

Eine Kostenstelle basiert auf einem [Kostenstellentyp](#), der als Vorlage für die Definition von Kostenstellen dient. Die einer Kostenstelle zugeordneten Kosten basieren auf bestimmten [Kostenarten](#), die für den zugehörigen Kostenstellentyp vordefiniert sind. Eine Kostenstelle kann einer unbegrenzten Anzahl von [Kostenstellengruppen](#) zugeordnet werden.

## Kostenstellengruppe

Eine Kostenstellengruppe ist ein Container zur Gruppierung und hierarchischen Strukturierung von [Kostenstellen](#). Eine Kostenstellengruppe kann einer unbegrenzten Anzahl von Untergruppen sowie einer unbegrenzten Anzahl von Hierarchieebenen in der Baumstruktur zugeordnet werden. Eine Kostenstellengruppe kann nur einer übergeordneten Kostenstellengruppe untergeordnet werden.

Jeder Kostenstellengruppe kann eine unbegrenzte Anzahl von Kostenstellen zugeordnet werden. Eine Kostenstelle kann einer unbegrenzten Anzahl von Kostenstellengruppen zugeordnet werden.

## Kostenstellentyp

Ein Kostenstellentyp ist eine Vorlage zur Erzeugung von [Kostenstellen](#). Ein oder mehrere [Kostenarten](#) können dem Kostenstellentyp als Standardkostenarten zugeordnet werden. Wenn eine Kostenstelle erzeugt

---

wird, basiert sie auf einem Kostenstellentyp. Die Kostenarten, die dem Kostenstellentyp zugeordnet sind, werden automatisch auch der Kostenstelle zugeordnet.

## L

### Landeseinstellungen

Eine Landeseinstellung stellt die Basiskonfiguration der Standard-[Primärsprache](#) dar, die auf der Benutzeroberfläche angezeigt wird, wenn sich der Anwender zum ersten Mal anmeldet, sowie der Formate für Datum, Uhrzeit, und Zahlen usw., die in der Alfabet -Benutzeroberfläche zur Erfassung und Darstellung von Daten, Uhrzeiten und Zahlen verwendet werden.

### Landschaft

Siehe [IT-Landschaft](#)

### Layer

Siehe [Plattform-Layer](#)

### Lebenszyklus

Ein Lebenszyklus beschreibt die Abfolge von Phasen, die ein Architekturelement durchläuft. In Alfabet weisen viele [Objekte](#) (z.B. [Applikationen](#), [Komponenten](#), [Standardplattformen](#), [Business-Supports](#)) einen Lebenszyklus auf, obwohl nicht für alle Objekte ein Lebenszyklus definiert sein muss. Ein Lebenszyklus besteht aus [Lebenszyklusphasen](#), durch die der Aktivitäts- oder Produktivitätsstatus eines Objekts beschrieben wird. Alle Lebenszyklusphasen sind an den jeweils vorhergehenden und nachfolgenden Lebenszyklusphasen ausgerichtet.

Die Lebenszyklusdefinition umfasst auch die Definition des aktiven Zeitraums eines Objekts. Der aktive Zeitraum eines Objekts ist der Zeitraum, in dem das Objekt in Betrieb ist. Deshalb liegt der aktive Zeitraum eines Objekts zwischen dem Start- und Enddatum des Objekts. Der aktive Zeitraum kann an einer oder mehreren festgelegten Lebenszyklusphasen, die den Zeitraum darstellen, in dem das Objekt in Betrieb ist, ausgerichtet werden. In der Regel wird festgelegt, dass der [Objektstatus](#) des Objekts im Zeitraum zwischen Start- und Enddatum aktiv ist.

Ferner sollten die Lebenszyklusdefinitionen der abhängigen Objekte an dem Lebenszyklus des Objekts ausgerichtet sein, für das sie definiert wurden. Beispielsweise sollten die Lebenszyklen aller [lokalen Komponenten](#), [Informationsflüsse](#) und [Business-Supports](#) am Lebenszyklus der Applikation ausgerichtet sein, der sie zugeordnet sind. Im Fall von Komponenten sollten die Lebenszyklen aller lokalen Komponenten, Informationsflüsse und [technischer Services](#) am Lebenszyklus der Komponente ausgerichtet sein.

Ferner kann die Lebenszyklusdefinition einer Applikation die Spezifikation von Vorgänger- und Nachfolgeversionen beinhalten und die Lebenszyklusdefinition von Komponenten kann Nachfolgeversionen haben.



Nachfolgeversionen werden standardmäßig demselben [ICT-Objekt](#) wie der Original-Applikation/-Komponente zugeordnet, aber dies kann bei Bedarf geändert werden.

## Lebenszyklusphase

Eine Lebenszyklusphase beschreibt einen Aktivitäts- oder Produktivitätsstatus im [Lebenszyklus](#) eines [Objekts](#). Zu den typischen Lebenszyklusphasen zählen z.B.: Evaluation, Pilot, Production, Limited Production und Shut Down. Der aktive Zeitraum des Lebenszyklus ist der Zeitraum, in dem das Objekt produktiv und somit an den Start- und Enddaten des Objekts ausgerichtet ist. Es können eine oder mehrere Lebenszyklusphasen undefiniert bleiben. In diesem Fall werden sie ausgelassen. Jede Lebenszyklusphase wird an der vorhergehenden und nachfolgenden Lebenszyklusphase ausgerichtet, die definiert wurde.

## Lebenszyklusstatus

Siehe [Lebenszyklusphase](#).

## Lesezeichen

Ein Lesezeichen ist ein Verweis auf eine Ansichtseite, ein [Objektprofil](#) (oder [Objekt-Cockpit](#)), einen [Bericht](#) oder auf ein in einem Explorer ausgewähltes Objekt. Über ein Lesezeichen können die Anwender wichtige Ansichten in Alfabet schnell aufrufen.

Lesezeichen können in Lesezeichenverzeichnissen strukturiert werden. Ein Lesezeichen kann nur einem Lesezeichenverzeichnis zugeordnet werden.

## Logisches Gerät

Ein logisches Gerät ist ein [Gerät](#), das das logische [Clustering](#) von [physischen Geräten](#) ermöglicht. In einem logischen Gerät können mehrere physische Geräte zusammengefasst werden. Ein physisches Gerät kann in mehreren logischen Geräten zusammengefasst werden.

## Lokale Komponente

Eine lokale Komponente ist eine Komponente, die im Rahmen einer [Applikation](#) oder einer [Standard-Komponente](#) definiert ist. Lokale Komponenten haben dieselben Attribute wie Standardkomponenten, sind jedoch nicht außerhalb des Kontexts der Applikation oder Komponente, für die sie definiert sind, wiederverwendbar.

Eine lokale Komponente erbt [autorisierte Anwender](#) und/oder [autorisierte Anwendergruppen](#) von der Applikation oder der Komponente, die sie bereitstellt.

## Lokale Lösungskomponente

Eine lokale Lösungskomponente ist eine vorgeschlagene [lokale Komponente](#), die im Rahmen einer [Lösungsapplikation](#) oder Standard-[Lösungskomponente](#) definiert ist, die in der [Soll-Architektur](#) spezifiziert wurde, die für ein [Projekt](#) definiert ist. Die lokale Lösungskomponente ist als ein [Lösungsobjekt](#) außerhalb des [Inventorys](#) vorhanden, bis die für die Soll-Architektur vorgeschlagenen Architekturänderungen in das Inventory eingeecheckt werden. Nach dem Einchecken wird die lokale Lösungskomponente zu einem realen [Inventory-Objekt](#) und verfügt dann über den [ObjektstatusGeplant](#). Der Objektstatus muss bei Betriebsbeginn manuell auf **Aktiv** geändert werden.

## Lösungsapplikation

Eine Lösungsapplikation ist eine vorgeschlagene neue [Applikation](#) oder eine vorgeschlagene neue Version einer Applikation, die in der [Soll-Architektur](#) spezifiziert wurde, die für ein [Projekt](#) definiert wurde. Für die Klasse **Lösungsapplikation** können [Objektklassenstereotype](#) konfiguriert werden. Diese Objektklassenstereotype werden in der Regel an den Objektklassenstereotypen ausgerichtet, die für die Klasse **Applikation** verfügbar sind.

## Lösungsbaustein

Ein Lösungsbaustein unterstützt die fachliche Planung gemäß TOGAF. Das Konzept der Architekturbau- steine (ABBs) kann in Alfabet über [Domänen](#) erfasst werden. Durch die Konfiguration eines Domänenstereotyps zur Erfassung von ABBs auf der untersten Ebene der Domänenhierarchie können die Funktionalitäten, die das Unternehmen umsetzt, über die Zuordnung der ABBs zu Business-Funktionen spezifiziert werden.

## Lösungs-Bebauungsplan-Matrix

Eine Lösungs-Bebauungsplan-Matrix ermöglicht die Planung von [Lösungs-Business-Support](#) für die [Sollar- chitektur](#), die für ein [Projekt](#) definiert ist. Für ein Projekt können mehrere Lösungs-Bebauungsplan-Matri- zen definiert werden. In der Lösungs-Bebauungsplan-Matrix wird eine Matrix mit einer Reihe von [Organisa- tionen](#) (oder [Marktprodukten](#)) auf der einen und einer Reihe von [Business-Prozessen](#) (oder [Domänen](#)) auf der anderen Achse angezeigt. In der Lösungs-Bebauungsplan-Matrix kann für jede [Lösungsapplikation](#), die für die Sollarchitektur des Projekts definiert ist, und für jede Applikation, die in der [Ist-Architektur](#) des Pro- jekts definiert ist, ein Lösungs-Business-Support erzeugt werden.

## Lösungs-Business-Prozess

Ein Lösungs-Business-Prozess ist ein [Business-Prozess](#), der im Kontext eines [Lösungs-Business-Prozess- modells](#) vorgeschlagen wurde. Der Lösungs-Business-Prozess ist eine Kopie eines realen, im Inventory vor- handenen Business-Prozesses und kann daher geändert werden, ohne dass der eigentliche Business-Pro- zess im Inventory davon betroffen ist. Erst nachdem das Lösungs-Business-Prozessmodell in das Inven- tory eingeecheckt wurde, wird der vorhandene Business-Prozess von dem Lösungs-Business-Prozess über- schrieben.

Im Kontext eines Lösungs-Business-Prozessmodells können [Business-Service-Anfragen](#), [Domänen](#) und [Organisationen](#) für die Lösungs-Business-Prozesse definiert werden. Diese Änderungen werden innerhalb des Lösungs-Business-Prozessmodells vorgenommen und betreffen nicht die Inventory-Objekte, auf denen sie basieren.

Sobald ein Lösungs-Business-Prozessmodell genehmigt wird, kann das Lösungs-Business-Prozessmodell samt aller Lösungs-Business-Prozesse in das [Inventory](#) eingecheckt werden. Dadurch werden diese Objekte zu Inventory-Objekten.

## Lösungs-Business-Prozessmodell

Eine Lösungsdomäne ist eine vorgeschlagene Alternative für eine Domänenhierarchie. Eine Lösungsdomänenhierarchie besteht aus [Lösungsdomänen](#). Es können zwar mehrere Lösungsdomänenmodelle definiert sein, aber nur eines der Lösungsdomänenmodelle kann für das Einchecken genehmigt werden. Sobald ein Lösungsdomänenmodell in das Inventory eingecheckt wird, wird das vorhandene [Domänenmodell](#) überschrieben.

Solange ein Lösungsdomänenmodell keinen genehmigten Status aufweist und noch nicht in das [Inventory](#) eingecheckt wurde, gelten das Lösungsdomänenmodell sowie dessen Lösungsdomänen als "Schattenobjekte" – Kopien des Domänenmodells und der Domänen, die nicht im Inventory, sondern nur in der Lösung vorhanden sind. Änderungen die im Lösungsdomänenmodell vorgenommen werden, haben keine Auswirkungen auf die Inventory-Objekte, auf denen die Lösungsobjekte basieren, solange das Lösungsdomänenmodell nicht in das Inventory eingecheckt ist.

## Lösungs-Business-Support

Ein Lösungs-Business-Support ist ein vorgeschlagener [Business-Support](#), der im Kontext der [Sollarchitektur](#) spezifiziert wurde, die für ein [Projekt](#) definiert ist. Ein Lösungs-Business-Support kann auf einer [Lösungsapplikation](#) oder einer vorhandenen [Applikation](#) basieren. Ein Lösungs-Business-Support beschreibt die vorgeschlagene IT-Unterstützung für eine [Organisation](#)/ein [Marktprodukt](#) für die Unterstützung eines [Business-Prozesses](#)/einer [Domäne](#). Ein Lösungs-Business-Support kann auch definiert werden, um die Stilllegung eines operativen Business-Supports zu planen.

## Lösungsdomäne

Eine Lösungsdomäne ist eine [Domäne](#), die im Kontext eines [Lösungsdomänenprojekts](#) vorgeschlagen wurde. Die Lösungsdomäne ist eine Kopie einer realen, im Inventory vorhandenen Domäne und kann daher geändert werden, ohne dass die eigentliche Domäne im Inventory davon betroffen ist. Erst nach dem Einchecken des Lösungsdomänenprojekts in das Inventory überschreibt das Lösungsdomänenprojekt das vorhandene Domänenmodell.

Im Kontext eines Lösungsdomänenprojekts können untergeordnete Komänen hinzugefügt oder entfernt werden. Alle [Business-Funktionen](#), [Business-Objekte](#), [Business-Prozesse](#), [Applikationen](#), [ICT-Objekte](#), [Komponenten](#), [Standardplattformen](#) und [Anbieterprodukte](#), die einer gelöschten Lösungsdomäne zugeordnet sind, können einer Nachfolger-Lösungsdomäne entweder als [primäres Objekt](#) oder als [zugehöriges Objekt](#) zugeordnet werden. Diese Änderungen werden innerhalb des Lösungs-Domänenprojekts vorgenommen und betreffen nicht die [Inventory-Objekte](#), auf denen sie basieren.

Sobald ein Lösungsdomänenprojekt genehmigt ist, können das Lösungsdomänenprojekt und alle Lösungsdomänen in das [Inventory](#) eingecheckt werden. Die neue Lösungsdomäne überschreibt dann die ursprüngliche Domäne im Inventory. Alle neuen Objekte und Referenzen, die für das Lösungsdomänenprojekt erzeugt wurden, werden ebenfalls in das Inventory geschrieben.

## Lösungsdomänenprojekt

Ein Lösungsdomänenprojekt ist eine vorgeschlagene Alternative zu einem Bereich des aktuellen [Domänenmodells](#). Die Domäne, die die übergeordnete Domäne aller zu ändernden Domänen ist, wird aus dem [Inventory](#) ausgecheckt. Dadurch wird ein Lösungsdomänenprojekt erzeugt. Die Zieldomäne und deren untergeordneten Domänen werden dem Lösungsdomänenprojekt als Lösungsdomänen hinzugefügt. Die Domänen im Domänenmodell können jeweils nur Ziel eines Lösungsdomänenprojekts sein. Wenn ein Anwender versucht, eine Domäne auszuchecken, die bereits Teil eines anderen Lösungsdomänenprojekts ist, wird eine Fehlermeldung angezeigt.

Das Lösungsdomänenprojekt und dessen untergeordnete Domänen gelten als "Schattenobjekte" - Kopien der ursprünglichen Domäne, die nur im Lösungsdomänenprojekt vorhanden sind und nicht Teil des Inventors sind. Sobald die Domäne ausgecheckt und das Lösungsdomänenprojekt erzeugt wurde, werden alle weiteren Änderungen an der Inventory-Domäne nicht in die entsprechende Lösungsdomäne übertragen.

Im Kontext des Lösungsdomänenprojekts kann die Lösungs-Domänenhierarchie durch Erzeugen und Löschen von Lösungsdomänen strukturiert werden. Alle Objekte, die eine primäre Domäne oder eine zugehörige Domäne referenzieren, die zum Löschen vorgeschlagen wurde, können einer Nachfolgedomäne neu zugeordnet werden.

Sobald ein Lösungsdomänenprojekt einen genehmigten Release-Status erhält, kann es wieder in das Inventory eingecheckt werden. Die Zieldomäne und deren untergeordnete Domänen im Inventory werden im Domänenmodell gelöscht und durch die neue Domänenhierarchie ersetzt, die im Lösungsdomänenprojekt erzeugt wurde.

## Lösungs-Gerät

Ein Lösungsgerät ist ein vorgeschlagenes [Gerät](#), das für ein [ICT-Objekt](#) definiert ist, welches in der für ein Projekt spezifizierten [Soll-Architektur](#) definiert ist. Das Lösungsgerät ist als ein [Lösungsobjekt](#) außerhalb des [Inventorys](#) vorhanden, bis die von der Lösung vorgeschlagenen Architekturänderungen in das Inventory eingecheckt werden. Nach dem Einchecken wird das Lösungsgerät zu einem realen [Inventory-Objekt](#) und verfügt dann über den [Objektstatus](#) **Geplant**. Der Objektstatus muss bei Betriebsbeginn manuell auf **Aktiv** geändert werden.

## Lösungs-Informationsfluss

Ein Lösungs-Informationsfluss ist ein vorgeschlagener [Informationsfluss](#), der in der [Sollarchitektur](#), die für ein [Projekt](#) definiert ist, für eine [Lösungsapplikation](#) oder [Lösungskomponente](#) definiert ist. Als ein [Lösungsobjekt](#) ist der Lösungsinformationsfluss nur außerhalb des [Inventorys](#) vorhanden, bis die für die Soll-Architektur vorgeschlagenen Architekturänderungen in das Inventory eingecheckt werden. Nach dem Einchecken wird der Lösungs-Informationsfluss ein reales [Inventory-Objekt](#) und erhält den [Objektstatus](#) **Plan**. Der Objektstatus muss bei Betriebsbeginn manuell auf **Aktiv** geändert werden.

## Lösungskomponente

Eine Lösungskomponente ist eine vorgeschlagene neue [Komponente](#) oder eine vorgeschlagene neue Version einer Komponente, die in der [Soll-Architektur](#) spezifiziert wurde, die für ein [Projekt](#) definiert ist. Für die Klasse **Lösungskomponente** können [Objektklassenstereotype](#) konfiguriert werden. Diese Objektklassenstereotype werden in der Regel an den Objektklassenstereotypen ausgerichtet, die für die Klasse **Komponente** verfügbar sind.

## Lösungsobjekt

Ein Lösungsobjekt ist ein beliebiges [Architekturelement](#), das im Kontext einer [Soll-Architektur](#) für ein [Projekt](#) vorgeschlagen wurde. Lösungsobjekte können auf Objekten basieren, die der [Ist-Architektur](#) des Projekts zugeordnet sind, oder sie können im Kontext der Soll-Architektur von Grund auf erzeugt werden. Zu den Lösungsobjekten gehören [Lösungsapplikationen](#), [Lösungskomponenten](#), [Lösungsgeräte](#), [Lösungs-Informationsflüsse](#), [lokale Lösungskomponenten](#), [externe Lösungssysteme](#), [Lösungs-Standardplattformen](#), [technische Lösungsservices](#) und [Lösungs-Business-Supports](#). Ein Lösungsobjekt ist außerhalb des [Inventorys](#) vorhanden, bis die in der Soll-Architektur vorgeschlagenen Architekturänderungen in das Inventory eingeecheckt werden. Nach dem Einchecken wird das Lösungsobjekt zu einem realen [Inventory-Objekt](#) und verfügt dann über den [ObjektstatusPlan](#). Der Objektstatus muss bei Betriebsbeginn manuell auf **Aktiv** geändert werden.

## Lösungs-Standardplattform

Eine Lösungs-Standardplattform ist eine vorgeschlagene [Standardplattform](#), die für eine [Lösungsapplikation](#) oder [Lösungskomponente](#) angegeben ist, die in der [Soll-Architektur](#) spezifiziert wurde, die für ein [Projekt](#) definiert ist. Die Lösungs-Standardplattform ist als ein [Lösungsobjekt](#) außerhalb des [Inventorys](#) vorhanden, bis die für die Soll-Architektur vorgeschlagenen Architekturänderungen in das Inventory eingeecheckt werden. Nach dem Einchecken wird die Lösungs-Standardplattform zu einem realen [Inventory-Objekt](#) und verfügt dann über den [ObjektstatusGeplant](#). Der Objektstatus muss bei Betriebsbeginn manuell auf **Aktiv** geändert werden.

## M

### Mandant

Ein Mandant dient der Organisation und Strukturierung der [föderativen Architektur](#) einer Holdinggesellschaft. Durch die Zuordnung von Mandanten zu [Objekten](#) kann die Holdinggesellschaft die Objekte in der [Unternehmensarchitektur](#) strukturieren, um zu regulieren, ob die Objekte für einige oder alle Tochtergesellschaften sichtbar sind. Nur Anwender, die einem Mandanten zugeordnet sind, sehen Objekte mit dieser Mandantendefinition. Ein Objekt, das keinem Mandanten zugeordnet wurde, wird als nicht zum Eigentum eines Mandanten zugehörig eingestuft und ist daher für alle Anwender mit den entsprechenden Zugriffsberechtigungen in der Holdinggesellschaft sichtbar.

Der Einsatz von Mandanten in der Alfabet -Lösung ist optional. Wenn in einem Unternehmen Mandanten implementiert sind, dann hat die Regulierung der Sichtbarkeit eines Objekts durch die Mandatenzuordnung Vorrang vor allen anderen Zugriffsrechtskonzepten in Alfabet. Beispielsweise muss dem [autorisierten](#)

[Anwender](#) eines Objekts der entsprechende Mandant zugewiesen werden, um auf das Objekt, dessen autorisierter Anwender er ist, zugreifen zu können.

Im Bereich eines Mandanten gelten die herkömmlichen Regeln zur Steuerung der Zugriffsberechtigungen. Daher kann ein Anwender, dem ein Mandant zugeordnet ist, nur für diejenigen der für den Mandanten sichtbaren Objekte Schreib-/Lesezugriffsrechte erhalten, für die er über autorisierten Zugriff verfügt, das heißt für die er entweder der autorisierter Anwender, der [Stellvertreter](#), Mitglied einer [autorisierten Anwendergruppe](#) oder [Diskussionsgruppe](#) ist, eine [regelbasierte Zugriffsberechtigung](#) hat oder einen Workflow oder eine Aufgabe ausführt.

## Marktprodukt

Ein Marktprodukt ist das physikalische Artefakt oder der Service, der vom Unternehmen produziert und vermarktet wird. Ein Marktprodukt erfordert einen [Business-Support](#), der von [Applikationen](#) bereitgestellt wird. Für die Objektklasse **Marktprodukt** können [Objektklassenstereotype](#) konfiguriert werden.

## Marktproduktgruppe

Eine Marktproduktgruppe ist ein Container zur logischen Strukturierung von [Marktprodukten](#), um so das Produkt- und Serviceangebot des Unternehmens anzuzeigen, zu analysieren und zu kommunizieren. Es gibt verschiedene Möglichkeiten, Marktprodukte logisch zu strukturieren. Jedes Marktprodukt kann folglich mehreren Marktproduktgruppen zugeordnet werden.

## Maß

Maße beschreiben eine Kategorie von Messungen, die für das Unternehmen gemessen werden können. Ein Maß könnte z.B. "Kundenzufriedenheit" oder "Antwortzeit" sein. Die Instantiierung eines Maßes ist eine [Messung](#), die den tatsächlich gemessenen Wert des Maßes für ein aktives [Zeitreihen-Kennzahlensystem](#) darstellt. Jedes Maß ist einer Maßkategorie zugeordnet, wodurch die Verantwortlichkeiten und die Autorisierung für die Maße in der Maßkategorie sichergestellt wird.

Maße können auch [Zielsetzungen](#) zugeordnet werden, die für ein [Steuerungselement](#) definiert sind. In diesem Kontext kann der Wert, der für ein solches Maß zu einem bestimmten Zeitpunkt erfasst wurde, erfasst werden. Maße können im Hinblick auf eine tatsächliche Bewertung der Zielsetzung gemessen werden, und für die mit der Zielsetzung verbundenen Maße kann ein Zielwert definiert werden. Üblicherweise wird ein Maß nur einer Zielsetzung zugeordnet.

## Masterplattform

Eine Masterplattform ist eine Generalisierung einer [Standardplattform](#). Eine Masterplattform basiert auf einer [Plattformvorlage](#), welche [Plattform-Layers](#) und [Plattform-Tiers](#) strukturiert. Im Gegensatz zu einer Standardplattform, in der das Element eine [Komponente](#) ist, ist das Masterplattformelement eine [Komponentenkategorie](#).

## Masterplattformelement

Ein Masterplattformelement ist Bestandteil einer [Masterplattform](#). Ein Masterplattformelement ist mit einer [Komponentenkategorie](#) verbunden.

## Masterplattformkategorie

In einer Masterplattformkategorie werden inhaltsspezifische [Masterplattformen](#) zusammengefasst und klassifiziert. In Masterplattformkategorien können Sie die Masterplattformen in Ihrem Unternehmen hierarchisch strukturieren. Jede Masterplattform kann mit nur einer Masterplattformkategorie verbunden werden.

## Matrixansicht

Eine Matrixansicht ist ein spezifisches Segment einer [Business-Support-Matrix](#), die für die spezifischen Anforderungen eines Anwenders relevant ist. Die Matrixansicht ist für eine [Bebauungsplan-Matrix](#) definiert und umfasst eine begrenzte Gruppe an [Business-Prozessen](#)/Domänen und [Organisationen/Marktprodukten](#) in der Business-Support-Matrix. Einzelne [Business-Supports](#), die für die Matrixansicht nicht relevant sind, können ausgeblendet werden. Für eine Bebauungsplan-Matrix kann eine unbegrenzte Anzahl von Matrixansichten definiert werden. Auf diese Weise kann eine große und komplexe Business-Support-Matrix so reduziert werden, dass sie für einen bestimmten Anwender gut zu verwalten und relevant ist.

## Meilenstein

Ein Meilenstein dient der Planung und Verfolgung des Fortschritts und der Fertigstellung von [Projekten](#) oder [Enterprise-Releases](#). Meilensteine können im Kontext einer [Meilensteinvorlage](#) definiert werden, die mehrere vordefinierte Meilensteine zu einer Gruppe bündelt, oder sie können als individuelle [Ad-hoc-Meilensteine](#) unabhängig von einer Meilensteinvorlage erzeugt werden. Änderungen an geplanten Zieldaten und die Gründe für die Datumsänderung können in einer Meilensteinhistorie verfolgt und rekonstruiert werden. Die Umsetzung eines Meilensteins wird auf Ebene der [Projektgruppe](#) oder des Enterprise-Releases verfolgt.

## Meilensteinvorlage

Meilensteinvorlagen werden entsprechend der vom Unternehmen verfolgten Projektmethodik definiert, um die Stage-Gates oder [Meilensteine](#) der [Projektmethodik](#) oder des [Release-Zyklus des Unternehmens](#) zu definieren.

Eine Meilensteinvorlage fasst eine Reihe von Meilensteinen zusammen, z. B. die Definition der Abfolge und die Toleranzgrenze für jeden einzelnen Meilenstein und die Definition des Farbschemas für Nachverfolgung und Berichtswesen. Die Anwender können die Meilensteinvorlage entweder direkt einem Projekt oder Enterprise-Release zuordnen, oder, im Fall von Projekte, eine [Projektvorlage](#) auswählen und dem Projekt zuordnen.

Für jeden Meilenstein werden anhand der Konfiguration in der Meilensteinvorlage automatisch die anvisierten Zieltermine festgelegt. Diese Termine können aber für jeden Meilenstein manuell bearbeitet werden.

Anwender können aus einem Projekt jeden Meilenstein löschen, den sie für unnötig halten. Einem Projekt oder Enterprise-Release kann nur eine Meilensteinvorlage zugeordnet werden. Die Umsetzung von Meilensteinen kann für Enterprise-Releases oder Projekte nachverfolgt werden. Im Fall von Projektmeilensteinen kann die Nachverfolgung auf Ebene des Projekts, der [Projektgruppe](#) oder des [Buckets](#) vorgenommen werden.

## Meldungsfluss

Ein Meldungsfluss ist ein Konnektor in einem [Service-Diagramm](#), durch den angegeben wird, dass Informationen zwischen den durch Swim-Lanes dargestellten Agenten oder zwischen Aktivitäten mit demselben Agenten ausgetauscht werden. Ein Meldungsfluss kann eine [Aktivität](#) und ein [Ereignis](#) über verschiedene [Swim-Lanes](#) (aus verschiedenen [Pools](#)) oder zwei Aktivitäten in derselben Swim-Lane verbinden - insbesondere wenn ein [Business-Objekt](#) zwischen den zwei Aktivitäten übertragen wird.

## Messung

Eine Messung ist die Instantiierung eines [Maßes](#) und ermöglicht die Erfassung von ermittelten und angestrebten Messungen für das Maß innerhalb eines aktiven [Zeitreihen-Zeitraums](#). Beispielsweise könnte eine Messung den tatsächlich für das Maß "Kundenzufriedenheit" gemessenen Wert "5" liefern.

Die vorgenommenen Messungen für die definierten Maße können für das Unternehmen im Allgemeinen im Zeitverlauf bewertet werden. Die angestrebten Zielmessungen für [Zielsetzungen](#), die für ein bestimmtes [Steuerungselement](#) definiert sind, können ebenfalls im Zeitverlauf bewertet werden. Ziel- und gemessene Messungen können bei Bedarf bearbeitet werden.

## Migration

Eine Migration ist ein Prozess, in dem [Applikationen](#), [Lösungsbausteine](#) und/oder [ICT-Objekte](#) durch andere Applikationen/Lösungsbausteine/ICT-Objekte ersetzt werden. In Alfabet werden Migration im Kontext der [Bebauungsplanung](#) oder strategischen Planung geplant. Eine Migration beinhaltet üblicherweise [Migrationsregeln](#), die stufenweise Beschreibungen und Termine für jeden Schritt der geplanten Umwandlung angeben.

Migrationen können einer bestimmten [IT-Strategie](#), einem [Bebauungsplan](#) oder einem [Projekt](#) zugeordnet werden. Wenn die Migration einer IT-Strategie oder einem Bebauungsplan zugeordnet ist, dann kann überprüft werden, ob die vorgeschriebenen [Business-Supports](#), die für die von einer Migrationsregel betroffenen Applikationen, Lösungsbausteine, und ICT-Objekte definiert sind, mit der geplanten Migration in Einklang sind.

Eine Migration kann mehreren [Migrationsgruppen](#) zugeordnet werden.

## Migrationsgruppe

Eine Migrationsgruppe ist ein Container zur logischen Strukturierung von [Migrationen](#), um die physikalische IT-Infrastruktur des Unternehmens anzuzeigen, zu analysieren und zu kommunizieren. Es gibt



verschiedene Möglichkeiten, Migrationen logisch zu strukturieren. Daher kann jede Migration mit mehreren Migrationsgruppen verbunden werden.

## Migrationsregel



Eine Migrationsregel ist ein Schritt in einer Reihe aufeinander folgender Schritte, die notwendig sind, um eine geplante [Migration](#) durchzuführen. Die Migrationsregel beschreibt den in diesem Schritt erfolgenden Übergang von einem Vorgängerobjekt auf ein Nachfolgeobjekt sowie das Zieldatum zur Fertigstellung des Schritts. Vorgängerobjekte und Nachfolgeobjekte können [Applikationen](#), [Lösungsbausteine](#) und/oder [ICT-Objekte](#) sein.

## Mitarbeiter

Bei einem Mitarbeiter handelt es sich um eine Person in einer [Organisation](#), die einen bestimmten [Personalbedarf](#) decken soll.

## Monitor

Alfabet bietet eine Vielzahl von Monitoren, um die Anwender über an bestimmten [Objekten](#) vorgenommene Änderungen, die weitere Aktivitäten erfordern, über nicht wie geplant ausgeführte Überprüfungen oder über Transaktionen, die laut den dokumentierten Plänen durchgeführt werden sollten, zu informieren. Solche Monitore warnen die eingetragenen Anwender per E-Mail-Benachrichtigungen. Die Überwachung erfolgt über einen bestimmten Zeitraum in regelmäßigen Intervallen. Alfabet stellt zur Zeit folgende Monitor-typen bereit: [Aktivitätsmonitore](#), [Zeitmonitore](#), [Inaktivitätsmonitore](#), [Benachrichtigungsmonitore](#) und [Konsistenzmonitore](#).

# N

## Navigationsansicht

Eine Navigationsansicht ist eine konfigurierte Startseite für Alfabet, die wie eine Webseite im Internet aussieht und sich auch so verhält. Die Navigationsansicht bietet den Anwendern Verknüpfungen zu Alfabet - Funktionalitäten und -Dokumenten, eingebettete Berichte, Bilder und informative Texte. Navigationsansichten dienen als Alternative zur Standardstartseite von Software AG.

## Navigationsseite

Über eine Navigationsseite können Sie eine Startseite mit einem vordefinierten Layout der Elemente konfigurieren. Die Konfiguration der Navigationsseite wurde durch die flexiblere Konfiguration der [Navigationsansicht](#) ersetzt und ist aus Gründen der Abwärtskompatibilität noch verfügbar.

## Navigationsseite

Mit Navigationsseiten können Unternehmen eine Seite mit Verknüpfungen und Informationen für die Anwender des Unternehmens konfigurieren, um sie effizient zu bestimmten Funktionen in der Alfabet -Benutzeroberfläche zu leiten. Die Navigationsseite, die als Startseite dient, wird zu einem [Anwenderprofil](#) zugeordnet. Die Navigationsseite kann so konfiguriert werden, dass sie Anwender bei ihrer Arbeit in Alfabet auf verschiedene Weise unterstützt. Eine Navigationsseite kann frei konfiguriert werden und könnte z. B. Links zu bestimmten Funktionalitäten in Alfabet, Links zu anderen Navigationsseiten, Informationstexten oder Bildern zu unternehmensspezifischen Workflows in der Alfabet -Lösung, zu Standard-Framework-Ansätzen wie Zachmann oder URL-Links zu internen Dokumenten oder dem Internet enthalten.

## Netzwerk

Ein Netzwerk ist ein System bestehend aus [Geräten](#) und anderen physischen Netzwerken, die über [Netzwerkrouuten](#) verbunden sind. Ein Netzwerk kann entweder eine Quelle oder ein Ziel einer Netzwerkroute sein. Die Netzwerkdefinition umfasst einen räumlichen Umfang wie z. B. WLAN, LAN, Internet usw. Ein Netzwerk kann untergeordnete Netzwerke haben.

## Netzwerkroute

Eine Netzwerkroute stellt die Weiterleitung eines [Netzwerks](#) oder eines [Geräts](#) an ein anderes Netzwerk dar. Netzwerkrouuten basieren auf einem Gerät des Typs Router und stellen die physische Infrastruktur zur Verfügung, die notwendig ist, um [Informationsflüsse](#) zu aktivieren, die über eine [Installation](#) installiert werden.

## Nutzen

Der Nutzen stellt einen Geldwert dar, der während eines gegebenen Zeitraums von einer Aktivität oder einem Service verdient werden soll, die/der mit einem Objekt in der [IT-Landschaft](#) verbunden ist. Der Nutzen ist der Schlüssel zur Definition des [Business-Case](#) eines [Projekts](#) und dient als Basis für die Analyse der Rendite.

## Nutzenart

Eine Nutzenart ist eine [Nutzen](#)-Klassifikation. Nutzentypen repräsentieren üblicherweise den Gewinn in einem Business-Case und werden vom Unternehmen definiert, um Konsistenz bei der finanziellen Bewertung von Aktivitäten und Leistungen für [Projekte](#) zu gewährleisten. Nutzenarten sind entscheidend für die Definition von [Business-Cases](#) in Alfabet.

## O

### Objekt

Ein Objekt ist ein Bestandteil der Alfabet -Architektur, das üblicherweise ein Element in der [Unternehmensarchitektur](#) beschreibt. Ein Objekt gehört zu einer [Objektklasse](#), die für das Alfabet -Metamodell definiert ist. Bestimmte Objekte können auch als [Architekturelemente](#), Artefakte oder Instanzen bezeichnet werden. In Alfabet verfügt eine Vielzahl von Objekten über Start- und Enddaten, durch die der geplante Aktivitätszeitraum für das Objekt definiert wird. Ferner haben die Objekte eine [Lebenszyklus](#)definition und einen [Objektstatus](#), durch den die Objekte als aktiv verwendet, als für die Nutzung geplant oder als in der Vergangenheit genutzt unterschieden werden können.

### Objekt-Cockpit

Ein Objekt-Cockpit bietet den Anwendern eine direkte und transparente Übersicht über Daten eines ausgewählten [Objekts](#). Es handelt sich hierbei um eine verkürzte und fokussierte Darstellung von Objektdaten, üblicherweise aus Sicht einer bestimmten Perspektive heraus, wie zum Beispiel einer Architekturperspektive oder einer strategischen Perspektive.

Pro Objektprofil können mehrere Objekt-Cockpits verfügbar sein. zum Beispiel könnte ein Objekt-Cockpit für die Objektklasse Applikation Daten anzeigen, die erforderlich sind, um die Applikation im Kontext der Ist-Architektur zu verstehen, während ein anderes Cockpit Daten anzeigt, die die Bebauungsplanung betreffen und ein drittes Cockpit Daten anzeigt, die die Kosten- und Budgetplanung betreffen. Das Objekt-Cockpit ist eine Konfiguration, die zusätzlich zum umfassenderen [Objektprofil](#) verfügbar ist.

### Objektklasse

Eine Objektklasse ist ein Konstrukt, das die [Objekte](#), die für diese Klasse instanziiert werden, semantisch beschreibt. Objektklassen im Klassenmodell haben einen spezifizierten Satz von Standard-[Objektklasseneigenschaften](#) und können ferner auch [benutzerdefinierte Eigenschaften](#) haben, die von einem Lösungsentwickler konfiguriert wurden. Objektklassen referenzieren üblicherweise andere Objektklassen und erzeugen so ein semantisches Netzwerk, das z.B. die [IT-Landschaft](#) des Unternehmens beschreiben kann.

### Objektklasseneigenschaften

Alle [Objektklassen](#) im Klassenmodell verfügen über vorkonfigurierte Eigenschaften, die der semantischen Beschreibung der Objektklasse dienen. Einige Standardeigenschaften sind [Pflictheigenschaften](#) und werden in Alfabet -Editoren gelb hervorgehoben.

### Objektklassenstereotyp

Ein Objektklassenstereotyp ist eine Unterklassifikation innerhalb einer [Objektklasse](#). Eine zulässige Objektklasse kann über mehrere Objektklassenstereotypen verfügen, von denen jeder einen bestimmten Satz an Attributen, Referenzdatendefinitionen und Klassenkonfigurationen erfasst. Beispielsweise die Objektklasse

„Applikation“, die die Objektklassenstereotypen „Business-Applikationen“ und „Technische Applikationen“ haben kann.

Nur eine beschränkte Anzahl an Objektklassen unterstützt die Konfiguration von Objektklassenstereotypen.

## Objektprofil

Im Objektprofil werden Informationen zusammengefasst, die für ein ausgewähltes [Objekt](#) relevant sind. Im Objektprofil werden üblicherweise [Objektklasseneigenschaften](#) und deren Werte für das Objekt sowie Arbeitsbereiche mit Ansichten und Berichten angezeigt, die für das ausgewählte Objekt relevant sind. Die Anwender können die verfügbaren Ansichten aufrufen und die Daten über das ausgewählte Objekt definieren oder bearbeiten.

Üblicherweise werden Objektprofile entsprechend der Anforderungen der Anwendergemeinschaft konfiguriert. Daher können für eine Objektklasse abhängig des [Anwenderprofils](#), mit dem die Anwender Alfabet aufrufen, verschiedene Objektprofile mit unterschiedlichen Daten und Funktionalitäten verfügbar sein. Mit jedem Objektprofil können mehrere [Objekt-Cockpits](#) verbunden sein, die konfiguriert wurden, um bestimmte Daten über das Objekt zu visualisieren.

## Objektstatus

Ein Objektstatus beschreibt den operativen Status eines [Objekts](#) im Unternehmen. Der Objektstatus gibt an, ob ein Objekt aktiv verwendet wird, die Nutzung geplant ist oder das Objekt in der Vergangenheit bereits genutzt wurde. Da durch die Start- und Enddaten eines Objekts der geplante Aktivitätszeitraum des Objekts festgelegt wird, muss der Objektstaus von **Plan** zu **Aktiv** geändert werden, sobald das Startdatum eines Objekts erreicht ist. Gleichmaßen muss der Objektstatus von **Aktiv** zu **Stillgelegt** geändert werden, wenn das Enddatum des Objekts erreicht wird. Wenn für eine Objektklasse das [Lebenszyklus](#)-Konzept verfügbar ist, werden Start- und Enddatum des Objekts zunächst am aktiven Zeitraum des Objekts ausgerichtet.

Der Objektstatus eines Objekts kann von einem Anwender geändert werden, der über Schreib- und Lesezugriffsberechtigungen für das Objekt verfügt. Für Objekte wie [Applikationen](#) oder [Komponenten](#) müssen beim Ändern des Objektstatus relevante Objektabhängigkeiten berücksichtigt werden. Für Applikationen muss der den Objektstatus ändernde Anwender z. B. berücksichtigen, ob der Objektstatus auf die der Applikation zugeordneten [Informationsflüsse](#) oder [Business-Supports](#) übertragen werden soll.

## Operation des technischen Service

Eine Operation des technischen Service ist eine detaillierte Beschreibung darüber, wie ein [technischer Service](#) von einer [Komponente](#) bereitgestellt werden soll, um eine Komponente bei der Bereitstellung eines [Business-Services](#) zu unterstützen. Eine Operation wird für den technischen Service definiert, deren Ausführung sie unterstützt. Die für einen technischen Service definierte Operation kann den Business-Services zugeordnet werden, die von der Komponente bereitgestellt und unterstützt werden. Für einen technischen Service können mehrere Operationen definiert sein.

Für die Klasse **Operation des Technischen Services** können [Objektklassenstereotype](#) konfiguriert werden.

## Operativer Aspekt

Ein operativer Aspekt ist eine Facette des Business, die möglicherweise von der IT unterstützt wird. Standardklassen, die in Alfabet operative Aspekte darstellen, umfassen Marke, Kundensegment, Markt und Vertriebskanal. Es können jedoch auch andere Objektklassen oder [Objektklassen-Stereotype](#) als operative Aspekte implementiert werden.

Für jede konfigurierte Aspektklasse können mehrere operative Aspekte erzeugt werden, um die individuellen Facetten des Business, die Unterstützung brauchen, zu formulieren. Zum Beispiel könnten Kapitalmarkt, Einzelhandel, Wachstumsmarkt und Direktbankgeschäft Aspekte für die Klasse Markt sein.

Operative Aspekte sind im Kontext Planung und Analyse des [Business-Supports](#) relevant und ermöglichen es dem Planer, die Aspekte des Business zu verstehen, zu dem eine [Organisation](#), ein [Marktprodukt](#), ein [Business-Prozess](#) oder eine [Domäne](#) beiträgt. Das Verständnis des Beitrags individueller Business-Supports zu operativen Aspekten des Unternehmens lässt erkennen, wo in Zukunft Support benötigt wird, um die [IT-Strategie](#) des Unternehmens umzusetzen. Operative Aspekte können für [operative Business-Supports](#), [taktische Business-Supports](#), [strategische Business-Supports](#) und [Lösungs-Business-Supports](#) definiert werden. Potenzielle Lücken oder Redundanzen im Support operativer Aspekte können im Kontext von [Business-Support-Matrizen](#) verstanden und analysiert werden.

Beachten Sie, dass das Konzept operativer Aspekte keine Verbindung zu dem Konzept [Aspektbewertungen](#) hat, die zur Bewertung der Leistung von [Applikationen](#) und [Komponenten](#) implementiert werden.

## Operativer Business-Support

Ein operativer Business-Support ist ein aktiver oder geplanter [Business-Support](#), der aktuell bereitgestellt wird oder als Folge der laufenden Entwicklungs- oder Rollout-Tätigkeiten bereitgestellt werden wird.

Ein operativer Business-Support kann von einer [Applikation](#) oder, wenn der Support über interne oder externe Services bereitgestellt wird, von einer [Organisation](#) bereitgestellt werden. Der operative Business-Support bietet Unterstützung für einen [Business-Prozess](#) (und [Business-Prozessversion](#)) oder eine [Domäne](#) und kann für eine Organisation oder ein [Marktprodukt](#) bereitgestellt werden.

## Organisation

Eine Organisation beschreibt eine administrative oder funktionale Einheit im Unternehmen. Organisationen bilden eine sich auf sich selbst beziehende Hierarchie. Eine Organisation kann zwar über eine unbegrenzte Anzahl von untergeordneten Organisationen verfügen, jedoch nur einer übergeordneten Organisation zugeordnet sein. Eine Organisation kann, genau wie ein [autorisierter Anwender](#), für ein [Objekt](#) eine primäre Verantwortung haben. Ferner kann auch definiert sein, dass der Organisation eine wichtige [Rolle](#) in Bezug auf ein Objekt zugeordnet ist. Organisationen werden bei ihren Business-Aktivitäten durch den [Business-Support](#) und die [Business-Services](#) unterstützt, die von [Applikationen](#) bereitgestellt werden.

Für die Klasse **Organisation** können [Objektklassenstereotype](#) konfiguriert werden.

## Organisationsgruppe

Eine Organisationsgruppe ist ein Container zur logischen Strukturierung von [Organisationen](#). Eine Organisation kann zu mehreren Organisationsgruppen zugeordnet sein.

## P

### Person

Siehe Anwender

### Personalanforderung

Mittels einer Personalanforderung wird die Anforderung des erforderlichen [Personalbedarfs](#) zur Ausführung eines bestimmten Projektaspekts durch ein [Projekt](#) definiert. In einer Personalanforderung können Sie die Art des erforderlichen Personals und die Personalstärke oder die Manntage, die für das Projekt erforderlich sind, sowie auch die [Organisationen](#), die für die Bereitstellung des Personals erforderlich sind und die für das Personal anfallenden Kosten definieren. Personalanforderungen können in [Personalanforderungsvorlagen](#) gebündelt und einer [Projektvorlage](#) zugeordnet werden, wodurch die Definition des für die Ausführung eines Projekts erforderlichen Personals vereinheitlicht wird.

### Personalanforderungsvorlage

Mit einer Personalanforderungsvorlage kann [Personalbedarf](#) festgelegt werden, um Ressourcen für ein [Projekt](#) zu planen. Eine Personalanforderungsvorlage enthält die für den Personalbedarf benötigte Anzahl der Mitarbeiter oder Manntage sowie die voraussichtlichen Kosten des Personalbedarfs im Kontext einer bestimmten Projektart. Bei einer [Personalanforderung](#) liegen dem Anwender daher vordefinierte Informationen vor, die bei der Einschätzung der Ressourcen und Kosten im Zusammenhang mit der Implementierung des Projektes helfen. Personalanforderungsvorlagen können einer [Projektvorlage](#) zugeordnet werden, um den Projektplanungsprozess zu rationalisieren.

### Personalangebot

Ein Personalangebot stellt das [Personal](#) dar, das von einer [Organisation](#) bereitgestellt werden kann. Die Organisation wurde festgelegt, um eine [Personalanforderung](#) für ein [Projekt](#) zu erfüllen.

### Personalbedarf

Anhand von Personalbedarf werden Fachkenntnisse beschrieben, die für die Ausführung eines bestimmten Aspekts eines [Projekts](#) erforderlich sind. ein Personalbedarf kann einer [Personalanforderungsvorlage](#) zugeordnete werden, durch die die Art des erforderlichen Personals sowie die Personalstärke oder die Anzahl

von für das Projekt erforderlichen Manntagen sowie die geschätzten Personalkosten definiert werden. Anwender, die eine [Personalanforderung](#) durchführen, erhalten dadurch vordefinierte Informationen, die ihnen dabei helfen, die Ressourcen und Kosten abzuschätzen, die aufgewendet werden müssen, wenn dieses Fachpersonal im Projekt eingesetzt wird.

Es muss nicht die gesamte für die Ausführung eines Projekts erforderliche Personalstärke geschätzt werden. Vielmehr sollte das Personal, das dem Unternehmen fehlt, bereits bei der Vorbereitung der Einreichung eines Projekts zur Priorisierung und Budgetierung projiziert werden.

## Perspektive

Eine Perspektive ist eine bestimmte Sicht, die auf ein [Steuerungselement](#) angewendet werden kann. Zu den typischen Perspektiven zählen "Innovation", "Prozess", "Kunde" und "Finanzen". Perspektiven werden üblicherweise dazu genutzt, die Vollständigkeit der Strategiedefinition eines Unternehmens oder einer untergeordneten Organisation zu überprüfen. Perspektiven dienen auch der Bestimmung der Ausrichtung zwischen Strategien, die von Steuerungselementen dargestellt werden. Eine Perspektive kann mehreren Steuerungselementen zugeordnet werden.

Wenn die Steuerungselemente einer [Organisation](#) analysiert werden, können die ermittelten und angestrebten [Messungen](#), die für die [Zielsetzungen](#) definiert sind, die dem Steuerungselement zugeordnet sind, im Hinblick auf die zugehörige Perspektive nachvollzogen werden. So wird eine Balanced-Scorecard-Analyse unterstützt, die die Umsetzung strategischer Absichten auf Basis von unternehmenskritischen Perspektiven nachvollziehbar macht.

## Pflichteigenschaft

Alle [Objektclassen](#) im Klassenmodell verfügen über vorkonfigurierte [Eigenschaften](#), die der semantischen Beschreibung der Objektclass dienen. Einige Standardeigenschaften sind Pflichteigenschaften und werden in Editoren mit einem roten Stern hervorgehoben.

Es ist kein Durchsetzungsmechanismus implementiert, der die Definition von Pflichteigenschaft erzwingt, es sei denn, die Eigenschaft ist Teil eines Standard- oder konfigurierten Klassenschlüssels, der die eindeutige Definition eines neuen Objekts durchsetzt. Wenn die Pflichteigenschaft in einem Klassenschlüssel enthalten ist, wird eine Fehlermeldung angezeigt, wenn die Eigenschaft nicht definiert ist.

## Physisches Gerät

Ein physisches Gerät ist ein [Gerät](#), das eine Hardware darstellt, auf der Software läuft. Ein physisches Gerät kann mit einem [logischen Gerät](#) mittels [Clustering](#) verbunden sein. Das Clustering beinhaltet verschiedenste Beziehungen zwischen physischen und logischen Geräten.

## Plattform

Eine Plattform beschreibt die technische Architektur im Hinblick auf die Zusammenstellung von [lokalen Komponenten](#), auf denen die spezielle [Applikation](#) oder [Komponente](#) läuft. Diese Zusammenstellung basiert auf einer [Plattformvorlage](#), die in einer Matrix-ähnlichen Struktur organisiert ist, bei der [Plattform-Layers](#)

und [Plattform-Tiers](#) die Achsen definieren. Die Struktur spiegelt die Unterteilung der Plattform im Hinblick auf die Verteilung über mehrere Hardware- und Softwaregruppen hinweg wider.

## Plattformelement

Ein Plattformelement ist Bestandteil einer [Plattform](#). Plattformelemente sind mit Standard-[Komponenten](#) oder [lokalen Komponenten](#) verbunden. Jeder qualitative Aspekt der lokalen Komponenten sowie die Informationen über [Business-Services](#), [Informationsflüsse](#) oder [Businessdaten](#) werden im Kontext einer Plattform nicht beachtet.

## Plattform-Layer

Ein Plattform-Layer ist eine Dimension der Partitionierung einer [Plattform](#). Plattform-Layers stellen eine Gruppierung von [Komponenten](#) auf Basis der Abstraktionsebene von Services dar, die die Komponenten innerhalb eines Layer bereitstellen.

Normalerweise ist ein Plattform-Layer von den darunter liegenden Layers abhängig. Eine typische Layer-Architektur umfasst einen Business-Layer, auf der sich die Business-Logik befindet, einen Software-Infrastruktur-Layer, auf dem sich das Betriebssystem und Infrastrukturservices befinden sowie einen Hardware-Layer. Plattform-Layers werden in einer [Plattformvorlage](#) mit [Plattform-Tiers](#) organisiert.

## Plattform-Tier

Ein Plattform-Tier ist eine Dimension der Partitionierung einer [Plattform](#). Plattform-Tiers stellen eine Gruppierung von [Komponenten](#) im Hinblick auf die Funktionalität dar, die in separaten physikalischen Maschinen installiert werden kann oder muss, um eine bessere Modularität und Skalierbarkeit zu erreichen.

Normalerweise kommuniziert jeder Tier nur mit den benachbarten Tiers. Eine typische 4-Tiers-Architektur umfasst einen Datenbankserver-Tier, einen Applikationsserver-Tier, einen Webserver-Tier und einen Client-Tier. Plattform-Tiers werden in einer [Plattformvorlage](#) mit [Plattform-Layers](#) organisiert.

## Plattformvorlage

Eine Plattformvorlage definiert einen Standardsatz an [Plattform-Layers](#) und [Plattform-Tiers](#). Dadurch wird ein Referenznetz zur Positionierung von [Komponenten](#) in [Standardplattformen](#) und [Plattformen](#) definiert.

Eine Plattformvorlage strukturiert die Plattform in zwei Dimensionen. Die horizontale Dimension stellt die Plattform-Tiers dar; eine Gruppierung von Komponenten in einer Plattform entsprechend der Funktionalitäten, die auf verschiedenen physikalischen Computern installiert sind. Die vertikale Dimension stellt die Plattform-Layers dar; eine Gruppierung von Komponenten, durch die die in einem Layer bereitgestellten Services klassifiziert werden.

Die Plattformvorlage kann auch eine strukturierte Beschreibung der Applikationsarchitektur anzeigen. Dies könnte zum Beispiel Folgendes umfassen:



- Der logische Layer der Architektur, der die Komponenten von Applikationen/Komponenten auf einer logischen oder einer Business-Ebene beschreibt.
- Der technische Layer, der die Implementierung oder Realisierung der Applikation/Komponente beschreibt.
- Der physikalische Layer, der die Laufzeitumgebung und die Zuweisung der Applikationen/Komponenten in Netzwerken und in verschiedenen Geräten beschreibt.

## Pool

Ein Pool ist ein Umrandungselement in einem [Servicediagramm](#), das als Rahmen für die Servicediagrammelemente dient, die einen einzelnen [Business-Prozess](#) darstellen. Ein Pool kann aus einer Vielzahl von [Swim-Lanes](#) bestehen, die die unterschiedlichen Agenten darstellen, die den Business-Prozess steuern, der im Service-Diagramm dargestellt wird. Die Servicediagrammelemente, die interne Aktivitäten darstellen, werden innerhalb der Poolgrenzen positioniert. Externe Aktivitäten oder [Ereignisse](#), wie z.B. ein Start- oder Endereignis, werden üblicherweise außerhalb des Pools positioniert.

## Portfolio

Ein Portfolio ist ein Blasendiagramm, in dem die relative Leistung einer Reihe von [Objekten](#), die derselben [Objektklasse](#) angehören, in zwei oder drei voneinander unabhängigen Dimensionen der Leistungsmessung dargestellt wird. Typische Portfolios enthalten eine Gruppe von [Applikationen](#), die Eigentum einer bestimmten [Organisation](#) sind, oder von einem bestimmten Prozess verwendet werden.

Das Portfolio besteht aus einer X-, Y- und ggf. einer Leistungsachse. Die Leistungsachse ist die Z-Dimension und wird durch die Blasengröße dargestellt. Einer Portfolio-Achse kann ein [Kennzahltyp](#), ein [Kennzahlensystem](#) oder ein [Priorisierungsschema](#) zugeordnet werden. In Alfabet kann ein Portfolio auch als BCG-Quadrant angezeigt werden. Der Zugriff auf Portfolios kann über [Anwenderprofile](#) gesteuert werden. Ein Anwender kann ein beliebiges Portfolio anzeigen, für das kein Anwenderprofil oder dasselbe Anwenderprofil definiert ist, mit dem dieser Anwender derzeit angemeldet ist.

## Primäres Objekt

Ein primäres Objekt ist ein [Architekturelement](#), das einer [Domäne](#) zugeordnet ist, wobei die Domäne der Eigentümer des Objekts ist und den Hauptfunktionskontext des Objekts definiert. In diesem Fall gilt die Domäne als die primäre Domäne des Objekts. Die Architekturelemente, die einer Domäne zugeordnet werden können, umfassen: [Business-Funktionen](#), [Business-Objekte](#), [Business-Prozesse](#), [Applikationen](#), [ICT-Objekte](#), [Komponenten](#), [Standardplattformen](#) und [Anbieterprodukte](#).

Eine Domäne kann über eine unbegrenzte Anzahl von primären Objekten verfügen, aber ein Objekt kann nur eine einzige primäre Domäne haben. Eine effektive serviceorientierte Architektur erfordert eine Reihe von getrennten primären Domänen in der Unternehmensarchitektur.

## Primäre Sprache

Die Primärsprache ist die Sprache der für Ihr Unternehmen angegebenen Basis-[Landeseinstellungen](#). Die Primärsprache ist die Standardsprache und wird angezeigt, wenn Anwender sich zum ersten Mal anmelden und die Alfabet -Benutzeroberfläche öffnen. Anwender können die Darstellung der Benutzeroberfläche bei Bedarf von der Primärsprache in eine [Sekundärsprache](#) ändern.

## Prinzip

Ein Prinzip ist eine umfassende Richtlinie zur Steuerung der Unternehmensarchitekturdefinition und der Entscheidungsfindungsprozesse im Unternehmen. Eine [Organisation](#) ist verantwortlich für das Prinzip. Ein Prinzip kann zwar über eine unbegrenzte Anzahl von untergeordneten Prinzipien verfügen, jedoch nur einem übergeordneten Prinzip zugeordnet sein.

## Priorisierungsschema

Ein Priorisierungsschema ist ein gewichteter Verbund aus [Kennzahlensystemen](#). Ein Priorisierungsschema wird üblicherweise genutzt, um die Rangfolge eines Satzes an Objekten festzulegen, zum Beispiel während der Budgetierung und Priorisierung von [Projekten](#) oder um die Achsen eines [Portfolios](#) zu definieren.

Priorisierungsschemata sollen [Bewertungen](#) über komplementäre Bewertungsdimensionen zusammenfassen. Während Kennzahlensystem Kennzahlen für semantisch zusammenhängende Leistungsmessung von Objekten zusammenfassen, fassen Priorisierungsschemata semantisch nicht (oder wenig) zusammenhängende Leistungsmessungen von Objekten zusammen. Priorisierungsschemata, die einem Portfolio zugeordnet sind, beschreiben jedoch üblicherweise die Leistung in Abhängigkeit von zusammenhängenden Dimensionen.

## Profil

Siehe [Objektprofil](#) oder [Anwenderprofil](#).

## Projekt

Ein Projekt stellt eine Aktivität dar, die ausgeführt wird, um ein bestimmtes Ziel in der IT-Landschaft zu erreichen. In der Regel basieren Projekte auf konfigurierten [Projektstereotypen](#). Ein Projekt kann zwar über eine unbegrenzte Anzahl untergeordneter Projekte (Projekte des untergeordneten Projektstereotyps), jedoch nur über ein übergeordnetes Projekt (ein Projekt des übergeordneten Projektstereotyps) verfügen. Projekte können in [Projektgruppen](#) gebündelt und zu Budgetierungs- und Priorisierungszwecken [Buckets](#) zugeordnet werden.

Je nach der für ein Projektstereotyp definierten Konfiguration ist es möglich, [Anforderungen](#) in Projekten zu bündeln, die vom Projekt betroffene [Ist-Architektur](#) zu dokumentieren, die [Soll-Architektur](#) für die IT-Landschaft zu planen, die Projektkosten in einem [Business-Case](#) zu planen und zu bewerten, die [Personalanforderungen](#) und [Ressourcenanfragen](#) für das Projekt zu erfassen und den entsprechenden [Projektressourcenplan](#) zu verwalten sowie die Projektzieldaten mithilfe von [Meilensteinen](#) zu überwachen.

Zudem können Sie mithilfe von [Projekt-Szenarios](#) auf Grundlage eines bestehenden Projekts alternative "Was-wäre-wenn"-Szenarios im Hinblick auf Kosten, Ressourcen, Zeitplanung und Soll-Architektur planen. Schließlich kann für ein Projekt eine [Projektausgangsbasis](#) erstellt werden, um den Umfang der in einem Projekt eingetretenen Änderungen nachvollziehen zu können.

## Projektausgangsbasis

Eine Projektausgangsbasis deckt den Umfang des Basis-[Projekts](#) zum Zeitpunkt ihrer Erstellung ab. Sie bietet eine schreibgeschützte Ansicht der Daten des Basisprojekts und ermöglicht die zeitliche Nachverfolgung von Abweichungen des Basisprojekts vom ursprünglichen Projektrahmen im Hinblick auf die Gesamtprojektkosten und -einnahmen, die Kosten für [Personalanforderungen](#) und [Ressourcenanfragen](#), [Bewertungen](#), Zeitplanung und das Erreichen von [Projektmeilensteinen](#) sowie Änderungen an der [Soll-Architektur](#).

## Projektgruppe

Eine Projektgruppe ist ein Container zur logischen Strukturierung von [Projekten](#). Es gibt verschiedene Möglichkeiten, Projekte logisch zu strukturieren. Deshalb kann ein Projekt mehreren Projektgruppen zugeordnet sein. Projektgruppen unterstützen die Budgetierung und Priorisierung der Projekte in der Gruppe.

## Projektkennzahlensystem

Ein Projektkennzahlensystem ist ein [Kennzahlensystem](#), das konfiguriert wurde, um die Überwachung von [Projekten](#) zu ermöglichen. Projekte können bezüglich der Erreichung von Zielwerten für die [Kennzahltypen](#), die dem Projektkennzahlensystem zugeordnet sind, überwacht werden.

## Projektlösung

Ein Projektlösung ermöglicht es, alternative [Projekte](#) zu entwerfen und gegeneinander abzuwägen. Es können zwar mehrere Projektlösungen für ein bestimmtes Projekt definiert werden, jedoch kann nur eine Projektlösung als genehmigte Projektlösung ausgewählt werden. Sobald eine Projektlösung ausgewählt wurde, gilt diese selbst als Projekt und ersetzt das ursprüngliche Projekt, für das sie eingangs in der Projekthierarchie definiert wurde. [Projektszenarien](#) stellen eine Erweiterung des bestehenden Projektlösungskonzepts dar, bieten jedoch mehr Flexibilität und Funktionen bei der Planung und Implementierung von "Was-wäre-wenn"-Szenarien.

## Projektressourcenplan

Ein Projektressourcenplan ist die detaillierte Planung und Zeitplanung für die [Personalanforderungen](#) und [Ressourcenanforderungen](#) für ein [Projekt](#). Im Unterschied zur Projektressourcenplanung auf den höheren Ebenen der Projektplanung, die sich auf den allgemeinen Umfang der für das Projekt erforderlichen [Personalanforderungen](#) und [Ressourcenanforderungen](#) konzentriert, hat der Projektressourcenplan das Management der Ressourcen zum Ziel, einschließlich der Personen und [Organisationen](#), die für die Durchführung

der zugehörigen Aufgaben verantwortlich sind, sowie der Gesamtkosten und der Zeitplanung für den erforderlichen [Personalbedarf](#) und die erforderlichen [Ressourcen](#).

## Projektstereotyp

Ein Projektstereotyp stellt eine Ebene im Projekt-Management-Framework dar. Die Projekthierarchie besteht aus einer beliebigen Anzahl von [Objektclassenstereotypen](#), die für die Klasse `Project` definiert sind. In Alfabet kann für jeden Projektstereotyp eine unbegrenzte Anzahl von Objekten erzeugt werden.

Die Projekthierarchie kann als lineare Hierarchie oder als verzweigte Baumstruktur konfiguriert werden, mit der komplexe Projektstrukturen und Hierarchien von Stereotypen erfasst werden können. Jeder Projektstereotyp kann mehrere untergeordnete Projektstereotypen definieren. Der Projektstereotyp auf der obersten Ebene stellt die abstrakteste Ebene eines [Projekts](#) dar, während der Projektstereotyp der untersten Ebene die granularste Ebene der Projektaufgaben repräsentiert, die ausgeführt werden müssen, um das Projekt zu realisieren. Ein typisches lineares Projekt könnte beispielsweise aus folgenden Projektstereotypen bestehen: Ebene 1: Programm, Ebene 2: Leistungsbeschreibung, Ebene 3: Projekt, Ebene 4: Sub-Projekt, Ebene 5: Projektaktivität.

Der Name, die Anzahl und die hierarchische Struktur der Projektstereotypen sind konfigurierbar. Ein Projektstereotyp ist mit einer Objektklasse vergleichbar. Wie bei Objektklassen sind die auf einem Projektstereotyp basierenden Objekte Instanzen des Projektstereotyps. Der Projektstereotyp "Projektaktivität" verfügt über Objekte, die als "Projektaktivität" bezeichnet werden. Jeder Projektstereotyp darf nur jeweils einen übergeordneten und einen untergeordneten Projektstereotyp haben. Projekte eines bestimmten Stereotyps dürfen aber weder zu einem Projekt des übergeordneten Projektstereotyps noch zu Projekten des untergeordneten Projektstereotyps zugeordnet werden.

Die Definitionen, die für jeden Projektstereotyp vorgenommen werden können, müssen konfiguriert werden. Deshalb können die verfügbaren Ansichten in der Hierarchie von Projektstereotyp zu Projektstereotyp variieren. Beispielsweise wird durch die Konfiguration bestimmt, für welchen Projektstereotyp die Anwender [Anforderungen](#) in Projekten bündeln, die [Ist-Architektur](#) dokumentieren, eine [Soll-Architektur](#) planen, alternative [Projektszenarios](#) bewerten, Kosten über [Kostenstellen](#) verfolgen und Projekte mittels [Meilensteinen](#) überwachen können.

## Projektszenario

Mithilfe von Projektszenarios können alternative "Was-wäre-wenn"-Szenarios für das Projekt untersucht, geplant und potenziell implementiert werden. Das Projektszenario kann alternative [Soll-Architekturen](#), Projektkosten und -einnahmen, den Umfang der für das Projekt erforderlichen [Personalanforderungen](#) und [Ressourcenanfragen](#), unterschiedliche [Bewertungen](#) verschiedener Aspekte des Projekts sowie unterschiedliche Versionen von Projektabhängigkeiten und Zeitplänen betrachten.

Das Projektszenario ist eine unabhängige Kopie des [Projekts](#) und beinhaltet dessen Start- und Enddatum, [benutzerdefinierte Eigenschaften](#) sowie den [Business-Case](#), die [Ist-Architektur](#), die [Soll-Architektur](#), die Personalanforderungen und Ressourcenanfragen, die [Steuerungselemente](#), die [Vertragsleistungen](#) und die [Input-Anforderungen](#) des Basisprojekts. Zudem werden alle untergeordneten Projekte kopiert.

Jedes Projektszenario kann weiter definiert und geändert werden, ohne dass sich dies auf das Basisprojekt auswirkt. Das Projektszenario kann jederzeit mit dem Basisprojekt zusammengeführt werden, um dieses zu aktualisieren. Das Projektszenario und das Basisprojekt bestehen weiter und können bei Bedarf weiter modifiziert werden.

## Projektvorlage

Eine Projektvorlage besteht aus einer konfigurierten [Meilensteinvorlage](#), die eine Reihe von [Meilensteinen](#) bündelt, und einer Vorlage für [Personalanforderungen](#), die den [Personalbedarf](#) gruppiert. Einem [Projekt](#) und dessen untergeordneten Projekten kann eine Projektvorlage zugeordnet werden, damit die Projekthierarchie über eine konsistente Definition von Meilensteinen und [Personalanforderungen](#) verfügt.

## Q

## R

### Referenzmodell

Ein Referenzmodell ist eine hierarchisch strukturierte Sammlung von [Framework-Gruppen](#) und dient der Definition einer semantischen Struktur für [Objekte](#), die in Alfabet ungeachtet der [Objektklasse](#), zu der sie gehören, verwaltet werden. In Alfabet können die Anwender durch ein Referenzmodell navigieren.

### Regel

Eine Regel ist eine Bedingung, die beschreibt, wie [Business-Prozesse](#) miteinander verbunden sind. Regeln werden in Business-Prozessdiagrammen angezeigt. Zu den verfügbaren Regeltypen zählen AND, OR, XOR, Synchronisator, Connector.

### Regelbasierte Zugriffsberechtigung

Eine regelbasierte Zugriffsberechtigung ist eine Regel, die auf einer Alfabet-Abfrage oder nativen SQL-Abfrage basiert und die Zugriffsberechtigungen für gefundene Objekte oder Objektklassen spezifiziert.


### Release

Siehe [Stack](#).

### Release-Status

Ein Release-Status beschreibt den Status der Genehmigung für oder einer Vereinbarung über ein [Objekt](#) im Unternehmen. Typische Statuswerte sind beispielsweise *Draft*, *Described*, *Reviewed*, *Approved*, *Rejected* und *Retired*. Die Release-Status hängen jedoch größtenteils von der Objektklasse ab, die sie beschreiben. Beispielsweise werden die Release-Status für ein [Architekturelement](#) wie eine Applikation, Komponente oder Standardplattform üblicherweise genutzt, um das Einverständnis mit dem Status der

dokumentierten Informationen auszudrücken, wohingegen die Release-Status für ein Planungsartefakt, wie eine Anforderung oder ein Projekt, einen Genehmigungsstatus darstellen. Der Release-Status für eine [Aufgabe](#) beschreibt den Fortschritt oder die Fertigstellung der jeweiligen Aufgabe.

Der Release-Status eines Objekts kann von einem Anwender geändert werden, der über Schreib- und Lesezugriffsberechtigungen für das Objekt verfügt. Bei allen Release-Status, die als nicht bearbeitbare Release-Status konfiguriert sind, kann der Anwender die Attribute im Editor des Objekts und auf den Ansichtsseiten des Objekts nicht bearbeiten. Die Anwender sehen oben im [Objektprofil](#) eines Objekts, das aufgrund seiner aktuellen Release-Status-Definition nicht bearbeitet werden kann, das  Symbol.

## Ressource

Eine Ressource ist jede Entität, die als Supportquelle für eine andere Entität im Unternehmen benötigt wird. Eine Ressource kann durch viele verschiedene Objektklassen zur Verfügung gestellt werden, wie zum Beispiel durch eine [Organisation](#), eine [Applikation](#), ein [Gerät](#), eine [Installation](#) und so weiter. Eine Ressource wird üblicherweise von einem [Projekt](#), einem [Service-Produkt](#) oder einer Organisation angefordert.

## Ressourcenanforderung

Eine Ressourcenanforderung erfolgt üblicherweise durch ein [Projekt](#), eine [Applikation](#), eine [Komponente](#), ein [Gerät](#), eine [Installation](#), ein [Service-Produkt](#), eine [Standardplattform](#) oder eine [Organisation](#), um den Bedarf an einer [Ressource](#) zu decken. Üblicherweise werden für Ressourcenanforderungen [Objektklassens-tereotypen](#) konfiguriert, um zwischen verschiedenen Typen an Ressourcen zu unterscheiden, die auf verschiedenen Anbietertypen basieren, welche zum Beispiel Applikationen, Komponenten, Geräte, Installationen, Service-Produkte, Standardplattformen oder Organisationen sein können. Eine Ressource ist einer [Kostenart](#) zugeordnet und umfasst Angaben über die für die Ressource erforderliche Menge sowie die Gesamtkosten für die angeforderte Ressource.

## Richtlinie

Ein Richtlinie ist eine Leitlinie oder Business- oder Verhaltensregel, die vom Unternehmen und seinen Bestandteilen eingehalten werden muss oder sollte. Eine Richtlinien kann mit einem [Steuerungselement](#) oder einem [Prinzip](#) verbunden sein. Die Richtlinie hat ein Start- und Enddatum und kann ein Überprüfungsdatum haben.

## Risiko

Ein Risiko bezieht sich auf die Wahrscheinlichkeit mit der ein bestimmtes Objekt in der IT-Landschaft des Unternehmens einer vorhandenen [Bedrohung](#) ausgesetzt ist, und Ziel eines potentiellen Angriff werden kann. Die Definition eines Risikos umfasst die Bewertung des potentiell durch das Risiko verursachten Schadens sowie die potentielle Wahrscheinlichkeit, ob das Risiko eintritt. Es kann eine [Risikominderung](#) definiert werden, in der beschrieben wird, wie das Risiko reduziert oder vermieden werden kann.

Ein Risiko kann explizit für ein bestimmtes [Risiko-Objekt](#) definiert werden oder über eine konfigurierte [Risikovorlage](#), in der Standardrisiken und eventuell auch deren vorgeschlagene Minderungen gebündelt sind, zu dem Risiko-Objekt hinzugefügt werden.

## Risikomanagementgruppe

Eine Risikomanagementgruppe ist ein Container zur logischen Strukturierung von [Risiko-Objekten](#) zum Anzeigen, Analysieren und Kommunizieren von [Risiken](#) für das Unternehmen. Jedes Objekt, das einer Risikomanagementgruppe zugeordnet ist, wird als Risiko-Objekt betrachtet. Es handelt sich um das Risiko-Objekt, das im Rahmen des Risikomanagements bewertet wird. Dadurch wird sichergestellt, dass diese Informationen nur für Anwender in der Anwendergemeinschaft sichtbar sind, die Zugriff auf das Risiko-Objekt haben. Ein Objekt kann zwar nur einmal zu einer ausgewählten Risikomanagementgruppe hinzugefügt, aber mehreren Risikomanagementgruppen zugeordnet werden, wodurch dasselbe Risiko-Objekt in unterschiedlichen Kontexten berücksichtigt werden kann.

## Risikomanagementvorlage

Die Risikobeurteilung in Alfabet besteht aus zwei Phasen. Eine erste Phase ist die **Risikobeurteilungsphase** und ermöglicht es dem Unternehmen, zu beurteilen, welche Objekte potenziell gefährdet sind. Mit dieser Phase können Objekte beurteilt werden, um zu bestimmen, ob sie in die Risikobewertung einbezogen werden sollen. Die zweite Phase ist die **Risikobewertungsphase** und ermöglicht es dem Unternehmen, das Risiko für die durch die Risikobewertung vorgesehenen Objekte zu dokumentieren und zu bewerten. Dies umfasst die potenziellen Schadenskosten des Objekts und die Wahrscheinlichkeit einer Beschädigung des Objekts sowie Mittel zur Risikominimierung. Dieser Ansatz ermöglicht es dem Unternehmen, den Risikobewertungsprozess zu rationalisieren und gezielt die Objekte anzusprechen, die am stärksten gefährdet sind.

## Risikominderung

Die Risikominderung formuliert einen Schritt zur Vermeidung, Reduzierung oder Eindämmung des [Risikos](#). Die Risikominderung kann auf einer vorkonfigurierten [Risikominderungsvorlage](#) basieren. Für jede Risikominderung kann dokumentiert werden, welches [Architekturelement](#) von ihr betroffen sein könnte. Außerdem kann eine [Anforderung](#) erzeugt werden, um die Notwendigkeit, die Risikominderung in der IT-Architektur umzusetzen, auszudrücken. Sobald die Anforderung, die Risikominderung umzusetzen, gestellt wurde, kann ein [Projekt](#) erzeugt werden, um die Risikominderung zu implementieren.

## Risikominderungsvorlage

Bei einer Risikominderungsvorlage handelt es sich um die vorkonfigurierte Definition einer [Risikominderung](#) mit dem Namen der Risikominderung, dem Zieldatum, zu dem die Risikominderung implementiert sein muss, und mit einer Zahl, die die Priorität der Risikominderung wiedergibt. Eine Risikominderungsvorlage kann für eine bestimmte [Bedrohung](#), eine Bedrohungsgruppe oder eine [Risikominderungsvorlagen-Kategorie](#) erstellt werden. Jede Risikominderungsvorlage kann genau einer Risikominderungsvorlagen-Kategorie zugeordnet werden.

## Risikominderungsvorlagenkategorie

In einer Risikominderungsvorlagenkategorie werden [Risikominderungsvorlagen](#) zusammengefasst und inhaltlich klassifiziert. In Risikominderungsvorlagenkategorien können Sie die Risikominderungsvorlagen in Ihrem Unternehmen auch hierarchisch strukturieren. Jede Risikominderungsvorlage kann nur einer Risikominderungsvorlagenkategorie zugeordnet werden.

## Risiko-Objekt

Ein Risiko-Objekt referenziert ein Objekt, für das ermittelt wurde, dass es ein bestimmtes [Basis-Gefährdungspotential](#) überschreitet, und das daher im Kontext der Geschäftsfähigkeit "Risikomanagement" hinsichtlich des Schadensrisikos und der Risikowahrscheinlichkeit bewertet werden muss.

Für jedes Risiko-Objekt müssen als Antworten auf die in dem konfigurierten Risiko-Fragebogen gestellten Fragen Werte bereitgestellt werden. Ferner können potentielle [Risiken](#) einschließlich ihres potentiellen Schadens, ihrer Eintrittswahrscheinlichkeit sowie eine Minderung zur Reduktion oder Vermeidung des Risikos definiert werden.

Risiko-Objekte können in einer [Risikomanagementgruppe](#) strukturiert werden, um deren Risiko nachvollziehen und aus verschiedenen Perspektiven bewerten zu können. Ein Risiko-Objekt kann immer nur zu einer Risikomanagementgruppe zugeordnet werden. Ein Objekt (z.B. eine Applikation) kann jedoch von zwei unterschiedlichen Risiken-Objekten referenziert werden, die zu unterschiedlichen Risikomanagementgruppen gehören, wodurch eine Analyse potentieller Risiken aus mehreren Perspektiven möglich wird.

## Risikovorlage

In einer Risikovorlage werden mehrere [Risiken](#) zusammengefasst und, falls erforderlich, auch deren vorgeschlagene Minderungen. Einer [klassenbasierten Risikomanagementvorlage](#) können mehrere Risikovorlagen zugeordnet werden.

Wenn ein [Risiko-Objekt](#) beurteilt wird, kann die Risikovorlage ausgewählt werden, um dem Risiko-Objekt eine Gruppe von Standardrisiken und, falls zutreffend, deren Minderungen hinzuzufügen. Irrelevante Risiken können aus dem Risiko-Objekt entfernt werden und zusätzliche Risiken, die eventuell nicht in der Risikovorlage erfasst wurden, können für das Risiko-Objekt definiert werden.

## Rolle

Über eine Rolle wird die funktionale Beziehung oder Verantwortlichkeit eines Anwenders oder einer [Organisation](#) zu einem [Objekt](#) definiert. Eine Rolle basiert auf einem für eine [Objektklasse](#) konfigurierten [Rollentyp](#). Rollen werden nur zu Informationszwecken definiert und bieten Detailinformationen über Anwender oder Organisationen, die wiederum über Informationen über das Objekt verfügen oder daran interessiert sind. Die Definition einer Rolle für ein Objekt hat daher keinerlei Auswirkung auf Zugriffsberechtigungen.



## Rollentyp

Ein Rollentyp wird konfiguriert, um eine funktionale [Rolle](#) für Objekte in einer bestimmten [Objektklasse](#) zu definieren. Rollentypen können so konfiguriert werden, dass sie explizit für einen Anwender oder eine [Organisation](#) oder für beides verfügbar sind.

## Root-Objekt

Ein Root-Objekt ist ein [Objekt](#), das für die oberste Ebene einer hierarchischen Explorerbaumstruktur definiert wurde. Root-Objekte können in den Hierarchien gefunden werden, die mit [Organisationen](#), [Business-Prozessmodellen](#), [Business-Funktionen](#) oder [Applikationsgruppen](#) verbunden sind.

## S

## Sachverhalt

Bei einem Sachverhalt handelt es sich um ein Thema, das für ein bestimmtes Objekt in Alfabet behandelt werden muss. Für das Objekt können ein Sachverhaltstyp sowie die Architekturelemente definiert werden, die je nach definiertem Sachverhaltstyp von dem Sachverhalt betroffen sein können. Für jeden Sachverhalt können Priorität sowie Status und geplantes Fertigstellungsdatum definiert werden. Mit dem Sachverhalt können zudem weitere relevante Sachverhalte verknüpft werden. Von einem Sachverhalt kann eine [Anforderung](#) abgeleitet werden. Zudem können Sachverhalte mit [Funktionen](#) verknüpft werden, die implementiert werden sollen, um verknüpfte Anforderungen zu erfüllen.

## Schnittstelle

Eine Schnittstelle ist die technische [Komponente](#), die mit einem [Informationsfluss](#) verbunden ist, und die für den Austausch von Daten erforderliche technische Umwandlung umsetzt. Die Quell- und Zielschnittstellen müssen als Komponenten oder [lokale Komponenten](#) in der Plattform der entsprechenden [Applikation](#) definiert werden. Ein Informationsfluss kann über Quell- und Zielschnittstellen sowie auch über [Schnittstellensysteme](#) verfügen.

## Schnittstellensystem

Ein Schnittstellensystem ist eine Sammlung von [Architekturelementen](#), die sich zwischen den Quell- und Ziel-[Schnittstellen](#) eines [Informationsflusses](#) befindet und [Business-Daten](#) umwandeln oder [Services](#) aufrufen kann. In vielen Fällen erfolgt der Datenaustausch über ein bestimmtes Kommunikationssystem, wie z.B. ein EAI-Bus, der von einem Informationsfluss modelliert werden kann. Das System als solches wird als Schnittstellensystem bezeichnet. EAI-Plattformen, Meldungswarteschlangen, Anforderungsvermittler, Bus-Systeme oder Service-Orchestrierung sind typische Beispiele für Schnittstellensysteme.

Ein Schnittstellensystem kann sowohl durch [Applikationen](#) und [Komponenten](#) dargestellt werden. Ein Schnittstellensystem kann eigenen [Business-Support](#) bereitstellen. In diesem Fall sollte es durch eine

Applikation dargestellt werden. Oder es stellt den technischen Support für den Informationsaustausch bereit. In diesem Fall könnte es als Komponente modelliert werden. Die von einem Schnittstellensystem aufgerufenen technischen Services können nach Schnittstellensystemen spezifiziert werden, die von einer Komponente oder lokalen Komponente des Typs **Service** abgeleitet sind. Bei den für einen Informationsfluss angegebenen Schnittstellensystemen kann die Reihenfolge festgelegt werden, in der die Services aufgerufen werden.

## Sekundärsprache

Eine Sekundärsprache ist jede Sprache, die nicht die [Primärsprache](#) ist und für das Unternehmen konfiguriert wurde. In einer Sekundärsprache können die Benutzeroberfläche dargestellt und Objektdaten erfasst werden. Für die Alfabet -Lösung können mehrere Sekundärsprachen konfiguriert werden.

## Sequenzfluss

Ein Sequenzfluss ist ein Konnektor in einem [Service-Diagramm](#), der eine [Aktivität](#) und ein [Ereignis](#) in derselben [Swim-Lane](#) miteinander verbindet. Ein Sequenzfluss gibt die Reihenfolge an, in der die Aktivitäten ausgeführt werden. Das heißt, dass eine Aktivität mit ihrer Vorgänger-Aktivität und ihrer Nachfolge-Aktivität über einen Sequenzfluss verbunden ist.

## Service-Diagramm

Ein Service-Diagramm ist ein Diagramm, das einem [Business-Prozess](#) zugeordnet ist, und die Abhängigkeiten zwischen dessen Subprozessen (Aktivitäten) oder zwischen den [Business-Services](#), die durch den Business-Prozess angefordert werden, anzeigt. Ein Business-Prozess kann über mehrere Service-Diagramme verfügen. Ein Service-Diagramm gehört jedoch nur zu einem Business-Prozess.

Ein Service-Diagramm kann Diagrammelemente des Typs [Ereignis](#) oder [Aktivität](#), einfache Verbindungselemente des Typs [Sequenzfluss](#), [Meldungsfluss](#) oder [Verbindung](#), komplexe Verbindungselemente des Typs [Gateway](#) sowie Umrandungselemente des Typs [Pool](#) oder [Swim-Lane](#) enthalten. Ferner können auch [Business-Objekte](#), Business-Services und [Business-Funktionen](#) in einem Service-Diagramm angezeigt werden. Des Weiteren können Business-Services und Business-Funktionen mit sonstigen Service-Diagrammelementen verbunden werden.

Ein Service-Diagramm kann mittels einer Verknüpfung, die einer Aktivität zugeordnet ist, auf ein anderes Service-Diagramm verweisen. Service-Diagramme können nur auf Ebene der Business-Prozesse angezeigt werden.

Der Entwurf von Service-Diagrammen in Alfabet erfolgt gemäß BPMN-Konventionen.

## Servicelevel-Vereinbarung (SLA)

Eine SLA ist eine Servicelevel-Vereinbarung, die für ein [Service-Produkt](#) definiert ist. Die Servicelevel-Vereinbarung beinhaltet Informationen wie zum Beispiel eine Mengenangabe, die für das Service-Produkt relevant ist (wie zum Beispiel Service-Desk-Supportzeiten, maximale Wartezeit auf Störungsbehebung, maximale Testfehlerhäufigkeit, maximale Größe von Log-Ausgaben und so weiter). Einem Service-Produkt kann

nur eine Servicelevel-Vereinbarung (SLA) zugeordnet sein. Die Servicelevel-Vereinbarung übernimmt automatisch die Definition des [autorisierten Anwenders](#) von dem Service-Produkt, dem sie zugeordnet ist.

Für die Objektklasse Servicelevel-Vereinbarung können [Objektklassenstereotypen](#) konfiguriert werden. Dadurch können für verschiedene Typen von Servicelevel-Vereinbarungen verschiedene Attribute verwendet werden.

## Service-Produkt

Ein Service-Produkt ist ein Service, den eine [Organisation](#) besitzt und anderen Einheiten zur Verfügung stellt. Ein Service-Produkt besteht aus einem oder mehreren [Service-Produktelementen](#), die das Service-Produkt bilden. Jedes Service-Produktelement ist mit einem Objekt, wie zum Beispiel [Applikation](#), [Komponente](#), [ICT-Objekt](#), [Installation](#), [Gerät](#), [Standardplattform](#), Organisation oder einem anderen Service-Produkt, verbunden, dass das Service-Produktelement für das Service-Produkt bereitstellt.

Jedem Service-Produkt kann eine [Servicelevel-Vereinbarung \(SLA\)](#) zugeordnet sein.

Service-Produkte sind nach den Eigentümerorganisationen geordnet. Eine Organisation und die Service-Produkte, die sie besitzt, bilden einen Service-Produktkatalog. Zusätzlich können Service-Produkte unabhängig von den Eigentümerorganisationen, die die Service-Produkte bereitstellen, in Service-Produktgruppen eingeordnet werden.

Ein Service-Produkt kann als [Vertragsleistung](#) für einen [Vertrag](#) definiert werden, wobei das Service-Produkt durch den Vertrag genutzt wird. Ein Service-Produkt kann einem Marktprodukt zugeordnet werden, um anzuzeigen, dass das Service-Produkt über das [Marktprodukt](#) verfügbar ist.

## Service-Produktelement

Ein Service-Produktelement ist eine Einheit, die ein [Service-Produkt](#) unterstützt. Ein Service-Produkt kann aus einem oder mehreren Service-Produktelementen gebildet werden. Jedes Service-Produktelement ist mit einem Objekt einer zugelassenen Objektklasse (wie zum Beispiel [Applikation](#), [Komponente](#), [ICT-Objekt](#), [Installation](#), [Gerät](#), [Standardplattform](#), [Organisation](#) oder ein anderes Service-Produkt) verbunden, dass das Service-Produktelement für das Service-Produkt bereitstellt.

Für ein Service-Produktelement kann ein Name definiert werden, um den Zweck des mit dem Service-Produktelement verbundenen Objekts zu erfassen. Zum Beispiel kann das Service-Produkt "Anwender-Support-Service" aus den Service-Produktelementen "Support-Service-Applikationen", "Support-Service-Installationen", und "Support-Service-Geräte" bestehen, die die Applikationen, Installationen und Geräte, über die die Support-Services geleistet werden, repräsentieren.

Das Service-Produktelement kann auch eine Volumendefinition beinhalten, um ein Leistungsverzeichnis darstellen zu können (zum Beispiel, wenn für ein Service-Produkt zwei Instanzen eines bestimmten Servers benötigt werden, um eine hohe Verfügbarkeit zu gewährleisten).

Das Service-Produktelement übernimmt den [autorisierten Anwender](#) und/oder die [autorisierten Anwendergruppen](#) des Service-Produkts, dem es zugeordnet ist.

Für die Klasse **Service-Produktelement** können [Objektklassenstereotype](#) konfiguriert werden.

## Servicevertrag

Ein Servicevertrag erfasst die Gebühren, die eine [Organisation](#) zahlt, um Services bereitzustellen. Die Servicegebühren sind die Kosten für die Nutzung der [Applikationen](#) und [ICT-Objekte](#), die für die Bereitstellung der Services erforderlich sind. Ein Servicevertrag basiert auf einer [Servicevertragsvorlage](#).

## Servicevertragsvorlage

Eine Servicevertragsvorlage definiert die Grundlage für die Preisgestaltung von [Serviceverträgen](#). Der Preis für die Servicekosten eines Servicevertrags wird wie folgt berechnet: Einmalige Gebühr\*Erstmalige Verwendung + Basispreis + Nutzungspreis\*Nutzung. Die Werte für **Einmalige Gebühr**, **Basispreis** und **Nutzungspreis** sind in der Servicevertragsvorlage vordefiniert, auf der der Servicevertrag basiert.

## Soll-Architektur

Die Soll-Architektur ist ein mittelfristiger Plan für die Unternehmensarchitektur und definiert eine klare, genehmigte strategische Richtlinie. Die geplante, konsolidierte Soll-Architektur wird in einem oder mehreren [Bebauungsplänen](#) erfasst, durch die der geplante [taktische Business-Support](#) definiert wird. Die Realisierung der Sollarchitektur erfolgt im Rahmen von [Projekten](#).

Die Soll-Architektur eines Projekts enthält üblicherweise [Inventory-Objekte](#) der [Ist-Architektur](#), die u. U. von den vorgeschlagenen Architekturänderungen betroffen sind, sowie neue [Lösungs-Objekte](#), wie [Lösungsapplikationen](#), [Lösungs-Komponenten](#), [Lösungs-Geräte](#), [Lösungs-Informationsflüsse](#), [lokale Lösungskomponenten](#), [externe Lösungssysteme](#), [Lösungs-Standardplattformen](#), [technische Lösungsservices](#) und [Lösungs-Business-Supports](#), die im Kontext der Soll-Architektur erzeugt wurden. Die Lösungsobjekte werden als „Schattenobjekte“ betrachtet, d. h. Kopien der Basis[applikationen](#), [Informationsflüsse](#) usw., von denen sie abgeleitet wurden. Die Lösungsobjekte existieren nur im Rahmen der Soll-Architektur-Definition und können im Kontext der Soll-Architektur geändert werden, ohne dass die Inventory-Objekte, auf denen Sie basieren, davon betroffen sind. Die im Kontext der vorgeschlagenen Soll-Architektur definierten Architekturänderungen können in das Inventory eingechekkt werden, wobei die Lösungsobjekte, die auf Inventory-Objekten basieren, die entsprechenden Inventory-Objekte überschreiben. Alle [Architekturelemente](#), die dem Inventory mittels der vorgeschlagenen Soll-Architektur hinzugefügt werden, haben einen [Objektstatus](#), der angibt, dass das Objekt geplant ist. Der Objektstatus muss manuell in einen aktiven Objektstatus geändert werden, wenn die operative Nutzung der Soll-Architekturelemente beginnt (meist bei Fertigstellung der Implementierung des Projekts).

## Stack

Ein Stack ist eine operative Präzisierung einer [Applikation](#), [Komponente](#) oder [Standardplattform](#), die für die strategische Planung verwendet wird. Ein Stack stellt die physikalisch installierte Infrastruktur dar, die in einer [Installation](#) definiert ist. Stacks können hierarchisch organisiert werden und können mehrere Unter-Stacks umfassen, wodurch verschiedene Ebenen der Operationalisierung zur Verfügung stehen.

Auf jeder Präzisierungsebene ist die Variabilität des Stacks weiter eingeschränkt. Wenn beispielsweise ein Stack der ersten Ebene bei der Betriebssystemkompatibilität zwischen UNIX und Windows im Allgemeinen unterscheidet, dann unterscheiden die Stacks der nächsten Ebene zwischen verschiedenen Unix-Dialekten oder Betriebssystemversionen.

Für Komponenten, die auf der Basis von [Anbieterprodukten](#) erzeugt werden, geben die Stacks typischerweise die Release-Versionen des [Anbieters](#) auf Patch-Ebene an. Für Applikationen werden die Stacks als eine Präzisierung der [Plattform](#) der Applikation definiert. Die der Plattform untergeordneten Stacks können Elemente ausschließen, die in der Plattform verwendet worden sind, und somit komplexe Gruppenabhängigkeiten zwischen den Elemente wiedergeben, die in der [Plattform-Tier - Plattform-Layer](#)-Matrix der Plattform positioniert wurden. Außerdem können Stacks [Stack-Konfigurationselemente](#) als neue Elemente einführen, die keine Relevanz für die strategische Ebene des Plans haben, jedoch für die verschiedenen Ebenen der Operationalisierung von Bedeutung sind.

## Stack-Element

Ein Stack-Element ist Bestandteil eines [Stacks](#). Jedes Stack-Element verweist entweder auf ein Element der zugehörigen [Plattform](#) oder ist ein [Stack-Konfigurationselement](#). Ein Stack-element erbt den [autorisierten Anwender](#) und/oder die [autorisierten Anwendergruppen](#) von dem Stack, von dem es abgeleitet ist.

## Stack-Konfigurationselement

Ein Stack-Konfigurationselement ist ein technischer Aspekt einer [Applikation](#), [Komponente](#) oder einer [Standardplattform](#), das zwar für strategische Planungszwecke unwichtig, aber relevant für die operative und Release-Planung sowie die physische Installation ist.

Stack-Konfigurationselemente sind Teil der technischen Beschreibung eines [Stacks](#) und entweder einer Applikation oder einer Komponente oder Standardplattform zugeordnet. Stack-Konfigurationselemente selbst können auf eine Komponente verweisen. Zu den typischen Beispielen für Stack-Konfigurationselemente gehören Install-Shields, Installations-Skripte, Aufgabenwarteschlangen usw.

## Stack-Upgrade

Ein Stack-Upgrade beschreibt das Upgrade-Verfahren, das von einem [Stack](#) zu einem anderen Stack definiert werden muss. Das Stack-Upgrade ist das Ergebnis des Upgrade-Verfahrens. Das heißt, das Stack-Upgrade ist ein Objekt, das mit dem Ziel-Stack verbunden ist. Ein Stack-Upgrade verfügt über dieselben Geschäftsfähigkeiten wie ein normaler Stack.

## Standardkomponente

Siehe [Komponente](#)

## Standardplattform

Eine Standardplattform ist eine [Plattform](#), die außerhalb des Rahmens einer konkreten [Applikation](#) oder [Komponente](#) definiert ist. Eine Standardplattform zielt darauf ab, Standards für die technische Architektur festzulegen. Eine gängige Praxis besteht darin, die konkrete Plattform einer Applikation mit Standardplattformen zu vergleichen, um die Überschneidungen zu bewerten und somit den Grad der Standard-Compliance zu ermitteln.

Standardplattformen können verwendet werden, um die technischen Plattformen (z. B. "Standard-Windows-Desktop") sowie die Business-Plattformen (z. B. "Standard-SAP-BW-Plattform") darzustellen. Wie alle Plattformen basiert auch die Standardplattform auf einer [Plattform-Layer-](#) und [Plattform-Tier-](#)Struktur, die in einer [Plattformvorlage](#) definiert ist.

## Standardplatfformelement

Ein Standardplatfformelement ist Bestandteil einer [Standardplattform](#). Ein Standardplatfformelement ist mit einer Standard-[Komponente](#) verbunden.

## Standardplattformkategorie

In einer Standardplattformkategorie werden inhaltsspezifische [Standards](#) zusammengefasst und klassifiziert. In Standardplattformkategorien können Sie die Standardplattformen in Ihrem Unternehmen hierarchisch strukturieren. Jede Standardplattform kann mit nur einer Standardplattformkategorie verbunden werden.

## Standort

Ein Standort ist ein geografischer Ort; z.B. ein Land, eine Stadt, ein Gebäude oder ein Raum. Ein [Gerät](#) kann einem Standort zugeordnet werden, der den physikalischen Standort des Geräts darstellt. Standorte können hierarchisch definiert werden.

## Status

Siehe [Objektstatus](#)

## Stellvertreter

Ein Stellvertreter ist ein Alfabet -Anwender, dem Schreib- und Lesezugriffsrechte für ein [Objekt](#) gewährt wurden, um im Namen des [autorisierten Anwenders](#) zu agieren.

## Steuerungselement

Ein Steuerungselement stellt eine strategische Absicht auf einer bestimmten Abstraktionsebene dar. Die Abstraktionsebene wird durch den [Steuerungselement-Stereotyp](#) festgelegt, zu dem das Steuerungselement gehört. Steuerungselemente, die einem Steuerungselement-Stereotyp auf der obersten Ebene zugeordnet sind, stellen eine hoch-abstrakte strategische Absicht dar (z.B. 'Ein integrierter Finanzdienstleistungsanbieter werden'), während die Steuerungselemente auf der untersten Ebene üblicherweise die Aktion beschreiben, die erforderlich ist, um die übergeordnete strategische Absicht umzusetzen. Daher stellt

der Steuerungselement-Stereotyp auf der untersten Ebene üblicherweise strategische Initiativen dar (z.B. 'Neue Lösung für Kredit- und Darlehensmanagement einführen').

Steuerungselemente, die benachbarten Steuerungselement-Stereotypen zugeordnet sind, können über eine Beziehung der Über- und Unterordnung verknüpft werden. Ein Steuerungselement kann über mehrere übergeordnete Steuerungselemente sowie über mehrere untergeordnete Steuerungselemente verfügen. Die Beziehung zwischen zwei Steuerungselementen wird durch [Steuerungselementkanten](#) dargestellt, die die Berechnung der Rangfolge und der relativen Bedeutung der strategischen Ziele ermöglichen. Steuerungselemente können auch mit [Architekturelementen](#) verknüpft werden und somit beschreiben, welche Architekturelemente von einem bestimmten Steuerungselement betroffen sein könnten.

Eine [Perspektive](#) kann einem Steuerungselement zugeordnet werden, um eine bestimmte Ansicht des Unternehmens auf das Steuerungselement auszudrücken. Ferner können für jedes Steuerungselement mehrere [Zielsetzungen](#) definiert werden, um eine Präzisierung der strategischen Absicht auszudrücken, die umgesetzt werden soll. In einem aktiven [Zeitreihen-Kennzahlensystem](#) kann die Bewertung von [Maßen](#) anhand von Messungen für das Unternehmen vorgenommen werden und angestrebte Zielmesswerte können für Zielsetzungen festgelegt werden.

Eine [Organisation](#) kann als Eigentümer des Steuerungselements definiert sein, um so die Verantwortlichkeit für die Umsetzung der strategischen Absicht sicherzustellen. Balanced-Scorecard-Analysen bieten Einblick in die Umsetzung der Zielsetzungen, die mit den Steuerungselementen verbunden sind, die entsprechend der Perspektive unternehmenskritisch sind.

## Steuerungselementkante

Eine Steuerungselementkante verbindet zwei [Steuerungselemente](#) benachbarter [Steuerungselement-Stereotypen](#), wodurch angegeben wird, dass ein untergeordnetes Steuerungselement zur Erreichung des strategischen Ziels des übergeordneten Steuerungselements beitragen soll. Die einem übergeordneten Steuerungselement untergeordneten Steuerungselemente werden klassifiziert, um deren relativen Beitrag zur Ausführung des übergeordneten Steuerungselements festzulegen.

Ein Steuerungselement kann mit mehreren übergeordneten Steuerungselementen verbunden sein und daher unterschiedliche Positionen in der jeweiligen Rangfolge einnehmen. Die Kantengewichtung eines jeden Steuerungselements spiegelt den Beitrag des Steuerungselements zum entsprechenden übergeordneten Steuerungselement wider. Daher kann ein Steuerungselement über viele verschiedene Steuerungselementkantengewichtungen verfügen.

## Steuerungselement-Stereotyp

Ein Strategienetzwerk besteht aus einer willkürlich gewählten Anzahl von Ebenen, die als Steuerungselement-Stereotypen bezeichnet werden. Der Steuerungselement-Stereotyp des Steuerungselements auf der obersten Ebene stellt die abstrakteste Ebene des strategischen Ziels dar. Der Steuerungselement-Stereotyp auf der untersten Ebene stellt die strategischen Initiativen dar, die befolgt werden müssen, um die abstrakten strategischen Ziele umsetzen zu können. Ein typisches Strategienetzwerk besteht zum Beispiel aus folgenden Steuerungselement-Stereotypen: Ebene 1: Vision, Ebene 2: Externe Trends, Ebene 3: Business-Treiber, Ebene 4: Geschäftsanforderungen, Ebene 5: Architekturanforderungen, Ebene 6: Maßnahmen. Der Name, die Anzahl und die hierarchische Ordnung der Steuerungselement-Stereotypen ist konfigurierbar.

Ein [Steuerungselement](#) wird daher für einen bestimmten Steuerungselement-Stereotyp definiert. Jedes Steuerungselementpaar, das im Strategienetzwerk zusammen gehört, muss zu unterschiedlichen Steuerungselement-Stereotypen gehören, die in der Hierarchie benachbart sind. Das heißt, dass

Steuerungselemente durch eine Beziehung der Über- und Unterordnung miteinander verbunden sind. Die Beziehung zwischen zwei Steuerungselementen wird durch [Steuerungselementkanten](#) dargestellt, die die Berechnung des relativen Rankings der strategischen Ziele ermöglichen.

## Strategische Bebauungsplan-Matrix

Eine strategische Bebauungsplan-Matrix ist Bestandteil einer [IT-Strategie](#), die eine Aufteilung der Strategieplanung in Bereiche der Verantwortlichkeit, der Ähnlichkeit oder der Gemeinsamkeit ermöglicht. Eine IT-Strategie kann über mehrere strategische Bebauungsplan-Matrizen verfügen. In der für jede strategische Bebauungsplan-Matrix definierte [Business-Support-Matrix](#) kann eine andere Zielarchitektur angegeben werden. Jede Bebauungsplan-Matrix kann eine Gruppe bevorzugter [ICT-Objekte](#) enthalten, die im Kontext der Business-Support-Matrix [strategische Business-Supports](#) bereitstellen sollen. Falls der Support nicht IT-bezogen ist, kann eine Gruppe bevorzugter [Organisationen](#) definiert werden. Strategische Bebauungsplan-Matrizen unterstützen die Ausrichtung strategischer Business-Supports entlang von [Business-Prozessen](#) oder [Domänen](#) im Kontext der Zielarchitektur.

## Strategischer Business-Support

Ein strategischer Business-Support ist ein angestrebter [Business-Support](#), der im Kontext der langfristigen [Zielarchitektur](#) für einen unbestimmten Zeitraum definiert ist. Ein strategischer Business-Support wird in der Regel durch ein vorhandenes oder geplantes [ICT-Objekt](#) bereitgestellt. Allerdings können strategische Business-Supports auch zur Erbringung durch ein [virtuelles ICT-Objekt](#), das ein noch nicht im Inventar vorhandenes Ad-hoc-ICT-Objekt definiert, durch einen [Lösungsbaustein](#) zur Beschreibung fachlicher Planung gemäß TOGAF oder durch eine [Organisation](#) zur Beschreibung nicht IT-bezogenen Supports, wie interne oder externe Services, konfiguriert werden.

## Swim-Lane

Eine Swim-Lane ist ein Umrandungselement in einem [Servicediagramm](#), das dazu dient, unterschiedliche Agenten, die Aktivitäten ausführen, voneinander zu trennen. Ein Agent kann eine Rolle sein, die von einer Person übernommen wird, oder eine [Applikation](#), die einen [Business-Service](#) bereitstellt, der von dem im Servicediagramm angezeigten [Business-Prozess](#) angefordert wird. Aktivitäten, [Ereignisse](#) des Typs "Zwischenschritt" und [Sequenzflüsse](#) können innerhalb der Swim-Lanes positioniert werden. Ein [Meldungsfluss](#) kann Aktivitäten aus unterschiedlichen Swim-Lanes miteinander verbinden.

## Symbolgalerie

Eine Symbolgalerie ist eine konfigurierbare Sammlung von Symbolen, die für die grafische Darstellung des quantitativen Werts von [Kennzahlen](#) in [Bewertungen](#) und Diagrammen verwendet werden können. Eine Symbolgalerie ist üblicherweise ein Satz miteinander verknüpfter Bilder, wobei jedes Symbol einen bestimmten Wertebereich darstellt. Beispielsweise könnten die drei Phasen einer Ampel 3 unterschiedliche Kennzahlen darstellen.



## T

## Taktischer Business-Support

Ein taktischer Business-Support ist ein vorgeschriebener [Business-Support](#), der im Kontext der mittelfristigen [Soll-Architektur](#) für einen bestimmten Zeitraum definiert ist. Ein taktischer Business-Support kann von einer vorhandenen oder geplanten [Applikation](#), einem [Lösungsbaustein](#), einem [ICT-Objekt](#), einem [virtuellen ICT-Objekt](#) oder, bei Bereitstellung des Supports über interne oder externe Services, von einer [Organisation](#) bereitgestellt werden. Der taktische Business-Support bietet Unterstützung für einen [Business-Prozess](#) (und [Business-Prozessversion](#)) oder eine [Domäne](#) und kann für eine Organisation oder ein [Marktprodukt](#) bereitgestellt werden.

## Technische Architektur

Siehe [Technische Architektur des Unternehmens](#)

## Technische Architektur des Unternehmens

Die technische Architektur des Unternehmens definiert die technische Ansicht der Unternehmensarchitektur und beschreibt die Software- und Hardware-[Komponenten](#), die Infrastruktur und die zugehörigen Standards des Unternehmens, wie etwa [Komponentenkataloge](#), [Standardplattformen](#) und [Masterplattformen](#).

## Technischer Lösungsservice

Ein technischer Lösungsservice ist ein vorgeschlagener [technischer Service](#), der für eine [Lösungskomponente](#) des Typs **Service** angegeben ist, die in der [Soll-Architektur](#) spezifiziert wurde, die für ein [Projekt](#) definiert ist. Der technische Lösungsservice ist als ein [Lösungsobjekt](#) außerhalb des [Inventorys](#) vorhanden, bis die für die Soll-Architektur vorgeschlagenen Architekturänderungen in das Inventory eingecheckt werden. Nach dem Einchecken wird der technische Lösungsservice zu einem realen [Inventory-Objekt](#) und verfügt dann über den [ObjektstatusGeplant](#). Der Objektstatus muss bei Betriebsbeginn manuell auf **Aktiv** geändert werden.

## Technischer Service

Ein technischer Service ist ein Service, der von einer [Komponente](#) bereitgestellt wird, um technische Anforderungen zu erfüllen, die zur Unterstützung von [Business-Service-Anfragen](#) erforderlich sind. Ein technischer Service kann [Operationen des technischen Services](#) definieren, durch die die technische Ausführung von [Business-Funktionen](#) unterstützt wird.

Technische Services können für Komponenten und lokale Komponenten definiert werden, für die das Attribut **Komponententyp** auf `Service` gesetzt wurde. Wenn eine [lokale Komponente](#) des Typs `Service` in die Architektur einer [Applikation](#) oder einer anderen Komponente (z. B. des Typs `Infrastructure` oder `Business`) eingebettet ist, können Operationen des technischen Services der eingebetteten Komponente den

[Business-Services](#) zugeordnet werden, die von der Applikation oder der lokalen Komponente bereitgestellt werden.

Für die Klasse **Technischer Service** können [Objektclassenstereotype](#) konfiguriert werden.

## Technisches Umgebungselement

Ein technisches Umgebungselement beschreibt einen Aspekt der [technischen Umgebung](#), der dokumentiert werden soll. Das technische Umgebungselement beinhaltet normalerweise eine [Alfabet-Abfrage](#), über die der Satz an [Architekturelementen](#) gefunden wird, die für diesen Aspekt der technischen Umgebung spezifiziert werden können. Anwender mit den zulässigen [Anwenderprofilen](#) können aus dem Ergebnisdatensatz der Abfrage das Architekturelement auswählen, das ein Objekt in der technischen Umgebung unterstützt.

## Technische Umgebung

Eine technische Umgebung beschreibt einen bestimmten Bereich der Technologie, die zur Unterstützung der Entwicklung, der Pflege, des Betriebs oder des Testens eines Objekts erforderlich ist. Technische Umgebungen werden üblicherweise für Testzwecke und aus betriebsverwaltungstechnischen Gründen dokumentiert und können zum Beispiel Entwicklungs- oder Test-Tools beinhalten. Technische Umgebungen könnten auch zur Dokumentation der selbständig entwickelten [Applikationen](#) oder [Komponenten](#) implementiert werden. In diesem Fall sind verschiedene Design-, Entwicklungs- und Test-Tools (Komponenten) erforderlich, die im Kontext der Applikation/Komponente bewertet werden müssen, für die sie genutzt werden.

Eine Definition der technischen Umgebung kann für eine oder mehrere der folgenden [Objektclassen](#) konfiguriert werden: Applikation, Komponente, Installation, Gerät, Standardplattform und Anbieterprodukt. Wenn die Objektclassen [Objektclassenstereotypen](#) unterstützt, können alle Stereotypen so konfiguriert werden, dass sie die Definition technischer Umgebungen unterstützen. Durch die Definition der technischen Umgebung werden auch die [Anwenderprofile](#) bestimmt, die über Schreib-/Lesezugriffsrechte verfügen, um die technischen [Architekturelemente](#) für das ausgewählte Objekt zu bestimmen.

Jede Definition der technischen Umgebung kann über mehrere Definitionen von technischen Umgebungselementen verfügen und in der Definitionshierarchie für technischen Umgebungselemente über eine unbegrenzte Anzahl von Ebenen verfügen. Üblicherweise wird für ein technisches Umgebungselement eine [Alfabet-Abfrage](#) definiert, um die Architekturelemente zu finden, die für einen Aspekt der technischen Umgebung definiert werden können.

Zum Beispiel könnte die technische Umgebung "Testverfahren" die untergeordneten technischen Umgebungselemente "Funktionstests", "Leistungs- und Auslastungstests" und so weiter beinhalten. Jedes technische Umgebungselement würde eine Alfabet-Abfrage enthalten, über die nach den entsprechenden Test-Tools gesucht werden kann, die ein Anwender auswählen kann, um die Testumgebung des Objekts zu definieren. Beispielsweise kann eine Alfabet-Abfrage, die der Definition des technischen Umgebungselements "Funktionstest" zugeordnet ist, so konfiguriert werden, dass darüber die [Applikationen](#) gefunden werden, die für den Funktionstest von [Geräten](#) erforderlich sind.

## Technologie

Unter Technologie wird alles zusammengefasst, was technische Prozesse, Methoden oder Fachwissen nutzt. Technologien können in [Technologiegruppen](#) strukturiert werden. Ferner können Technologien zu anderen Technologien zugeordnet werden. Die Technologien "ADO.NET" und "C#" könnten der ausgewählten Technologie ".NET" zugeordnet werden, da es sich in beiden Fällen um .NET-Technologien handelt.

Eine Technologie kann mehreren [Komponenten](#), [Anbieterprodukten](#) und [Standardplattformen](#) zugeordnet werden, wodurch angegeben wird, dass diese Objekte Instanzierungen einer bestimmten Technologie sind. Eine Komponente, ein Anbieterprodukt oder eine Standardplattform kann mehreren Technologien zugeordnet werden. Lebenszyklusanalysen bieten einen Einblick in den Lebenszyklus der Technologie sowie der Komponenten, Anbieterprodukte und Standardplattformen, in denen die Technologie genutzt wird.

Technologien können auch einer [IT-Fähigkeit](#) zugeordnet sein, wodurch angegeben wird, dass die Technologie die IT-Fähigkeit potentiell über eine oder mehrere Komponenten unterstützen kann.

## Technologiegruppe

Eine Technologiegruppe ist ein Container zur logischen Strukturierung von [Technologien](#). Es gibt verschiedene Möglichkeiten, Technologien logisch zu strukturieren. Daher kann jede Technologie mit mehreren Technologiegruppen verbunden werden. Für die Objektklasse **Technologiegruppe** können [Objektklassenstereotype](#) konfiguriert werden.

## Textvorlage

- [Aufgaben](#)
- [Workflows](#)
- [Diskussionsgruppen](#)
- [Konsistenzmonitore](#), [Benachrichtigungsmonitore](#), [Aktivitätsmonitore](#), [Inaktivitätsmonitore](#) und [Zeitmonitore](#)
- Notizblockverwendung in [Business-Support-Matrizen](#)
- Strukturänderungen am Organisationsmodell
- Strukturänderungen am Business-Prozessmodell

## Tier

Siehe [Plattform-Tier](#)

---

## U

### Unternehmensarchitektur

Die Unternehmensarchitektur ist eine umfassende Methode und ein Framework, das verwendet wird, um die [Business-Prozesse](#), die [Organisationen](#), die Informationstechnologie (IT), die Software und die Hardware, lokale und Wide-Area-Networks (LANs bzw. WANs), Mitarbeiter, Operationen und Projekte eines Unternehmens zu verwalten und an der Gesamtstrategie des Unternehmens auszurichten.

### Unternehmenskalender

Ein Unternehmenskalender ist ein für das Unternehmen erzeugter Kalender, in dem bestimmte Tage als Ferien oder Wochenende gesperrt sind. Unterschiedliche Unternehmenskalender können für alle relevanten Regionen des Unternehmens erzeugt werden. Anwender können einen Unternehmenskalender als Vorlage importieren, wenn sie ihre eigenen persönlichen Kalender erzeugen.

## V

### Verantwortlicher Anwender

Siehe [Autorisierter Anwender](#) oder [Rolle](#).

### Verantwortlichkeit

Eine Verantwortlichkeit definiert die fachliche Beziehung, die ein bestimmter Anwender oder eine [Anwendergruppe](#) zu einem [Objekt](#) hat. Diese Verantwortlichkeit kann durch Definition einer [Rolle](#) für das ausgewählte Objekt spezifiziert werden.

Im Gegensatz zu einem [autorisierten Anwender](#) ist ein Anwender oder eine Organisation mit einer Rolle nicht zur Pflege der Daten eines Objekts verpflichtet. Rollen werden nur zu Informationszwecken definiert und bieten Detailinformationen über Anwender oder Organisationen, die wiederum über Informationen über das Objekt verfügen oder daran interessiert sind. Die Definition einer Rolle für ein Objekt hat keinerlei Auswirkung auf Zugriffsrechte.

### Verbindung

Eine Verbindung ist ein Verbindungselement in einem [Service-Diagramm](#), in dem die Beziehung zwischen einem Servicediagrammelement und einem [Architekturelement](#) dargestellt wird, das von diesem Element referenziert wird. Mittels einer Verbindung erfolgt die logische Verbindung von einem Element im Service-Diagramm zum Inventory-Objekt. Üblicherweise verbindet eine Verbindung einen [Business-Service](#), der

---

von einem [Business-Prozess](#) oder einer [Business-Funktion](#), die der angeforderte Business-Service referenziert, angefordert wird.

## Verbindungsdatenformat

Das Verbindungsdatenformat beschreibt das Datenformat, das für die Übertragung von [Businessdaten](#) über einen bestimmten [Informationsfluss](#) genutzt wird. Hierzu zählen z.B.: ASCII, XML.

## Verbindungsfrequenz

Eine Verbindungsfrequenz beschreibt, wie oft ein bestimmter [Informationsfluss](#) genutzt wird, um [Businessdaten](#) zwischen den zwei miteinander verbundenen [Applikationen](#) oder deren jeweiligen [Komponenten](#) zu übertragen. Hierzu zählen z.B.: täglich, monatlich.

## Verbindungsmethode

Eine Verbindungsmethode beschreibt die von einem bestimmten [Informationsfluss](#) für die Übertragung von [Business-Daten](#) zwischen den beiden zugehörigen [Applikationen](#) oder deren entsprechenden [Komponenten](#) genutzte Übertragungsmethode. Hierzu zählen z.B.: TCP/IP, Dateitransfer, Meldungswarteschlange.

## Verbindungstyp

Ein Verbindungstyp beschreibt den von einem bestimmten [Informationsfluss](#) für die Übertragung von [Businessdaten](#) zwischen den beiden zugehörigen [Applikationen](#) oder deren entsprechenden [Komponenten](#) genutzten Übertragungsmodus. Hierzu zählen z.B.: Batch und Online.

## Vertrag

Ein Vertrag ist ein rechtliches Dokument, in dem die Vereinbarungsbedingungen festgelegt werden zwischen Organisationen, die Produkte und Services erwerben und Organisationen oder [Anbietern](#), die die Produkte und Services bereitstellen.

Ein Vertrag kann als ein Rahmenvertrag von mehreren Verträgen referenziert werden.

Verträge werden in [Vertragsordnern](#) strukturiert. Ein Vertrag kann nur einer Vertragsgruppe zugeordnet werden.

Das Unternehmen kann für die Klasse Vertrag [Objektklassenstereotypen](#) konfigurieren. Ein Stereotyp für die Klasse "Vertrag" könnte z.B. "Software-Lizenzen" sein und ein anderer Objektklassenstereotyp könnten "Serviceverträge" sein.

## Vertragsbestandteil

Ein Vertragsbestandteil ist Teil eines [Vertrags](#), für dessen Pflege oder Ausführung üblicherweise eine bestimmte [Organisation](#) verantwortlich ist. Ein Vertragsbestandteil könnte z.B. der Lizenzerwerb eines Software-Produkts oder die Wartung des Software-Produkts sein.

Für einen Vertrag können mehrere Vertragsbestandteile definiert werden. Für einen Vertragsbestandteil können sowohl die verantwortliche Organisation als auch die Kosteninformationen erfasst werden. Ein Vertragsbestandteil kann auch über [Vertragsleistungen](#) verfügen, die die Architekturelemente und Ressourcen spezifizieren, die bereitgestellt werden müssen, um den Vertrag zu erfüllen.

Das Unternehmen kann für die Klasse Vertragsbestandteil [Objektklassenstereotypen](#) konfigurieren. Ein Stereotyp für die Klasse "Vertragsbestandteil" könnte z.B. "Lizenzerwerb" sein und ein anderer Objektklassenstereotyp könnten "Wartungsvereinbarung" sein.

## Vertragsgruppe

Eine Vertragsgruppe ist ein Container zur Strukturierung von [Verträgen](#). Es gibt verschiedene Möglichkeiten, Verträge logisch zu strukturieren. So kann jeder beliebige Vertrag mit mehreren Vertragsgruppen verbunden werden, wodurch derselbe Vertrag in verschiedenen semantischen Zusammenhängen betrachtet werden kann.

## Vertragsleistung

Eine Vertragsleistung ist ein bestimmtes Architekturelement oder ein Satz von Ressourcen, die bereitgestellt werden müssen, um die Bedingungen des zugehörigen [Vertrags](#) oder [Vertragsbestandteils](#) zu erfüllen. Für die Klasse **Vertragsleistung** können [Objektklassenstereotype](#) konfiguriert werden.

## Vertreter

Ein Vertreter ist eine Person, die als alternativer Anwender für die [Aufgaben](#) und [Workflowschritte](#) verantwortlich sind, wenn der eigentlich verantwortliche Anwender geplant abwesend ist. Der Vertreter kann entweder einzelnen Aufgaben oder Workflowschritten oder allen Aufgaben oder Workflowschritten eines Anwenders zugeordnet sein. Der Vertreter verfügt über die erforderlichen Zugriffsberechtigungen für die Aufgaben oder Workflowschritte, für die er über die Vertreterzuweisung verantwortlich ist.

## Virtuelle Organisation

Eine virtuelle Organisation ist entweder eine temporäre oder permanente [Organisation](#), wie z.B. ein Entscheidungsgremium, ein Steuerungskomitee oder andere Institutionen, die außerhalb der formalen Organisationsstruktur eines Unternehmens existieren. Eine virtuelle Organisation könnte ein IT-Komitee sein, das eine strategische Richtung für das Unternehmen bezüglich der Ausrichtung des Geschäfts und der IT vorgibt.

## Virtuelles ICT-Objekt

Ein virtuelles ICT-Objekt, stellt ein visionäres [ICT-Objekt](#) dar, das eventuell noch nicht im Inventory definiert ist, aber erforderlich ist, um einen [strategischen Business-Support](#) oder einen [taktischen Business-Support](#) darzustellen, der in einer [strategischen Bebauungsplan-Matrix](#) oder einer [Bebauungsplan-Matrix](#) definiert ist. Ein virtuelles ICT-Objekt dient nur zu Planungszwecken und wird außerhalb des Bereichs der Strategieplanung oder [Bebauungsplanung](#) nicht angezeigt.

Nachdem ein virtuelles ICT-Objekt in einer [Business-Support-Matrix](#) definiert wurde, kann es später durch ein reales ICT-Objekt oder eine [Applikation](#) ersetzt werden. In diesem Fall werden durch das reale ICT-Objekt alle Business-Supports ersetzt, die mit dem virtuellen ICT-Objekt verbunden sind.

## Vorgeschlagene lokale Komponente

Eine vorgeschlagene lokale Komponente ist in der Regel eine ermittelte Komponente, die über eine Ermittlung importiert wurde, und dient als potenzielle Technologie für die Aufnahme in eine [Applikations-](#) oder [Komponenten-](#)Plattform. Vorgeschlagene lokale Komponenten werden in der Regel verwendet, um ein Upgrade auf einfache technologische Änderungen durchzuführen, ohne den Planungs- und Investitionsbedarf, der bei bedeutenden Architekturänderungen anfallen würde, typischerweise mit Festlegung in einer entsprechenden [Projektlösung](#). Vorgeschlagene lokale Komponenten können auch verwendet werden, um vorhandene [lokale Komponenten](#) zugunsten einer vereinfachten [Versionierung](#) der Technologie zu ersetzen. Eine vorgeschlagene lokale Komponente lässt sich explizit in eine „echte“ lokale Komponente ändern. In diesem Fall erbt die lokale Komponente alle für die vorgeschlagene lokale Komponente definierten Attribute und die vorgeschlagene lokale Komponente wird aus der Alfabet-Datenbank gelöscht.

## Vorgeschlagener Informationsfluss

Ein vorgeschlagener Informationsfluss wird in der Regel über eine Ermittlung importiert und dient als potenzieller [Informationsfluss](#) für eine Quell- oder Ziel-[Applikations-](#) oder [Komponenten-](#)Plattform. Ein vorgeschlagener Informationsfluss lässt sich explizit in einen „echten“ Informationsfluss ändern. In diesem Fall erbt der Informationsfluss alle Attribute, die für den vorgeschlagenen Informationsfluss festgelegt wurden, und der vorgeschlagene Informationsfluss wird aus der Alfabet-Datenbank gelöscht.

## Vorschau

Eine Vorschau ist eine gekürzte Version des Objektprofils, die unterhalb einiger tabellarischer Berichte angezeigt wird, und in der die wichtigsten Aspekte eines Objekts zusammengefasst sind.

## W

### Wertschöpfungskette

Eine Wertschöpfungskette beschreibt eine Reihe von [Wertschöpfungsketten-Schritten](#), die zum Implementieren von Lösungen relevant sind, die dem Unternehmen einen kontinuierlichen Wertschöpfungsfluss ermöglichen. Mithilfe von Wertschöpfungsketten kann „Wert“ im Sinne von Nützlichkeit, Vorteilen oder Nutzen von IT-Lösungen in der [Business-Architektur](#) artikuliert und gemessen werden. Nicht monetäre Beispiele für „Wert“ sind beispielsweise die erfolgreiche Bereitstellung eines Produkts oder Services oder der Zugriff auf aktuelle Informationen, um bessere Geschäftsentscheidungen treffen zu können. Es können Zielwerte für die Wertschöpfungskette definiert werden, um strategische Absichten ([Steuerungselemente](#)) im Strategienetzwerk des Unternehmens als Ziel zu definieren, in dessen Höhe die Wertschöpfungskette Wert liefern soll.

### Wertschöpfungsketten-Schritt

Ein Wertschöpfungsketten-Schritt beschreibt die einzelnen Phasen, die erforderlich sind, damit die [Wertschöpfungskette](#) den relevanten Beteiligten Wert liefert. Der Wertschöpfungsketten-Schritt kann von einem [Business-Prozess](#) abgeleitet sein. Die Definition des Wertschöpfungsketten-Schritts beinhaltet die [Organisationen](#) der Beteiligten sowie die [Geschäftsfähigkeiten](#) oder [-Domänen](#), die vom Wertschöpfungsketten-Schritt genutzt werden. Für jeden einzelnen Wertschöpfungsketten-Schritt können mithilfe von [Wertschöpfungsketten-Schritt-Bedingungen](#) die Bedingungen definiert werden, um ihn zu beginnen und zu beenden. Es können [Kennzahlen](#) festgelegt werden, um den von dem Wertschöpfungsketten-Schritt gelieferten Wert zu erfassen.

### Wertschöpfungsketten-Schritt-Bedingung

Mithilfe einer Wertschöpfungsketten-Schritt-Bedingung können die Bedingungen angegeben werden, um einen [Wertschöpfungsketten-Schritt](#) zu beginnen oder zu beenden. Diese Bedingungen stellen eine semantische Beschreibung der Voraussetzungen dar, die erfüllt sein müssen, um den Wertschöpfungsketten-Schritt zu beginnen oder zu beenden. Für jede Bedingung können Bewertungen festgelegt werden, um die [Kennzahlen](#) zu definieren, mit denen gemessen wird, ob die Bedingung erfüllt ist.

### Wizard

Ein Wizard ist ein Assistent, der aus einem konfigurierten Satz an Standardeditoren, benutzerdefinierten Editoren, Standardansichtsseiten und konfigurierten Berichten besteht, die in [Wizard-Schritten](#) definiert sind. Üblicherweise führt der Wizard den Anwender durch einen linearen mehrstufigen Prozess der Datenerfassung für ein Objekt, auf das der Anwender Zugriffsberechtigungen hat.

tandard-Wizards sind zur Datenerfassung für Gebrauchsklassen verfügbar, in denen üblicherweise eine Vielzahl von Objekten dokumentiert wird. Für eine Objektklasse kann eine unbegrenzte Anzahl von benutzerdefinierten Wizards konfiguriert werden, damit die unterschiedlichen Anwender, die in unterschiedlichen



Kontexten arbeiten, die Objektdaten erfassen können, für die sie verantwortlich sind. Pro [Anwenderprofil](#) kann pro Objektklasse jedoch nur ein Wizard verfügbar sein.

## Wizard-Schritt

- Einen Standard-Editor
- Einen benutzerdefinierten Editor mit einer Registerkarte
- Einen benutzerdefinierten Editor mit mehreren Registerkarten
- Ein benutzerdefinierter Editor mit Sichtbarkeitsregeln, die die Sichtbarkeit benutzerdefinierter Editorseiten auf Basis der für eine benutzerdefinierte Objektklasseneigenschaft ausgewählten Werte definieren.
- Eine Standard-Ansichtsseite
- Einen tabellarischen, abfragebasierten benutzerdefinierten Bericht
- Einen abfragebasierten benutzerdefinierten Bericht mit Grafiken (Treemaps oder geschichtete Diagramme)

## Workflow

Ein Workflow ist ein gemeinschaftlicher Prozess von [Workflow-Schritten](#), die üblicherweise von einem oder mehreren Anwendern ausgeführt werden. Ein Workflow basiert auf einer konfigurierten [Workflow-Vorlage](#), durch die eine Abfolge von Workflow-Schritten festgelegt wird, die für ein bestimmtes Objekt und dessen Referenzen von bestimmten Anwendern ausgeführt werden müssen. Für Workflowschritte können bestimmte Vor- und Nachbedingungen definiert sein, durch die unterschiedliche Bearbeitungswege innerhalb des Workflows in Abhängigkeit der Erfüllung der jeweiligen Bedingungen bestimmt werden.

Üblicherweise ist der Workflow-Eigentümer der Anwender, der den Workflow initiiert und der für die Verwaltung des Workflows verantwortlich ist. Wenn ein Workflow von einem Workflow-Eigentümer initiiert wird oder wenn ein Workflow in den nächsten Workflowschritt übergeht können die entsprechenden Anwender mittels automatisch generierter E-Mails über ihre bevorstehenden Verantwortlichkeiten informiert werden. Durch diese Funktionalität wird gewährleistet, dass alle relevanten Anwender über ihre Verantwortlichkeiten in dem gemeinschaftlichen Workflow informiert und an diese erinnert werden. Die Optionen, Workflowschritte abzulehnen, zu delegieren und auszusetzen, Anwender an bevorstehende Zieldaten von Workflowschritten zu erinnern sowie einen Workflow, in dem ein Fehler aufgetreten ist, umzuleiten, ermöglichen es den Workflow-Eigentümern und den Workflow-Administratoren, die Fertigstellung der einzelnen Workflowschritte und somit des gesamten Workflows zu verfolgen, zu koordinieren und zu verwalten.

## Workflowschritt

Ein Workflowschritt ist eine in einem [Workflow](#) auszuführende Aktivität oder Aufgabe. Eine [Workflow-Vorlage](#) kann so viele Schritte beinhalten, wie notwendig sind, um die entsprechende Aufgabe zu erledigen. In Workflowschritten können Standard- und benutzerdefinierte Editoren, [Wizards](#), Ansichtsseiten, [konfigurierte Berichte](#) und [Objektprofile](#) implementiert werden, so dass Benutzer die Aufgabe fertig stellen können. Ein Workflowschritt kann den Zugriff auf Daten, deren Änderung oder Überprüfung beinhalten, oder so konfiguriert sein, dass er automatisch durch das Alfabet -System ausgeführt wird. Wenn der Workflowschritt

nicht automatisch durch das System ausgeführt wird, können ein oder mehrere Anwender definiert werden, die für die Durchführung verantwortlich sind. Ein Anwender kann einen Workflowschritt an einen anderen Anwender delegieren oder einen Workflowschritt ablehnen. Abhängig von der Konfiguration des Workflowschritts kann es erforderlich sein, dass ein Anwender einen Workflowschritt bestätigen muss, bevor der Workflow in die nächsten Workflowschritte übergehen kann. Die Bestätigung des Workflowschritts kann aber auch automatisch erfolgen. Einem Workflowschritt können auch eine oder mehrere Workflowschritt-Aktionen zugeordnet sein, die die Ausführung diverser Operationen beim Aufrufen, Verlassen, Ablehnen oder Ablaufen des Workflowschrittes veranlassen. Zusätzlich können für einen Workflowschritt Vorbedingungen oder Nachbedingungen definiert sein, die erfüllt sein müssen, damit auf den Workflowschritt zugegriffen werden kann oder damit er verlassen werden kann.

## Workflow-Vorlage

Eine Workflow-Vorlage ist ein benutzerdefinierter Entwurf für einen oder mehrere [Workflows](#). Mithilfe der Vorlage wird bestimmt, welche Objektklasse der Ausgangspunkt des Workflows ist, welche [Anwendergruppen](#) und/oder [Anwenderprofile](#) den Workflow initiieren und verwalten dürfen, welche [Workflow-Schritte](#) der Workflow in welcher Abfolge umfasst, mögliche Vor- und Nachbedingungen oder mit einem Workflowschritt einhergehende Aktualisierungen, und welche Art von Workflow-Benachrichtigungen an zusammenarbeitende Anwender in welchen Kontexten verschickt werden sollen. Der Anwender, der die Workflow-Vorlage erzeugt, ist der Eigentümer der Workflow-Vorlage.

Für eine Workflow-Vorlage muss das Attribut **Workflow-Status** auf "Plan" gesetzt werden, damit diese konfiguriert und überprüft werden kann. Sobald die Workflow-Vorlage fertiggestellt und genehmigt wurde, muss das Attribut **Workflow-Status** in "Active" geändert werden, um die Workflow-Vorlage für die Anwendergemeinschaft zur Verfügung zu stellen. Sobald die Workflow-Vorlage in Alfabet verfügbar ist, kann ein berechtigter Anwender auf Basis der Workflow-Vorlage einen Workflow initiieren. Es können mehrere Workflows gleichzeitig für eine Workflow-Vorlage initiiert und durchgeführt werden.

X

Y

Z

## Zeitmonitor

Ein Zeitmonitor ist ein [Monitor](#), der die in einem Verteiler eingetragenen Anwender über bevorstehende Termine, die für [Objekte](#) einer bestimmten [Objektklasse](#) definiert wurden, informiert (z.B. über ein bevorstehendes Start- oder Enddatum). In Alfabet gibt es zwei Arten von Zeitmonitoren: objektspezifische Zeitmonitore und systemweite Zeitmonitore, deren Ziel alle Objekte in einer bestimmten Objektklasse sind.

Objektspezifische Zeitmonitore werden von den einzelnen Anwendern definiert, um das Herannahen eines Termins für spezielle Objekte zu überwachen. Der Eigentümer des Monitors definiert eine Reihe von

[Eigenschaften](#), die überwacht werden sollen, sowie die Anwender, die benachrichtigt werden sollen, sobald der Monitor ausgelöst wird. Diese Anwender sowie auch der Monitoreigentümer werden informiert, sobald eine definierte Eigenschaft eines Objekts in der Klasse ein Zieldatum erreicht. Dann kann entschieden werden, welche Maßnahmen durchgeführt werden sollen. Die Überwachung erfolgt über einen bestimmten Zeitraum in regelmäßigen Intervallen.

Zusätzlich zu den herkömmlichen Alfabet -Zeitmonitoren, können systemweit definierte Zeitmonitore für Objektklassen konfiguriert werden. Wenn ein bestimmtes Datum bevorsteht, erhalten alle [autorisierten Anwender](#), die für die entsprechenden Objekte in der Objektklasse verantwortlich sind, automatisch eine Benachrichtigung per E-Mail, in der sie beauftragt werden, das Objekt zu überprüfen.

## Zeitreihen-Kennzahlensystem

Ein Zeitreihen-Kennzahlensystem ist ein [Kennzahlensystem](#), für das Zielwerte für aktive [Zeitreihen-Zeiträume](#) für Objekte in der Objektklasse definiert werden können, der das Kennzahlensystem zugeordnet ist. Die Anwender können für die dem Kennzahlensystem zugeordneten [Kennzahltypen](#) Zielwerte definieren. Einer Objektklasse können mehrere Zeitreihen-Kennzahlensysteme zugeordnet werden.

## Zeitreihen-Zeitraum

Ein Zeitreihen-Zeitraum ist ein Zeitraum (z.B. 2010-01), für den ein konfiguriertes [Zeitreihen-Kennzahlensystem](#) bewertet werden kann. Ein Zeitreihen-Zeitraum kann aktiviert und deaktiviert werden. Jeder aktive Zeitreihen-Zeitraum ist in Alfabet -Ansichten sichtbar und die Anwender können somit die [Kennzahltypen](#) bewerten, die dem Zeitreihen-Kennzahlensystem für die Objekte in der relevanten Objektklasse zugeordnet sind. Sobald ein Zeitreihen-Zeitraum deaktiviert wurde, kann dieser nicht mehr bearbeitet werden und ist auch nicht mehr in den Ansichten sichtbar.

## Zielarchitektur

Die Zielarchitektur stellt den gewünschten, langfristigen Status der [IT-Landschaft](#) ohne festgelegten Zeitrahmen oder konkrete Verweise auf Applikationen dar.

Eine Zielarchitektur wird durch eine [IT-Strategie](#) beschrieben. Es besteht die Möglichkeit, verschiedene Zielarchitekturen zu definieren, die sich hinsichtlich der langfristigen Business-Szenarien für das Unternehmen unterscheiden. Der Aktionsplan für die Erreichung der aktuell anwendbaren Zielarchitektur wird im [Bebauungsplan](#) dargestellt.

Die Zielarchitektur gleicht normalerweise eher einer strategischen Richtlinie als einem detaillierten Architekturplan. Daher wird die Zielarchitektur weniger detailliert beschrieben als die [Ist-Architektur](#) oder die [Soll-Architektur](#). Die Beschreibung der Zielarchitektur ist üblicherweise auf den [strategischen Business-Support](#) beschränkt und wird als statischer Zustand ohne Gültigkeitszeitraum oder [Lebenszyklus](#)-Beschreibung für dessen Bestandteile formuliert.

## Zielsetzung

Eine Zielsetzung stellt eine Verfeinerung und Spezifikation eines [Steuerungselements](#) dar, die umgesetzt werden kann. Zielsetzungen können nur einem Steuerungselement zugeordnet werden und mehrere

Steuerungselemente können nicht dieselbe Zielsetzung haben. Für ein Steuerungselement können mehrere Zielsetzungen definiert sein. Eine Zielsetzung erbt die Anwenderautorisierungsdefinition von dem Steuerungselement, dem sie zugeordnet ist.

Jede Zielsetzung sollte mindestens einem [Maß](#) zugeordnet sein, um die Ziel-[Messungen](#) für die Zielsetzung zu definieren. Dies ermöglicht die Überprüfung der Umsetzung oder Erreichung strategischer Absichten, die im Hinblick auf tatsächliche Messungen und Ziele gemessen und verfolgt werden sollen.

## Zugehörige Objekte

Ein zugehöriges Objekt ist ein [Architekturelement](#), das mit einer [Domäne](#) verbunden und dieser zugeordnet ist. Im Gegensatz zu einem [primären Objekt](#), das mit einer Domäne verbunden ist, ist ein zugehöriges Objekt nicht Eigentum der Domäne. Eine [Applikation](#) kann beispielsweise in verschiedenen funktionalen Kontexten verwendet werden und ist daher mit verschiedenen Domänen verbunden, die nicht die primäre Domäne der Applikation sind. Neben Applikationen können [Komponenten](#), [ICT-Objekte](#), [Business-Funktionen](#), [Business-Objekte](#), [Business-Prozesse](#), [Standardplattformen](#), [Marktprodukte](#) und [Anbieterprodukte](#) als zugehörige Objekte definiert werden.

Ob für eine Domäne zugehörige Objekte definiert werden können, ist von der Lösungskonfiguration abhängig. Wenn die Definition von zugehörigen Objekten zulässig ist, kann eine Domäne über eine unbegrenzte Anzahl von zugehörigen Objekten verfügen und ein Objekt kann mehreren zugehörigen Domänen zugeordnet werden. Es wird jedoch empfohlen, die Anzahl zugehöriger Domänen pro Objekt möglichst gering zu halten.