

IT-Transformation

Technische und philosophische
Herausforderungen

Ulrich Viehhaus

20. November 2024



2024-11-20

IT-Transformation

IT-Transformation
Technische und philosophische
Herausforderungen
Ulrich Viehhaus
20. November 2024



Inhalt

Einleitung

IT-Transformation

Frameworks

Architektur

Schnittstellen

Sicherheit

Datenschutz

Dos and Don'ts

Abschluss

2024-11-20

IT-Transformation

└─ Einleitung

└─ Inhalt

Inhalt

Einleitung
IT-Transformation
Frameworks
Architektur
Schnittstellen

Sicherheit
Datenschutz
Dos and Don'ts
Abschluss



Einleitung

*Wer bin ich, warum bin ich hier
und worum geht es?*



2024-11-20

IT-Transformation
└─ Einleitung
└─ Einleitung

Einleitung

*Wer bin ich, warum bin ich hier
und worum geht es?*



Am Tag

Ulrich Viefhaus

- DevOps Engineer und Information Security Officer bei der BRANDAD Solutions GmbH
- Certified Professional for Software Architecture Foundation Level
- Certified Application Security Engineer (Java)
- Certified Cloud Security Engineer
- Certified Ethical Hacker

2024-11-20

IT-Transformation

└─ Einleitung

└─ Am Tag

Am Tag

Ulrich Viefhaus

- DevOps Engineer und Information Security Officer bei der BRANDAD Solutions GmbH
- Certified Professional for Software Architecture Foundation Level
- Certified Application Security Engineer (Java)
- Certified Cloud Security Engineer
- Certified Ethical Hacker



In der Nacht

seism0saurus

- Blogger: seism0saurus.de
- Mastodon: [@seism0saurus@infosec.exchange](https://seism0saurus@infosec.exchange)
- Coorganisator der Open Security Conference
- Teilnehmer an Unkonferenzen
- OpenSource Entwickler

2024-11-20

IT-Transformation

└─ Einleitung

└─ In der Nacht

In der Nacht

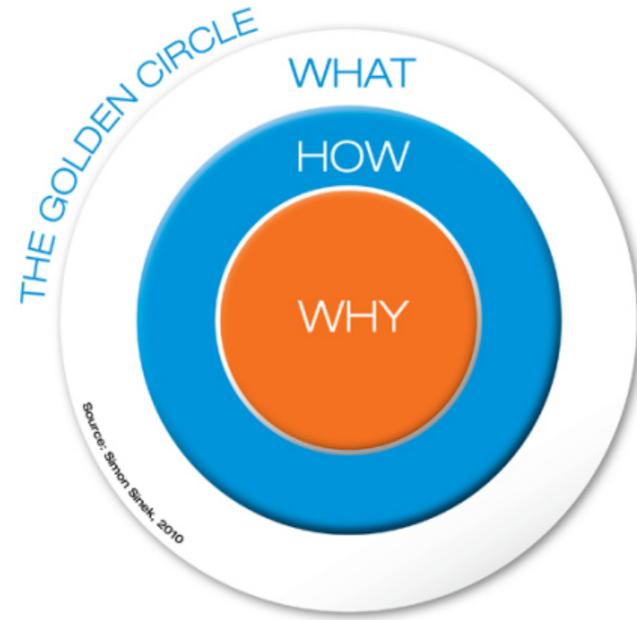
seism0saurus

- Blogger: seism0saurus.de
- Mastodon: [@seism0saurus@infosec.exchange](https://seism0saurus@infosec.exchange)
- Coorganisator der Open Security Conference
- Teilnehmer an Unkonferenzen
- OpenSource Entwickler



Start with why

From Simon Sinek



[1]

2024-11-20

IT-Transformation

└─ Einleitung

└─ Start with why

Start with why

From Simon Sinek



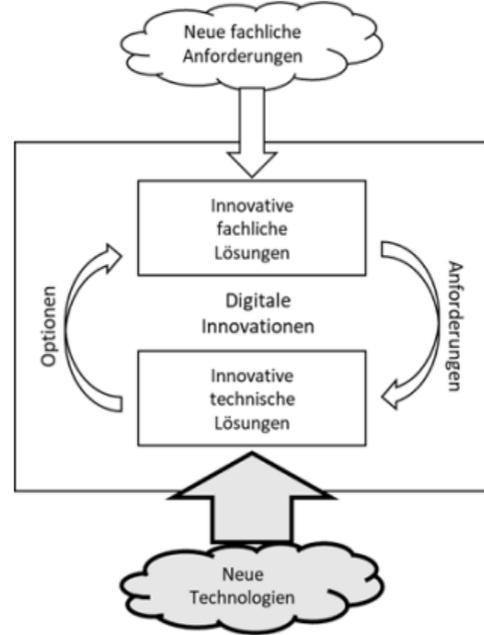
[1]

- Goldener Kreis von Simon Sinek, einem britisch-US-amerikanischen Autor und Unternehmensberater, unter Creative Commons 2.0[1]
- Sinek: Menschen kaufen nicht, was man macht; sie kaufen, warum man etwas macht



Innovationen

Wechselwirkung von neuen Anforderungen und neuen Möglichkeiten



[2]

2024-11-20

IT-Transformation

└─ Einleitung

└─ Innovationen



- Bild 1 aus Digital innovations im Journal Electronic Markets von Florian Wiesböck und Thomas Hess
- digitale Innovationen [umfassen] zwei Komponenten, eine technische und eine fachliche Lösung[3]
- Innovative fachliche Lösungen benötigen neue technische Lösungen (Technology-Pull). Z.B. Infektionsverfolgung bei Corona
- Innovative technische Lösungen ermöglichen neue fachliche Lösungen (Technology-Push). Z.B. KI
- Einen schlechten Prozess zu digitalisieren ergibt einen schlechten digitalen Prozess



IT-Transformation

*Was bedeutet Transformation
im Kontext der Informations-
technologie?*



2024-11-20

IT-Transformation
└ IT-Transformation
└ IT-Transformation

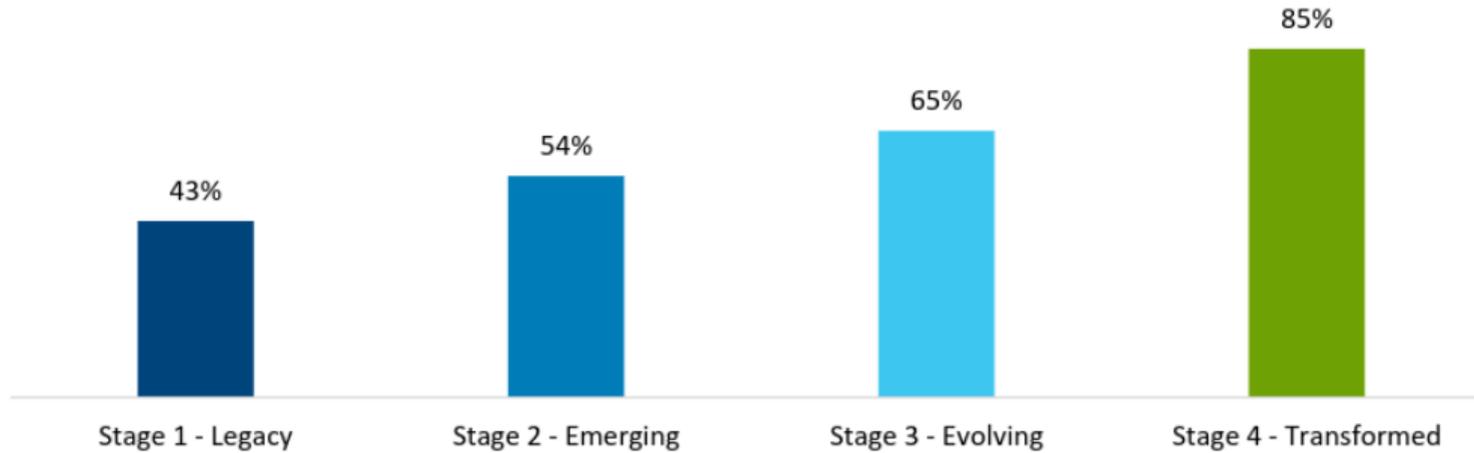
- Der Schmetterling erlebt eine Transformation, eine Metamorphose von der Raupe über die Puppe zum Schmetterling



Studie

Wettbewerbsfähigkeit

Percent of respondents believing their organization is in a *very strong* or *strong* position to compete and succeed in their market over the next few years, by IT Transformation Maturity



Source: Enterprise Strategy Group, 2017 [4]

2024-11-20

IT-Transformation
└ IT-Transformation
└ Studie



- Studie der Enterprise Strategy Group (ESG) hat im Auftrag von Dell EMC und Intel
- Umso reifer eine Organisation umso stärker glaubt sie, am Markt bestehen und sich in den nächsten Jahren behaupten zu können
- D.h. insgesamt reifere Organisationen investieren mehr in Innovationen und können schneller auf Veränderungen reagieren und sie setzen ihre Investitionen effizienter ein. Somit wird der Abstand zwischen den Organisationen immer größer
- Beispielsweise haben GAFAM (Google, Amazon, Facebook, Apple und Microsoft) einen großen Vorsprung an Cloudtechnologien



IT-Transformation

Definition

Definition IT-Transformation

IT-Transformation ist die vollständige Bestandsaufnahme, (Neu-)Bewertung und Überarbeitung der kompletten Unternehmens-IT mit dem Ziel, die passenden (und technischen) Voraussetzungen für andere, moderne Technologien oder Prozesse zu schaffen.[5]

2024-11-20

IT-Transformation
└ IT-Transformation
└ IT-Transformation

IT-Transformation
Definition

Definition IT-Transformation

IT-Transformation ist die vollständige Bestandsaufnahme, (Neu-)Bewertung und Überarbeitung der kompletten Unternehmens-IT mit dem Ziel, die passenden (und technischen) Voraussetzungen für andere, moderne Technologien oder Prozesse zu schaffen.[5]

- Die IT-Transformation ist Voraussetzung für die Digitale Transformation
- Wenn ich nur Faxgeräte als IT-Infrastruktur habe, kann ich Nutzer*innen keine Apps mit Selfservice anbieten
- Zu Beginn erfolgt normalerweise eine Analyse der bestehenden Infrastruktur und Ressourcen bis hinunter zu den einzelnen Prozessen der Organisation
- Ziel ist die Reduktion von Risiken und Abhängigkeit von veralteten Produkten oder Technologien



Drei Schwerpunkte der IT-Transformation

Nach Enterprise Strategy Group

- Modernisierung der wichtigen Rechenzentrumsressourcen
- Automatisierung der IT
- Transformation der Prozesse und der Zusammenarbeit zwischen Business und IT

2024-11-20

IT-Transformation

└ IT-Transformation

└ Drei Schwerpunkte der IT-Transformation

Drei Schwerpunkte der IT-Transformation
Nach Enterprise Strategy Group

- Modernisierung der wichtigen Rechenzentrumsressourcen
- Automatisierung der IT
- Transformation der Prozesse und der Zusammenarbeit zwischen Business und IT

- Modernisierung senkt operative Kosten und den Verwaltungsaufwand
- Neue Hardware verbraucht weniger Strom und reduziert somit Kosten und CO2 Fußabdruck
- Es muss nicht zwingend alles ausgetauscht werden. Nach der Analyse mit den wichtigsten Systemen beginnen und abwägen: It depends
- Automatisierung ist der Schlüssel für schnelle und gleichzeitig sichere Änderungen
- Automatisierung ist schwer. Es benötigt Expert*innen für Puppet, Ansible, Terraform, OpenTofu, Infrastruktur, Kubernetes...
- Transformation der Prozesse meint eine Verschlankeung/Vereinfachung der Prozesse und eine Anpassung an die beteiligten Personen

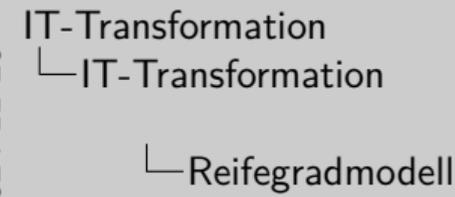


Reifegradmodell

Beispielfragen

- Is your IT organization committed to, interested in, or not interested in the utilization of software-defined networking technologies?
- Does your IT organization enable developers and/or line-of-business end-users to provision on-premises IT resources (VMs, storage capacity, network connectivity, etc.) in a self-service fashion?
- To whom does the most senior IT executive at your company report?

2024-11-20



Reifegradmodell

Beispielfragen

- Is your IT organization committed to, interested in, or not interested in the utilization of software-defined networking technologies?
- Does your IT organization enable developers and/or line-of-business end-users to provision on-premises IT resources (VMs, storage capacity, network connectivity, etc.) in a self-service fashion?
- To whom does the most senior IT executive at your company report?

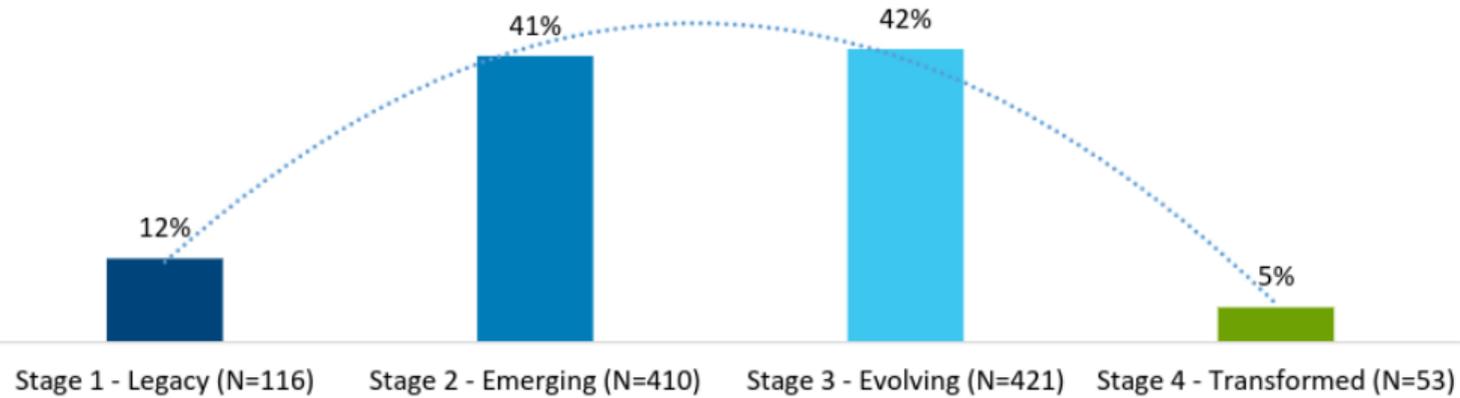
- Die Enterprise Strategy Group (ESG) hat im Auftrag von Dell EMC und Intel die drei Bereiche der IT-Transformation mit insgesamt 10 Fragen abgetestet
- Fünf zu Rechenzentren, drei zur Automatisierung und zwei zur Zusammenarbeit
- Basierend auf den Antworten gab es bis zu 100 Punkte



Reifegradmodell

Verteilung Unternehmen auf die vier Kategorien des Reifegradmodells

Respondent Organizations by IT Transformation Maturity Stage
(percent of respondents, N=1,000)



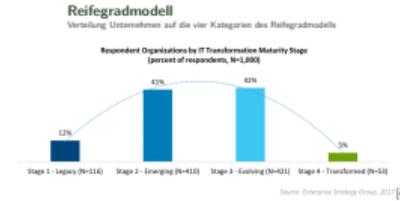
Source: Enterprise Strategy Group, 2017 [4]

2024-11-20

IT-Transformation

└ IT-Transformation

└ Reifegradmodell



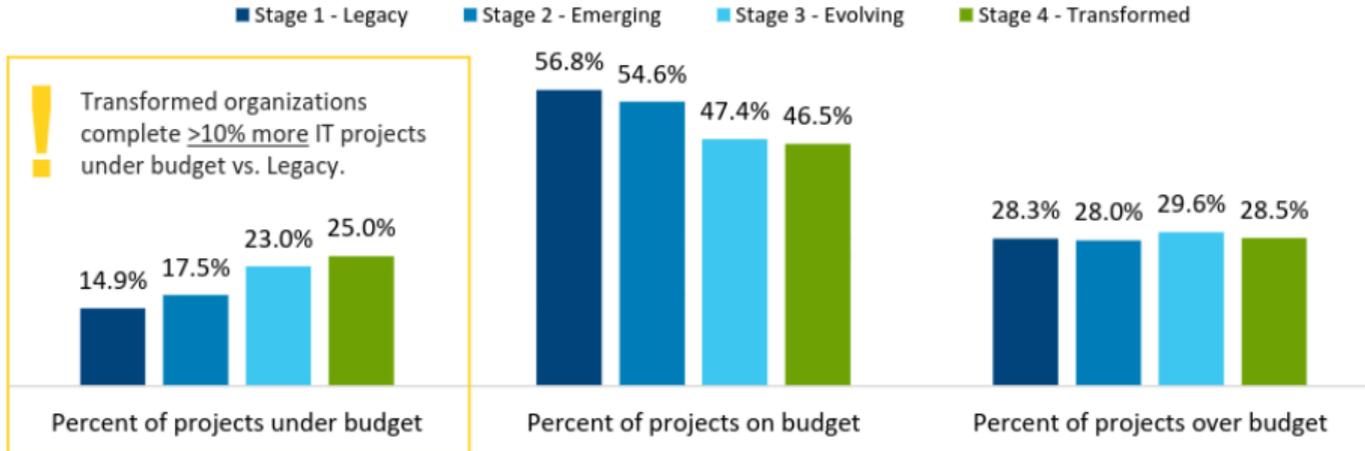
- Die 1000 teilnehmenden Unternehmen wurden dann in 25 Punkte Schritten eingeteilt in die vier Kategorien



Reifegradmodell

Vorteile für reifere Unternehmen: Kosten

Over the last few years, roughly what percent of your IT projects have been completed under, on, or over budget? (mean percent of projects)



Source: Enterprise Strategy Group, 2017 [4]

2024-11-20

IT-Transformation

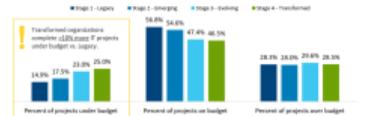
IT-Transformation

Reifegradmodell

Reifegradmodell

Vorteile für reifere Unternehmen: Kosten

Over the last few years, roughly what percent of your IT projects have been completed under, on, or over budget? (mean percent of projects)



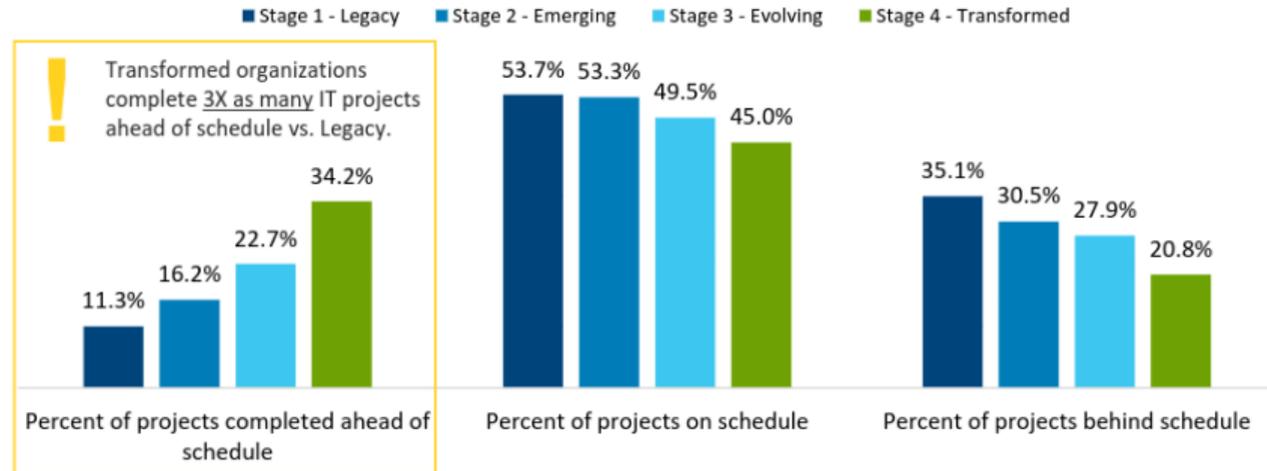
- Reifere Organisationen liegen häufiger (10 Prozentpunkte) unter dem geplanten Budget
- Die Quote der unterschätzten Projekte sinkt leider nicht signifikant
- Die Unternehmen geben tatsächlich 14 % weniger für wichtige Projekte aus, als unreife Organisationen, d.h. sie haben nicht nur einfach anders geplant



Reifegradmodell

Vorteile für reifere Unternehmen: Geschwindigkeit

Over the last few years, roughly what percent of your IT projects have been completed in each of the following timeframes? (mean percent of projects)

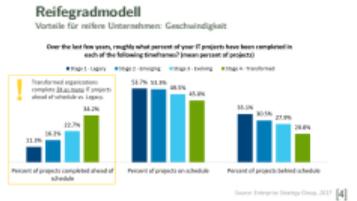


Source: Enterprise Strategy Group, 2017 [4]

2024-11-20

IT-Transformation
IT-Transformation

Reifegradmodell



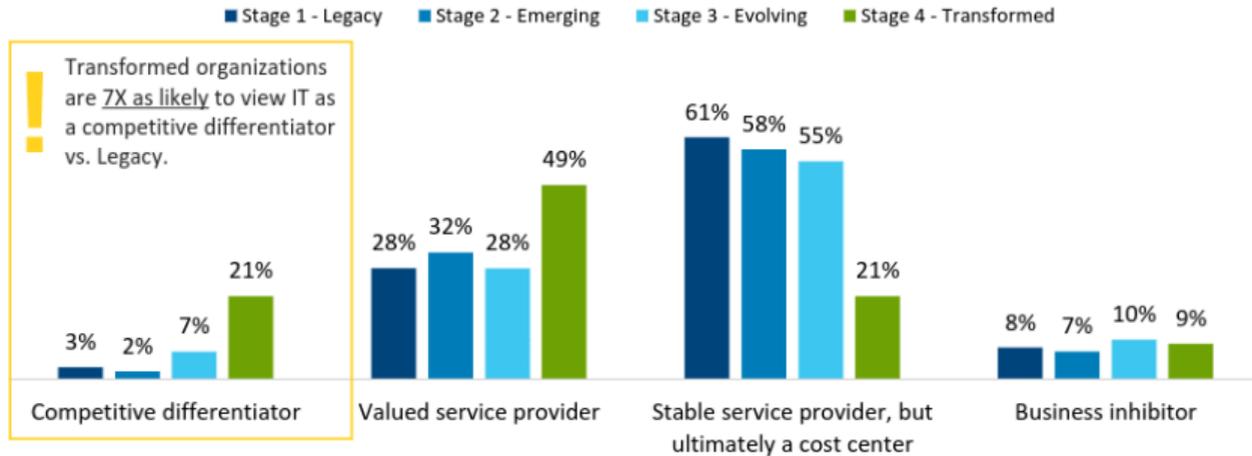
- Reifere Organisationen schließen dreimal so viele Projekte unter der geplanten Zeit ab wie unreife
- Der Anteil der zu spät abgeschlossenen Projekte sinkt um 15 Prozentpunkte



Reifegradmodell

Vorteile für reifere Unternehmen: IT als Wettbewerbsvorteil

In general, how would you characterize the way your line-of-business stakeholders currently view the IT organization? (Percent of respondents)

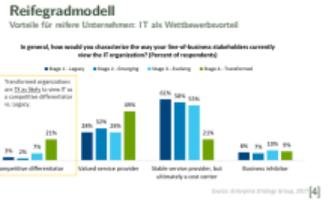


Source: Enterprise Strategy Group, 2017 [4]

2024-11-20

IT-Transformation
IT-Transformation

Reifegradmodell



- Reifere Organisationen erkennen IT als Wettbewerbsvorteil an und nicht als Kostenpunkt
- Das spiegelt sich wieder in der Wertschätzung der Mitarbeiter*innen und somit Gewinnung von Talenten
- Die reiferen Organisationen haben ihre Umsatzprognosen häufiger übertroffen und haben mehr in Innovationen investiert



Frameworks

Welche Frameworks unterstützen mich bei der IT-Transformation?



2024-11-20

IT-Transformation

└ Frameworks

└ Frameworks

- Frameworks nehmen uns Arbeit ab, in dem sie bewährte Vorgehensweisen und Bausteine bereitstellen



EAM-Frameworks

Auswahl an Enterprise Architecture Management Frameworks[6]

- ArchiMate
- Zachman-Framework
- TOGAF Standard (The Open Group Architecture Framework)
- ARIS (Architecture of Integrated Information Systems)
- DoDAF (Department of Defense Architecture Framework)
- NAF (The NATO Architecture Framework)

2024-11-20

IT-Transformation

└ Frameworks

└ EAM-Frameworks

EAM-Frameworks
Auswahl an Enterprise Architecture Management Frameworks[6]

- ArchiMate
- Zachman-Framework
- TOGAF Standard (The Open Group Architecture Framework)
- ARIS (Architecture of Integrated Information Systems)
- DoDAF (Department of Defense Architecture Framework)
- NAF (The NATO Architecture Framework)

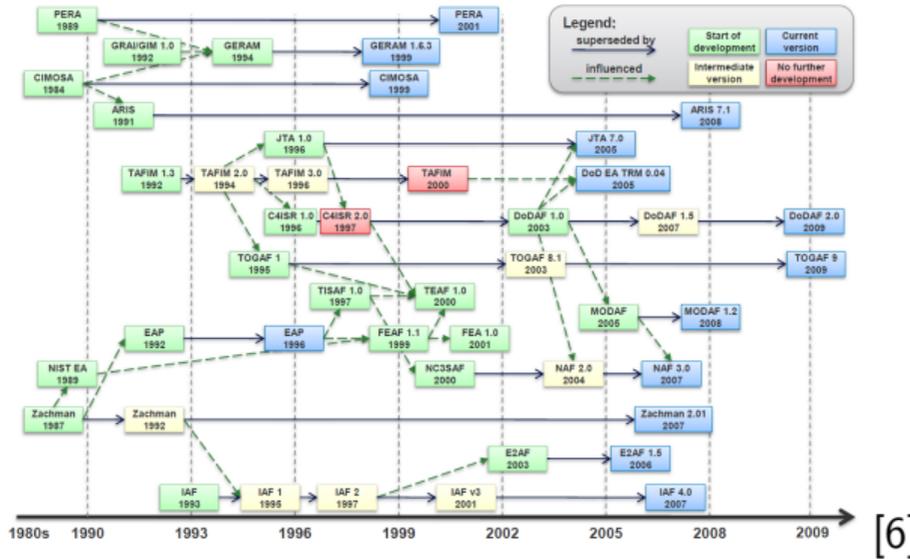
- Zachman ist bei IBM entstanden und wurde 1992 von Sowa und Zachman veröffentlicht
- DoDAF wurde 2003 als Regierungsprojekt des US-Verteidigungsministeriums veröffentlicht
- TOGAF ging aus TAFIM hervor und wurde 1995 das erste mal beschrieben. Inzwischen Version 10



EAM-Frameworks

Genealogie der EAM-Frameworks bis 2008

Introducing EAM in an enterprise is a challenge:
EA frameworks provide only limited support



[6]

2024-11-20

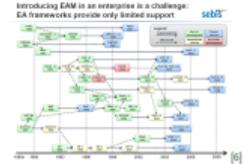
IT-Transformation

└ Frameworks

└ EAM-Frameworks

EAM-Frameworks

Genealogie der EAM-Frameworks bis 2008

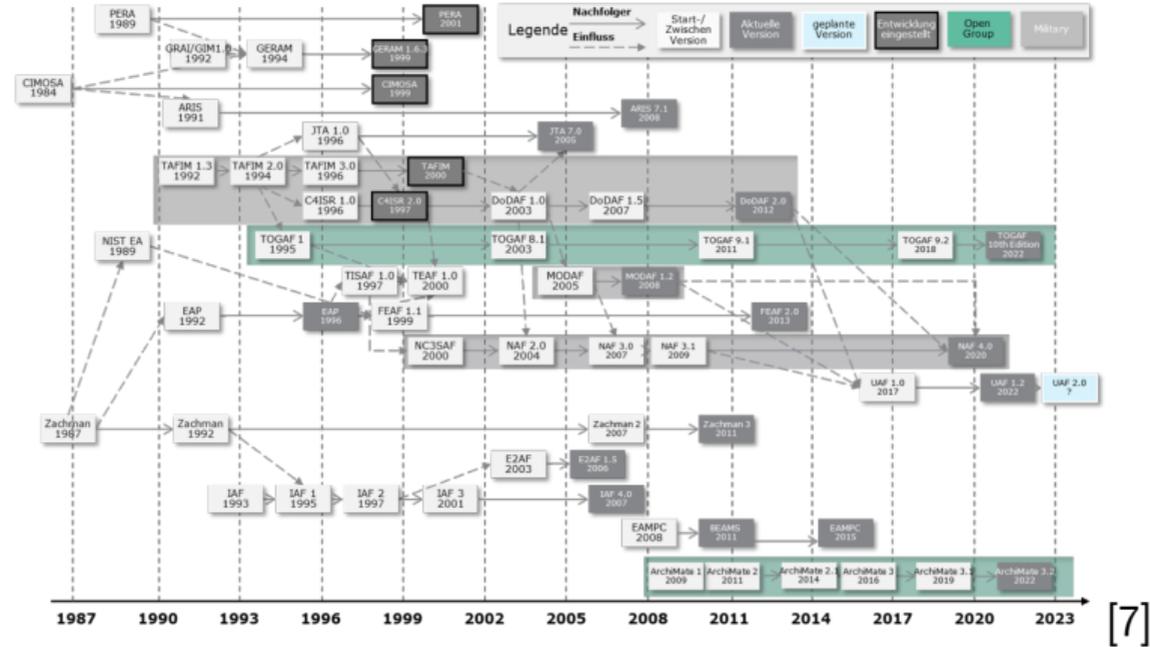


- Von sebis. Quelle unter anderem das Buch Applying Frameworks[6]



EAM-Frameworks

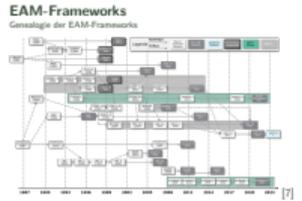
Genealogie der EAM-Frameworks



[7]

IT-Transformation
└ Frameworks
└ EAM-Frameworks

2024-11-20

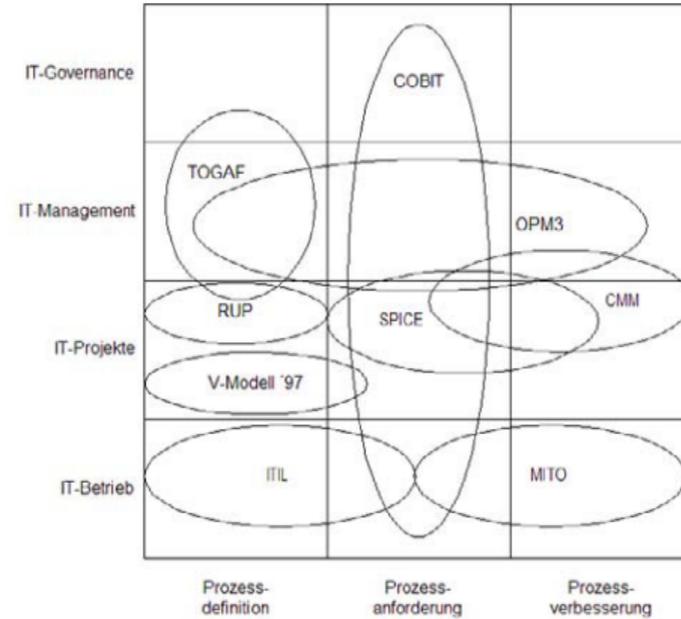


- Von dem Consultant Rolf Knoll für seine TOGAF Schulungen zusammengestellt. Veröffentlicht in einem LinkedIn Threat[7]



Ordnungsrahmen

Anwendungsbereiche von EAM-Frameworks



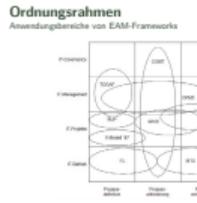
[8]

2024-11-20

IT-Transformation

└ Frameworks

└ Ordnungsrahmen



- Aus der Präsentation Erarbeitung von Blueprints mit Architekturframeworks von Marten Schönherr[8]



Weitere Frameworks

Frameworks für andere Ordnungsrahmen

- IT-Service-Management: ITIL 4[9]
- IT-Service-Management: ISO/IEC 20000[10]
- IT-Governance: COBIT 5[11]
- Qualitätsmanagement: ISO 9000 ff.[12]

2024-11-20

IT-Transformation
└ Frameworks

└ Weitere Frameworks

Weitere Frameworks
Frameworks für andere Ordnungsrahmen

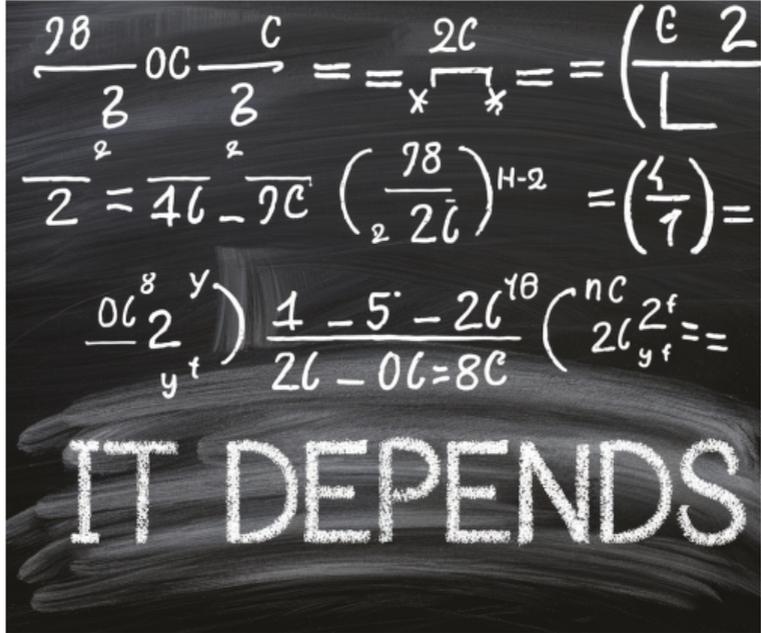
- IT-Service-Management: ITIL 4[9]
- IT-Service-Management: ISO/IEC 20000[10]
- IT-Governance: COBIT 5[11]
- Qualitätsmanagement: ISO 9000 ff.[12]

- ITIL stammt von der Britischen Regierung aus den 1980ern. Wurde seitdem weiterentwickelt. Framework für IT-Service-Management
- ISO/IEC 20000 geht auf British Standard BS 15000 zurück und wurde 2005 veröffentlicht. Die ISO dient als messbarer Qualitätsstandard für IT-Service-Management
- COBIT wurde 1996 vom internationalen Verband der IT-Prüfer entwickelt. Framework für IT-Governance
- ISO 9000 geht auf British Standard BS 5750 aus dem Jahre 1979 zurück. 1987 wurde die ISO 8402 eingeführt, die im Jahr 2000 von der ISO 9000 abgelöst wurde
- 1994 wurden die Normen ISO 9001, 9002 und 9003 verabschiedet. Nur noch 9001 ist von diesen drei gültig



Wahl des Frameworks

Individuell für jede Organisation



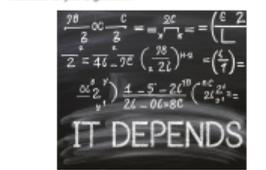
2024-11-20

IT-Transformation

└ Frameworks

└ Wahl des Frameworks

Wahl des Frameworks



- Consultant Sprech: It Depends
- Es gibt leider keine einfache Faustformel, welches Framework das richtige ist



Architektur

Welche Architekturen auf welchen Ebenen stehen für eine Transformation zur Verfügung?

2024-11-20

IT-Transformation

└─ Architektur

└─ Architektur

Architektur

Welche Architekturen auf welchen Ebenen stehen für eine Transformation zur Verfügung?



Architektur

Es gibt immer eine Architektur

Es gibt nicht keine Architektur.

Sie ist immer da und im schlimmsten Fall ein Big Ball of Mud[13].

2024-11-20

IT-Transformation

└─ Architektur

└─ Architektur

Architektur
Es gibt immer eine Architektur

Es gibt nicht keine Architektur.
Sie ist immer da und im schlimmsten Fall ein Big Ball of Mud[13].

- Der Ausdruck Big Ball of Mud wurde von Brian Foote und Joseph Yoder in ihrem 1999 erschienenen Artikel selben Namens geprägt.
- Entsteht wenn die Architektur nicht aktiv definiert oder deren Einhaltung forciert wird.



Architektur

Definition

Definition Architektur

Architektur ist eine Beschreibung von IT-, Fach- und Betriebsarchitektur einschließlich der technischen Basis, auf der IT-Verfahren oder Software zur Umsetzung der festgelegten Anforderungen bereitgestellt werden müssen.[14]

2024-11-20

IT-Transformation
└─ Architektur

└─ Architektur

Architektur
Definition

Definition Architektur

Architektur ist eine Beschreibung von IT-, Fach- und Betriebsarchitektur einschließlich der technischen Basis, auf der IT-Verfahren oder Software zur Umsetzung der festgelegten Anforderungen bereitgestellt werden müssen.[14]

- Siehe § 2 - KONSENS-Gesetz (KONSENS-G) Absatz 6
- Wir werden uns auf den Bereich IT-Architektur konzentrieren
- Der fachliche Teil ist zu spezifisch für den jeweiligen Einsatzbereich
- Betriebsarchitektur eher relevant für Administrator*innen oder Infrastrukturplaner*innen



Architekturebenen

Wie granular darf/soll es sein?

- IT-Architektur: Zusammenspiel aller IT-Systeme einer Organisation
- IT-Systemarchitektur: Aufbau eines IT-Systems aus Komponenten (Hardware und Software) und deren Zusammenspiel
- Softwarearchitektur: Funktionsweise von Computerprogrammen, Daten und deren Zusammenspiel

2024-11-20

IT-Transformation

└─ Architektur

└─ Architekturebenen

Architekturebenen

Wie granular darf/soll es sein?

- IT-Architektur: Zusammenspiel aller IT-Systeme einer Organisation
- IT-Systemarchitektur: Aufbau eines IT-Systems aus Komponenten (Hardware und Software) und deren Zusammenspiel
- Softwarearchitektur: Funktionsweise von Computerprogrammen, Daten und deren Zusammenspiel

- Siehe Wikipedia Artikel Architektur (Informatik)[15]
- Es gibt auch noch Datenarchitekturen, Netzwerkarchitekturen, Rechnerarchitekturen, Prozessorarchitekturen und weitere
- Wir werden uns Top-Down durch die Schichten arbeiten
- Von einigen dieser Begriffe gibt es über 150 Definitionen



IT-Architektur

Definition

Definition IT-Architektur

[IT-Architektur ist] die Gesamtheit der technischen Aspekte der Informationstechnologie in einem Unternehmen[...]. Dabei wird sowohl die Hardware als auch die Software betrachtet.[...] Diese Komponenten sind durch Geschäftsprozesse und Datenbeziehungen verbunden.[16]

2024-11-20

IT-Transformation
└─ Architektur

└─ IT-Architektur

IT-Architektur
Definition

Definition IT-Architektur
[IT-Architektur ist] die Gesamtheit der technischen Aspekte der Informationstechnologie in einem Unternehmen[...]. Dabei wird sowohl die Hardware als auch die Software betrachtet.[...] Diese Komponenten sind durch Geschäftsprozesse und Datenbeziehungen verbunden.[16]

- Siehe Artikel IT-Architektur des Transformationsdienstleisters Natuvion[16]
- Konkret sind damit die Gestaltung der Infrastruktur wie Hardware, Netzwerke und Standorte sowie Softwarelösungen, Schnittstellen und Steuerungsmechanismen zur Verwaltung der IT-Ressourcen gemeint
- Diese umfassen Themen wie Konfiguration, Ausfallsicherheit, Datenschutz und Katastrophenmanagement
- Die Planung und Umsetzung einer durchdachten IT-Architektur erfolgt immer im Hinblick auf eine optimale Unterstützung der Geschäftsstrategie und einen reibungslosen Geschäftsbetrieb
- Enterprise Architektur ist nochmals etwas abstrakter und beschreibt eher das Zusammenspiel von Geschäftsprozessen und IT
- Beispiel für Enterprise Architektur Framework ist TOGAF (The Open Group Architecture Framework)
- Schwerpunkt liegt auf der Unternehmens-/Organisations-Sicht
- Ausgeprägter strategischer Fokus, geringer technischer Fokus



IT-Architektur

Beispiele für IT-Architekturen

- Private Cloud mit Microservices
- On-Premises mit Monolithen und Microservices
- Mainframe mit Thin Clients

2024-11-20

IT-Transformation
└─ Architektur

└─ IT-Architektur

IT-Architektur
Beispiele für IT-Architekturen

- Private Cloud mit Microservices
- On-Premises mit Monolithen und Microservices
- Mainframe mit Thin Clients



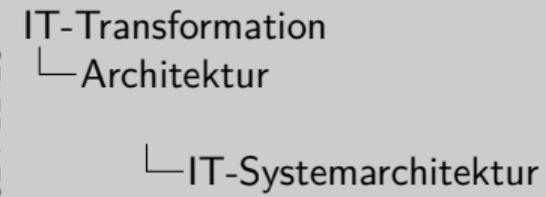
IT-Systemarchitektur

Definition

Definition IT-Systemarchitektur

[IT-Systemarchitektur beschreibt die] Gesamtheit aller Systemkomponenten und deren Interaktionen [und definiert] wie die einzelnen Teile des Systems zusammenarbeiten, um die gewünschte Funktionalität zu erreichen.[17]

2024-11-20



IT-Systemarchitektur
Definition

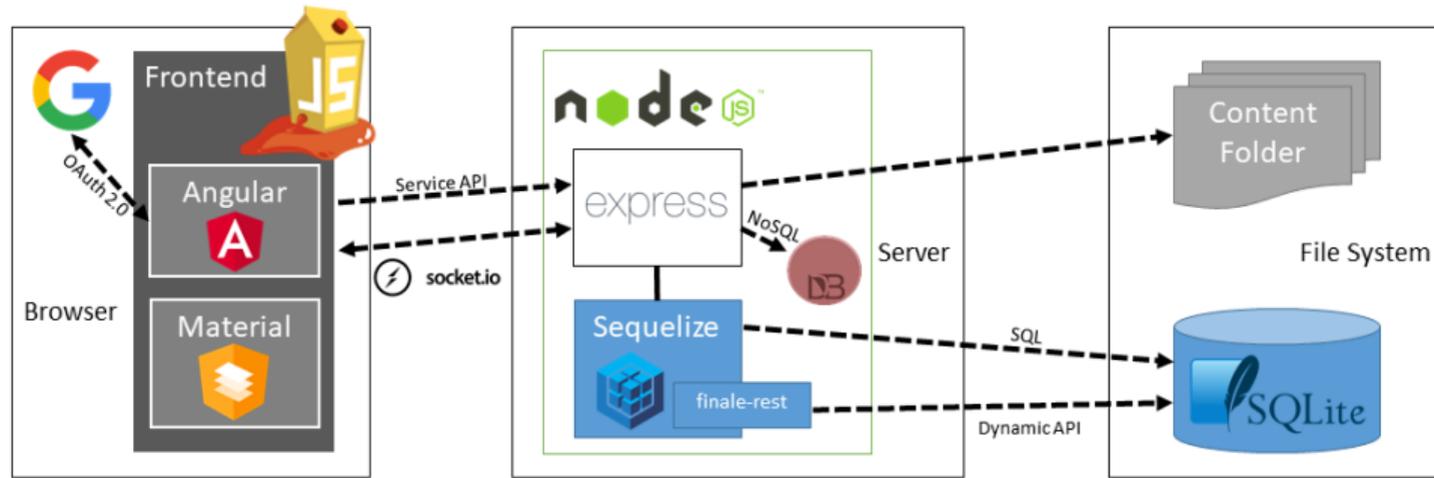
Definition IT-Systemarchitektur
[IT-Systemarchitektur beschreibt die] Gesamtheit aller Systemkomponenten und deren Interaktionen [und definiert] wie die einzelnen Teile des Systems zusammenarbeiten, um die gewünschte Funktionalität zu erreichen [17]

- Siehe Artikel Systemarchitektur: Das Fundament einer erfolgreichen Software-Konzeption des Transformationsdienstleisters LINKITSYSTEMS[17]
- Im Gegensatz zur IT-Architektur beschäftigt sich die Systemarchitektur nur mit einem System
- Schwerpunkt liegt auf dem System
- Mittlerer strategischer Fokus, mittlerer technischer Fokus
- Betroffene Systeme z.B. OLAP (Online Transaction Processing)[18] oder OLTP(Online Analytical Processing)[19]



IT-Systemarchitektur

Beispielfür Systemarchitektur

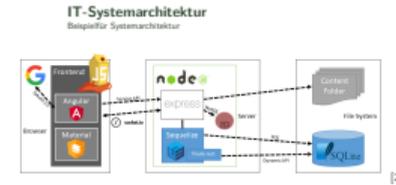


[20]

2024-11-20

IT-Transformation
└─ Architektur

└─ IT-Systemarchitektur



- Darstellung der Architektur des Juice Shop von OWASP[21]



Softwarearchitektur

Definition

Definition Softwarearchitektur

Eine Softwarearchitektur [...] beschreibt die grundlegenden Komponenten und deren Zusammenspiel innerhalb eines Softwaresystems.[22]

2024-11-20

IT-Transformation
└─ Architektur

└─ Softwarearchitektur

Softwarearchitektur
Definition

Definition Softwarearchitektur
Eine Softwarearchitektur [...] beschreibt die grundlegenden Komponenten und deren Zusammenspiel innerhalb eines Softwaresystems.[22]

- Siehe Artikel Systemarchitektur: Das Fundament einer erfolgreichen Software-Konzeption des Transformationsdienstleisters LINKITSYSTEMS[22]
- Im Gegensatz zur IT-Systemarchitektur geht es in der Softwarearchitektur nur um die Gestaltung einer Software als Teil eines Systems
- Schwerpunkt liegt auf der Software
- Geringer strategischer Fokus, hoher technischer Fokus



Architektur

Unschärfe

Je nach Organisation oder Unternehmen gibt es eine Unschärfe zwischen diesen Ebenen, den zuständigen Rollen und der Gestaltung.

2024-11-20

IT-Transformation

└─ Architektur

└─ Architektur

Architektur
Unschärfe

Je nach Organisation oder Unternehmen gibt es eine Unschärfe zwischen diesen Ebenen, den zuständigen Rollen und der Gestaltung.

- Kleinere Unternehmen haben oft keine*n explizite*n Architekt*in.
- Softwareentwicklung entscheidet selbst über Strukturen und CTO oder Geschäftsführung gibt einen Rahmen vor.
- In Konzernen oder größeren Strukturen gibt es eine genaue Definition von Rollen und Entscheidungshirarchien.



Cloud oder On-Premises

Was macht Cloud Computing aus und sollte ich lieber On-Premises bleiben?

2024-11-20

IT-Transformation
└─ Architektur

└─ Cloud oder On-Premises

Cloud oder On-Premises

Was macht Cloud Computing aus und sollte ich lieber On-Premises bleiben?



Cloud

Umgangssprachliche Definition

Definition von Cloud

Die Cloud ist der Computer von jemand anderem

2024-11-20

IT-Transformation

└─ Architektur

└─ Cloud

Cloud

Umgangssprachliche Definition

Definition von Cloud

Die Cloud ist der Computer von jemand anderem

- Scherzhafte Definition
- In Bezug auf Datenschutz und -sicherheit aber eine wichtige Perspektive



Cloud vs. On-Premises

Unterscheidung

Cloud

Cloud Computing beschreibt ein Modell, das bei Bedarf – meistens über das Internet und geräteunabhängig – zeitnah und mit wenig Aufwand geteilte Computerressourcen als Dienstleistung, etwa in Form von Servern, Datenspeicher oder Applikationen, bereitstellt und nach Nutzung abrechnet.[23]

On-Premises

On-Premises oder On-Prem (in den eigenen Räumlichkeiten, vor Ort oder lokal) bezeichnet ein Nutzungs- und Lizenzmodell für Computerprogramme. Bei [...] On-Prem-Software erwirbt oder mietet der Lizenznehmer Software und betreibt diese in eigener Verantwortung auf eigener Hardware.[24]

2024-11-20

IT-Transformation

└─ Architektur

└─ Cloud vs. On-Premises

- Bis ca. 2010 war die lokale Nutzung bzw. die Lizenzierung für die lokale Nutzung von Software der Normalfall und hatte daher keine besondere Bezeichnung
- Erst seitdem die lokale Nutzung zunehmend von Software as a Service (SaaS) oder Cloud Computing verdrängt wird, ist der Begriff Off-Premises als Antonym entstanden
- Noch etwas ungenau als Definition

Cloud vs. On-Premises
Unterscheidung

Cloud

Cloud Computing beschreibt ein Modell, das bei Bedarf – meistens über das Internet und geräteunabhängig – zeitnah und mit wenig Aufwand geteilte Computerressourcen als Dienstleistung, etwa in Form von Servern, Datenspeicher oder Applikationen, bereitstellt und nach Nutzung abrechnet.[23]

On-Premises

On-Premises oder On-Prem (in den eigenen Räumlichkeiten, vor Ort oder lokal) bezeichnet ein Nutzungs- und Lizenzmodell für Computerprogramme. Bei [...] On-Prem-Software erwirbt oder mietet der Lizenznehmer Software und betreibt diese in eigener Verantwortung auf eigener Hardware.[24]



Cloud

Die fünf Merkmale von Cloud Computing nach NIST[25]

On-demand self-service

Der Nutzer kann Leistungen selbst und ohne menschliche Interaktion bei Bedarf bereitstellen lassen (Selbstbedienung).

Broad network access

Die Leistungen sind mit Standardmechanismen über das Netzwerk erreichbar, die eine Nutzung mit heterogenen Endgeräten wie Smartphones, Tablets, Notebooks oder Workstations fördern.

2024-11-20

IT-Transformation

└─ Architektur

└─ Cloud

- Definition des National Institute of Standards and Technology

Cloud

Die fünf Merkmale von Cloud Computing nach NIST[25]

On-demand self-service

Der Nutzer kann Leistungen selbst und ohne menschliche Interaktion bei Bedarf bereitstellen lassen (Selbstbedienung).

Broad network access

Die Leistungen sind mit Standardmechanismen über das Netzwerk erreichbar, die eine Nutzung mit heterogenen Endgeräten wie Smartphones, Tablets, Notebooks oder Workstations fördern.



Cloud

Die fünf Merkmale von Cloud Computing nach NIST[25]

Resource pooling

Die Computerressourcen des Anbieters werden zusammengefasst, um mehrere Nutzer nach dem Mandantenprinzip bedarfsgerecht bedienen zu können. Damit einhergehend hat der Nutzer in der Regel keine Kontrolle darüber, mit welchen Computerressourcen die Leistung erbracht wird (z.B. auf welchem Server oder mit welcher Datenbank).

Rapid elasticity

Die Leistung kann dynamisch bereitgestellt und freigegeben werden, um bedarfsgerecht skalieren zu können. Aus Sicht des Nutzers scheinen die verfügbaren Computerressourcen unbegrenzt und Leistungen können jederzeit in beliebiger Höhe angepasst werden.

2024-11-20

IT-Transformation

└─ Architektur

└─ Cloud

- Definition des National Institute of Standards and Technology

Cloud

Die fünf Merkmale von Cloud Computing nach NIST[25]

Resource pooling

Die Computerressourcen des Anbieters werden zusammengefasst, um mehrere Nutzer nach dem Mandantenprinzip bedarfsgerecht bedienen zu können. Damit einhergehend hat der Nutzer in der Regel keine Kontrolle darüber, mit welchen Computerressourcen die Leistung erbracht wird (z.B. auf welchem Server oder mit welcher Datenbank).

Rapid elasticity

Die Leistung kann dynamisch bereitgestellt und freigegeben werden, um bedarfsgerecht skalieren zu können. Aus Sicht des Nutzers scheinen die verfügbaren Computerressourcen unbegrenzt und Leistungen können jederzeit in beliebiger Höhe angepasst werden.



Cloud

Die fünf Merkmale von Cloud Computing nach NIST[25]

Measured service

Cloud-Systeme steuern und optimieren Computerressourcen abhängig von der jeweiligen Leistung anhand von messbaren Zahlen. Das können Speicher, Bandbreite oder aktive Benutzerkonten sein. Die Nutzung der Leistung kann überwacht, gesteuert und als Bericht bereitgestellt werden, um die Transparenz für den Nutzer wie auch den Anbieter zu gewährleisten.

2024-11-20

IT-Transformation

└─ Architektur

└─ Cloud

Cloud

Die fünf Merkmale von Cloud Computing nach NIST[25]

Measured service

Cloud-Systeme steuern und optimieren Computerressourcen abhängig von der jeweiligen Leistung anhand von messbaren Zahlen. Das können Speicher, Bandbreite oder aktive Benutzerkonten sein. Die Nutzung der Leistung kann überwacht, gesteuert und als Bericht bereitgestellt werden, um die Transparenz für den Nutzer wie auch den Anbieter zu gewährleisten.

- Definition des National Institute of Standards and Technology



Deployment Modelle

Vier Modelle nach NIST[25]

Private Cloud

Die Cloudinfrastruktur ist exklusiv für eine Organisation, einschließlich deren Unterorganisationen (z.B. Abteilungen) provisioniert. Sie kann durch die Organisation, einem Dienstleister oder einer Kombination daraus betrieben werden. Die Private Cloud kann On- oder Off-Premise betrieben werden.

Community Cloud

Die Cloudinfrastruktur ist exklusiv für eine Gruppe von Organisation, einschließlich deren Unterorganisationen (z.B. Abteilungen) provisioniert, die gemeinsame Ziele verfolgen oder gemeinsame Anforderungen haben. Sie kann durch die Organisation, einem Dienstleister oder einer Kombination daraus betrieben werden. Die Private Cloud kann On- oder Off-Premise betrieben werden.

2024-11-20

IT-Transformation

└─Architektur

└─Deployment Modelle

Deployment Modelle
Vier Modelle nach NIST[25]

Private Cloud

Die Cloudinfrastruktur ist exklusiv für eine Organisation, einschließlich deren Unterorganisationen (z.B. Abteilungen) provisioniert. Sie kann durch die Organisation, einem Dienstleister oder einer Kombination daraus betrieben werden. Die Private Cloud kann On- oder Off-Premise betrieben werden.

Community Cloud

Die Cloudinfrastruktur ist exklusiv für eine Gruppe von Organisation, einschließlich deren Unterorganisationen (z.B. Abteilungen) provisioniert, die gemeinsame Ziele verfolgen oder gemeinsame Anforderungen haben. Sie kann durch die Organisation, einem Dienstleister oder einer Kombination daraus betrieben werden. Die Private Cloud kann On- oder Off-Premise betrieben werden.

- On- oder Off-Premise: Z.B. Hardware im eigenen Rechenzentrum mit OpenStack ist On-Premise. Ein Dienstleister, der eine OpenStack Umgebung exklusiv zur Verfügung stellt Off-Premise
- Community Cloud: Z.B. verschiedene Instanzen von Einwohnermeldeämtern. Oder ein Konzern mit mehreren Töchtern wie z.B. Stellantis, Opel, Peugeot...



Deployment Modelle

Vier Modelle nach NIST[25]

Public Cloud

Die Cloudinfrastruktur ist für alle zugänglich provisioniert. Sie kann z.B. durch eine Firma, einen Verein oder eine Universität betrieben werden. Die Private Cloud wird auf den Systemen des Anbieters betrieben.

Hybrid Cloud

Die Cloudinfrastruktur ist aus zwei oder mehr der bisher genannten Cloud Modelle zusammengesetzt. Dabei sind die einzelnen Cloudinstanzen getrennt voneinander und über Schnittstellen miteinander verbunden, so dass Daten und Anwendungen migriert werden können.

2024-11-20

IT-Transformation

└─Architektur

└─Deployment Modelle

- Public Cloud sind die Hyperscaler wie Google, Microsoft, Amazon aber auch deutsche Unternehmen wie StackIT von der Schwarz Unternehmensgruppe
- Hybrid bietet z.B. die Möglichkeit den Standardworkload mit eigenen Servern zu bestreiten und bei Peaks auf Public Cloud zu setzen

Deployment Modelle
Vier Modelle nach NIST[25]

Public Cloud

Die Cloudinfrastruktur ist für alle zugänglich provisioniert. Sie kann z.B. durch eine Firma, einen Verein oder eine Universität betrieben werden. Die Private Cloud wird auf den Systemen des Anbieters betrieben.

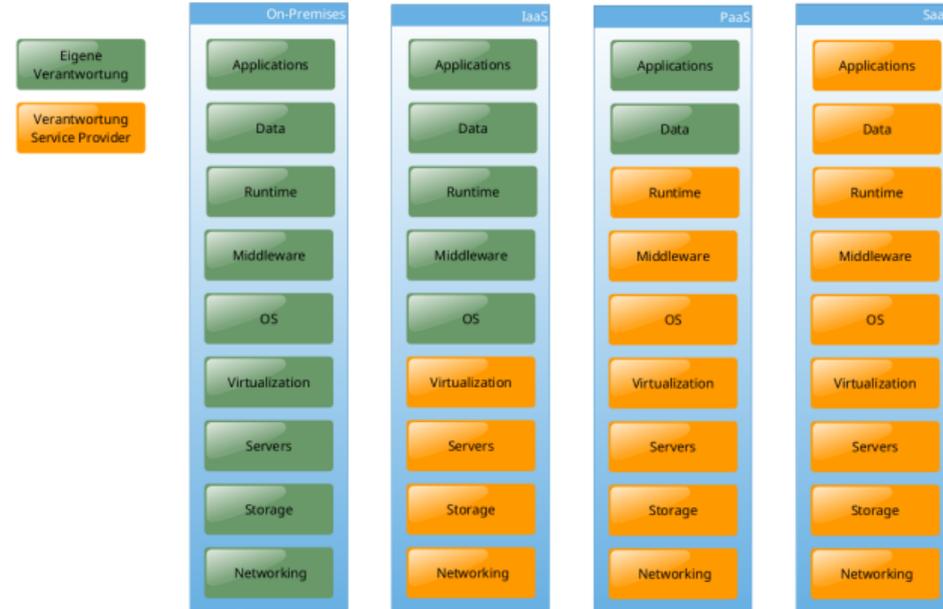
Hybrid Cloud

Die Cloudinfrastruktur ist aus zwei oder mehr der bisher genannten Cloud Modelle zusammengesetzt. Dabei sind die einzelnen Cloudinstanzen getrennt voneinander und über Schnittstellen miteinander verbunden, so dass Daten und Anwendungen migriert werden können.



Service Modelle

Vier Service Modelle



2024-11-20

IT-Transformation

└─ Architektur

└─ Service Modelle



- On Premises ist dann ein Cloudmodell, wenn nach den 5 Kriterien des NIST eine Cloudinfrastruktur aufgebaut wurde
- Infrastructure as a Service bietet mehr Flexibilität bzgl. kurzfristiger Serverressourcen
- Platform as a Service bietet einfachen Start für eigene Entwicklungsprojekte
- Software as a Service ist für den Konsum von Dienstleistungen/Produkten für Kunden am einfachsten
- Das National Institute of Standards and Technology beschreibt die drei Modelle IaaS, PaaS und SaaS[25]
- Es gibt auch Sonderformen wie Function as a Service (FaaS), Bare Metal as a Service (BMAas), Artificial Intelligence as a Service (AIaaS), Data Intensive Computing as a Service (DICaaS), High Performance Computing as a Service (HPCaaS) und sogar Cybercrime as a Service



Software as a Service (SaaS)

Definition

Software as a Service (SaaS)

Bei Software as a Service (SaaS) können Benutzer*innen fertige Softwareanwendungen nutzen, ohne sich um die zugrunde liegende Infrastruktur kümmern zu müssen. Dies ermöglicht es Unternehmen und Einzelpersonen, auf leistungsstarke Anwendungen zuzugreifen, ohne diese selbst installieren und warten zu müssen.



2024-11-20

IT-Transformation

└─ Architektur

└─ Software as a Service (SaaS)

Software as a Service (SaaS)

Definition

Software as a Service (SaaS)

Bei Software as a Service (SaaS) können Benutzer*innen fertige Softwareanwendungen nutzen, ohne sich um die zugrunde liegende Infrastruktur kümmern zu müssen. Dies ermöglicht es Unternehmen und Einzelpersonen, auf leistungsstarke Anwendungen zuzugreifen, ohne diese selbst installieren und warten zu müssen.



- Angelehnt an The NIST Definition of CloudComputing[25]
- Beispiele sind Office365 oder Wikipedia



Software as a Service (SaaS)

Vorteile

- Geringe Investitionskosten
- Einfache Wartung/Aktualisierung
- Flexibilität
- Zugänglichkeit
- Kurze Implementationszeit
- Kompatibilität im Reich des Anbieters

2024-11-20

IT-Transformation

└─ Architektur

└─ Software as a Service (SaaS)

Software as a Service (SaaS)

Vorteile

- Geringe Investitionskosten
- Einfache Wartung/Aktualisierung
- Flexibilität
- Zugänglichkeit
- Kurze Implementationszeit
- Kompatibilität im Reich des Anbieters

- Bei Datenschutz und IT-Sicherheit muss man sich auf die Aussagen des Anbieters verlassen



Software as a Service (SaaS)

Nachteile

- Kompatibilität außerhalb des Anbieters
- Eingeschränkte Kontrolle und Anpassbarkeit
- Abhängigkeit vom Internet
- Datenportabilität
- Langfristige Kosten

2024-11-20

IT-Transformation

└─ Architektur

└─ Software as a Service (SaaS)

Software as a Service (SaaS)

Nachteile

- Kompatibilität außerhalb des Anbieters
- Eingeschränkte Kontrolle und Anpassbarkeit
- Abhängigkeit vom Internet
- Datenportabilität
- Langfristige Kosten

- Stichwort Vendor-Locking bei Microsoft
- Viele Unternehmen kommen wegen DSGVO nicht in Frage: Google, Facebook, Microsoft
- Siehe Europäischem Datenschutzbeauftragten



Platform as a Service (PaaS)

Definition

Platform as a Service (PaaS)

Bei Platform as a Service (PaaS) bekommen Benutzer*innen Anwendungsplattformen bereitgestellt, auf denen sie eigene Anwendungen entwickeln und betreiben können. Dies ermöglicht es Entwicklern*innen, sich auf die Entwicklung von Anwendungen zu konzentrieren, ohne sich um die zugrunde liegende Infrastruktur kümmern zu müssen.



2024-11-20

IT-Transformation

└─ Architektur

└─ Platform as a Service (PaaS)

Platform as a Service (PaaS)

Definition

Platform as a Service (PaaS)

Bei Platform as a Service (PaaS) bekommen Benutzer*innen Anwendungsplattformen bereitgestellt, auf denen sie eigene Anwendungen entwickeln und betreiben können. Dies ermöglicht es Entwicklern*innen, sich auf die Entwicklung von Anwendungen zu konzentrieren, ohne sich um die zugrunde liegende Infrastruktur kümmern zu müssen.



- Angelehnt an The NIST Definition of CloudComputing[25]
- Beispiele sind Red Hat OpenShift, Kubernetes, IBM Cloud Foundry oder Heroku



Platform as a Service (PaaS)

Vorteile

- Schneller und einfacher Einstieg in die Entwicklung
- Skalierbarkeit
- Anpassbarkeit der eigenen Anwendungen auf der Plattform
- Initiale Kosten

2024-11-20

IT-Transformation

└─ Architektur

└─ Platform as a Service (PaaS)

Platform as a Service (PaaS)
Vorteile

- Schneller und einfacher Einstieg in die Entwicklung
- Skalierbarkeit
- Anpassbarkeit der eigenen Anwendungen auf der Plattform
- Initiale Kosten



Platform as a Service (PaaS)

Nachteile

- Kein Einfluss auf die Plattform bzw. Infrastruktur
- Nur angebotene Schnittstellen und Services
- Je nach Plattform Migrationsprobleme bzw. kein Multicloud
- Langfristige Kosten
- Leistung

2024-11-20

IT-Transformation

└─ Architektur

└─ Platform as a Service (PaaS)

- Sonderwünsche können nicht berücksichtigt werden
- Stichwort Vendor-Locking
- Kosten können bei automatischer Skalierung unerwartet hoch ausfallen
- Wenig Einfluss auf die Leistung der zugrundeliegenden Infrastruktur

Platform as a Service (PaaS)

Nachteile

- Kein Einfluss auf die Plattform bzw. Infrastruktur
- Nur angebotene Schnittstellen und Services
- Je nach Plattform Migrationsprobleme bzw. kein Multicloud
- Langfristige Kosten
- Leistung

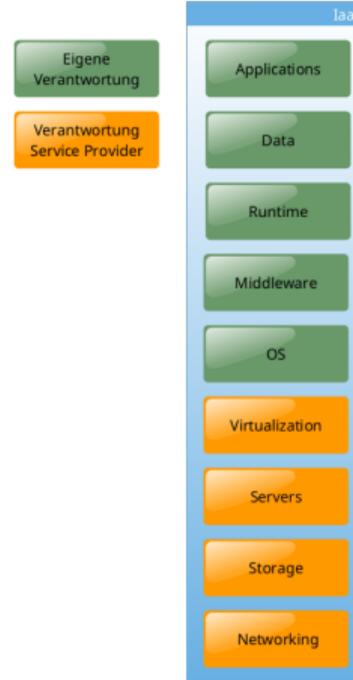


Infrastructure as a Service (IaaS)

Definition

Infrastructure as a Service (IaaS)

Bei Infrastructure as a Service (IaaS) erhalten Benutzer*innen Zugriff auf virtuelle Maschinen, Speicher und Netzwerkressourcen. Dies ermöglicht es ihnen, ihre eigene IT-Infrastruktur aufzubauen und zu verwalten, ohne physische Hardware besitzen zu müssen.



2024-11-20

IT-Transformation

└─ Architektur

└─ Infrastructure as a Service (IaaS)

Infrastructure as a Service (IaaS)

Definition

Infrastructure as a Service (IaaS)

Bei Infrastructure as a Service (IaaS) erhalten Benutzer*innen Zugriff auf virtuelle Maschinen, Speicher und Netzwerkressourcen. Dies ermöglicht es ihnen, ihre eigene IT-Infrastruktur aufzubauen und zu verwalten, ohne physische Hardware besitzen zu müssen.

- Angelehnt an The NIST Definition of CloudComputing[25]
- Beispiele sind Microsoft Azure oder StackIt



Infrastructure as a Service (IaaS)

Vorteile

- Skalierbarkeit
- Kosteneffizienz
- Flexibilität
- Wartungsfrei
- Disaster Recovery

2024-11-20

IT-Transformation

└─ Architektur

└─ Infrastructure as a Service (IaaS)

Infrastructure as a Service (IaaS)

Vorteile

- Skalierbarkeit
- Kosteneffizienz
- Flexibilität
- Wartungsfrei
- Disaster Recovery

- Skalierbarkeit: Infrastrukturbedarf je nach Bedarf nach oben oder unten zu skalieren, ohne in physische Hardware zu investieren
- Kosteneffizienz: Kann die Kosten senken, da nur die genutzten Ressourcen berechnet werden. Ist pro Ressource aber teurer als gekaufte Hardware
- Flexibilität: Bietet die gegenüber PaaS Flexibilität, die verwendeten Betriebssysteme, Entwicklungsumgebungen und Anwendungen zu wählen
- Wartungsfrei: Die physische Hardware muss nicht gewartet werden. Zeitersparnis und bessere Uptime
- Disaster Recovery: Meistens Backup- und Disaster-Recovery-Lösungen verfügbar, um Ausfälle zu kompensieren



Infrastructure as a Service (IaaS)

Nachteile

- Sicherheit geteilter Ressourcen in Eigenverantwortung
- Abhängigkeit vom Internet
- Komplexität
- Kosten

2024-11-20

IT-Transformation

└─ Architektur

└─ Infrastructure as a Service (IaaS)

Infrastructure as a Service (IaaS)
Nachteile

- Sicherheit geteilter Ressourcen in Eigenverantwortung
- Abhängigkeit vom Internet
- Komplexität
- Kosten

- Sicherheit: IT- und Datensicherheit in geteilter Umgebung muss vom Unternehmen sichergestellt werden
- Abhängigkeit vom Internet: Verbindung zu der Cloudumgebung notwendig. Je nach Art der Übertragung (Backups, Offsite-Replikation) hohe Bandbreiten nötig
- Komplexität: Das korrekte Aufsetzen und Betreiben einer solchen Infrastruktur erfordert erfahrenes Personal
- Kosten: Versteckte Kosten, falls Dienste automatisch skalieren oder falsch konfiguriert sind. Langfristige Kosten



On Premise

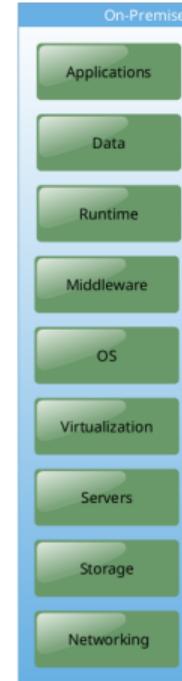
Definition

On Premise

Bei On Premise installiert, wartet und überwacht ein Unternehmen seine Hard- und Software selbst. Dabei behält das Unternehmen volle Kontrolle über die Hard- und Software und kann Modifikationen vornehmen. Der Aufwand ist groß.

Eigene
Verantwortung

Verantwortung
Service Provider



2024-11-20

IT-Transformation
└─ Architektur

└─ On Premise

On Premise
Definition

On Premise
Bei On Premise installiert, wartet und überwacht ein Unternehmen seine Hard- und Software selbst. Dabei behält das Unternehmen volle Kontrolle über die Hard- und Software und kann Modifikationen vornehmen. Der Aufwand ist groß.



- Beispiele sind ein eigenes Rechenzentrum oder ein Bereich bei einem Colocation Anbieter



On Premise

Vorteile

- Volle Kontrolle über gesamte Hard- und Software
- Geringe langfristige Kosten durch Einmalanschaffung
- Überprüfbarkeit der Einhaltung von Vorschriften und Sicherheitsnormen

2024-11-20

IT-Transformation
└─ Architektur

└─ On Premise

On Premise
Vorteile

- Volle Kontrolle über gesamte Hard- und Software
- Geringe langfristige Kosten durch Einmalanschaffung
- Überprüfbarkeit der Einhaltung von Vorschriften und Sicherheitsnormen



On Premise

Nachteile

- Hohe initiale Kosten
- Komplexität
- Bedarf an Fachpersonal für alle Schichten wie Hardware, Netzwerk, OS, Virtualisierung, Plattform, Security etc.
- Sicherheit in Eigenverantwortung

2024-11-20

IT-Transformation

└─ Architektur

└─ On Premise

On Premise
Nachteile

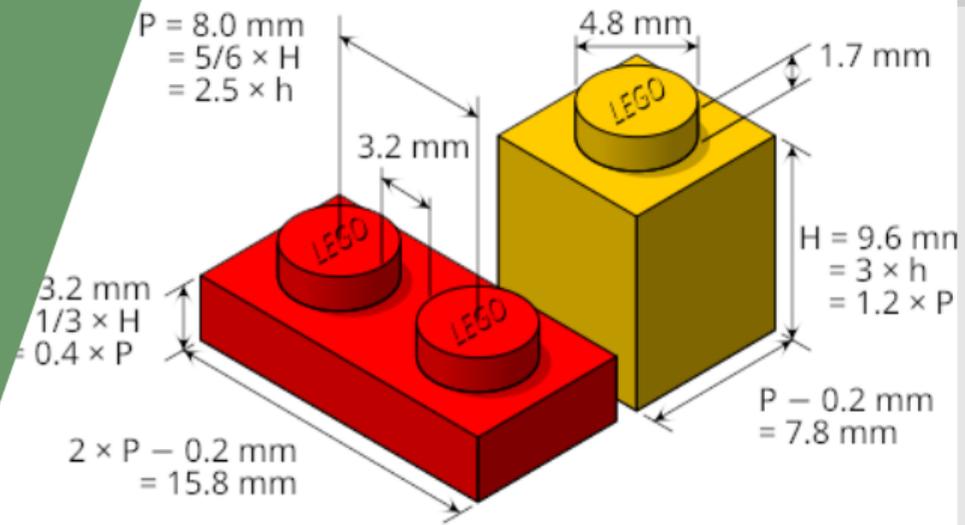
- Hohe initiale Kosten
- Komplexität
- Bedarf an Fachpersonal für alle Schichten wie Hardware, Netzwerk, OS, Virtualisierung, Plattform, Security etc.
- Sicherheit in Eigenverantwortung

- Komplexität durch zahlreiche selbst zu verwaltende Schichten
- Gerade Hochverfügbare Systeme sind ohne Bereitschaftsdienste und Automatisierung nicht umsetzbar
- Backup, Security und Disaster-Recovery dürfen nicht zu kurz kommen



Microservices oder Monolith

Wie kann ich meine Applikationen strukturieren?



2024-11-20

IT-Transformation

└─ Architektur

└─ Microservices oder Monolith

- Bild: Lego Dimensions unter CC BY-SA 3.0[26]



Microservices vs Monolith

Unterscheidung

Monolith

Eine abgeschlossener und für sich stehende Software.[27]

Microservices

Erledige nur eine Aufgabe und erledige sie gut.[28]

2024-11-20

IT-Transformation

└─ Architektur

└─ Microservices vs Monolith

Microservices vs Monolith
Unterscheidung

Monolith

Eine abgeschlossener und für sich stehende Software [27]

Microservices

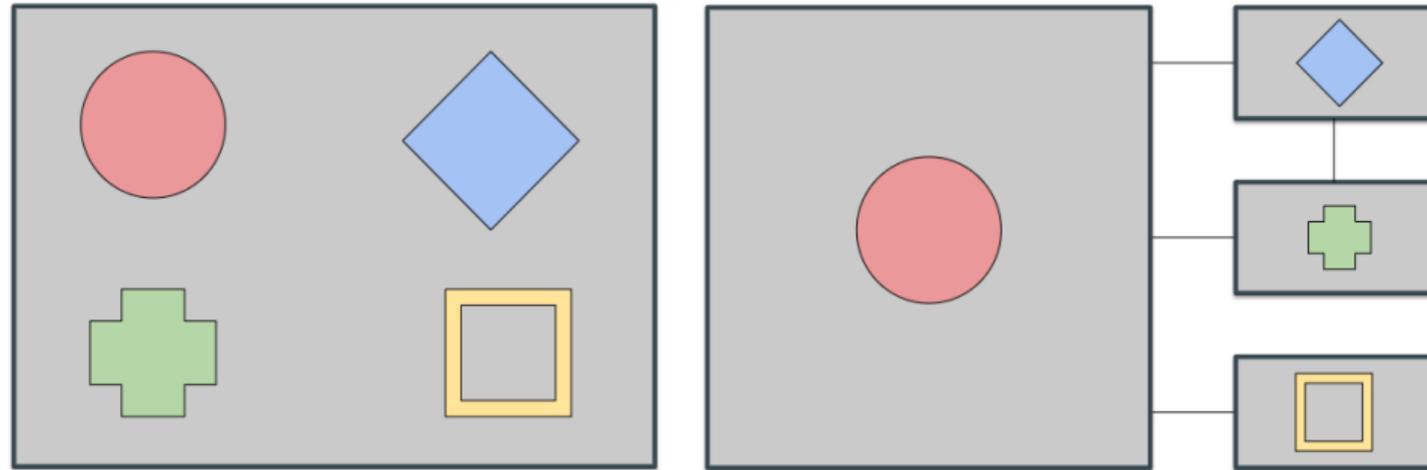
Erledige nur eine Aufgabe und erledige sie gut [28]

- Monolithische Programme fanden ihren Ursprung auf riesigen Mainframe Systemen, wo sie ohne Modularisierung entwickelt wurden
- Microservices stehen für einzelne Services, die jeder genau eine Funktion erfüllen und im Zusammenspiel die gesamte Anforderung abdecken



Microservices vs Monolith

Grafische Unterscheidung



monolith

microservices

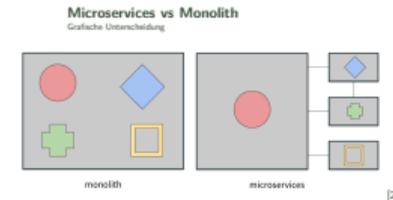
[29]

2024-11-20

IT-Transformation

└─ Architektur

└─ Microservices vs Monolith

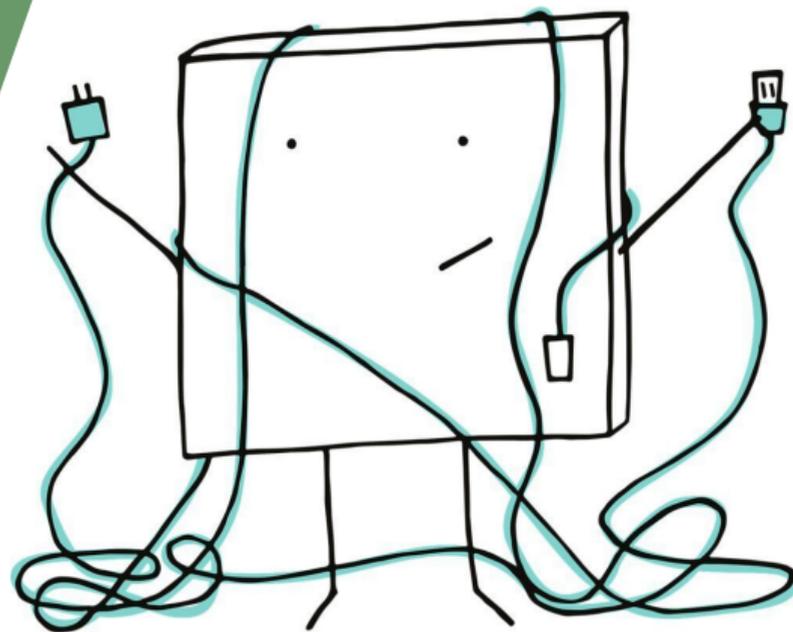


- Darstellung von Monolithischer und Microservice Architektur von GitLab im Blogbeitrag [What are microservices?\[29\]](#)
- Z.B.: Ein Newslettersystem. Personen können Artikel schreiben und darauf basierend Newsletter verschicken.
- Ein Monolithischer Service beinhaltet ein Usermodul, ein Newslettermodul, ein Artikelmodul und ein Frontendmodul.
- In einer Microservicearchitektur agiert die Nutzer*in mit dem Frontendmicroservice. Dieser kommuniziert mit dem User Microservice, um die Daten des Users zu bekommen. Um Artikel zu bearbeiten und Newsletter zu verschicken kommuniziert er mit den beiden entsprechenden Microservices



SOLID-Prinzipien

Welche Prinzipien unterstützen eine gute Architektur?



2024-11-20

IT-Transformation

└─ Architektur

└─ SOLID-Prinzipien

- Beispiel für Prinzipien in der Softwarearchitektur bei Objektorientierter Entwicklung
- Die SOLID-Prinzipien wurden von Robert C. Martin zusammengestellt
- Sie sollen zu höherer Wartbarkeit und somit längerer Lebensdauer von Software führen



SOLID-Prinzipien

Zusammenstellung nach Robert C. Martin

Definition Single-Responsibility-Prinzip

Es sollte nie mehr als einen Grund dafür geben, eine Klasse zu ändern.[30]

Definition Open-Closed-Prinzip

Module sollten sowohl offen (für Erweiterungen) als auch verschlossen (für Modifikationen) sein.[31]

2024-11-20

IT-Transformation

└─ Architektur

└─ SOLID-Prinzipien

SOLID-Prinzipien
Zusammenstellung nach Robert C. Martin

Definition Single-Responsibility-Prinzip

Es sollte nie mehr als einen Grund dafür geben, eine Klasse zu ändern.[30]

Definition Open-Closed-Prinzip

Module sollten sowohl offen (für Erweiterungen) als auch verschlossen (für Modifikationen) sein.[31]

- Single-Responsibility-Prinzip nach Robert C. Martin: Trennung von Verantwortlichkeiten für Kapselung und Kohäsion
- Jede Komponente der Architektur sollte nur eine Aufgabe erledigen und alle Funktionen dieser Aufgabe abdecken
- Open-Closed-Prinzip nach Bertrand Meyer: Vererbung ist ein Beispiel. Sie verändert eine Klasse nicht, sondern erweitert sie



SOLID-Prinzipien

Zusammenstellung nach Robert C. Martin

Definition Liskovsches Substitutionsprinzip

Sei $q(x)$ eine beweisbare Eigenschaft von Objekten x des Typs T . Dann soll $q(y)$ für Objekte y des Typs S wahr sein, wobei S ein Untertyp von T ist.[32]

Definition Interface-Segregation-Prinzip

Clients sollten nicht dazu gezwungen werden, von Interfaces abzuhängen, die sie nicht verwenden.[33]

Definition Liskovsches Substitutionsprinzip

Sei $q(x)$ eine beweisbare Eigenschaft von Objekten x des Typs T . Dann soll $q(y)$ für Objekte y des Typs S wahr sein, wobei S ein Untertyp von T ist.[32]

Definition Interface-Segregation-Prinzip

Clients sollten nicht dazu gezwungen werden, von Interfaces abzuhängen, die sie nicht verwenden.[33]

- Liskovsches Substitutionsprinzip nach Barbara Liskov und Jeannette Wing: Eine abgeleitete Klasse muss in jedem Kontext für ihre Basisklasse austauschbar sein. Sie sollte das Verhalten der Basisklasse nur erweitern, ohne es einzuschränken.
- Wenn ich eine Klasse Vogel habe und diese die Methode fliegen, darf ich nicht in einer vererbten Klasse Pinguin das fliegen streichen
- Das würde zu Verwirrungen bei der Nutzung der Klassenhierarchie und zu unerwartetem Laufzeitverhalten führen
- Interface-Segregation-Prinzip nach Robert C. Martin: Der Umfang einer Schnittstelle wird durch die Anforderungen des Clients bestimmt, nicht umgekehrt
- Ich habe einen Mailserver der senden und empfangen kann und einen Client, der Mails verschicken soll.
- nach dem Interface-Segregation-Prinzip reicht es aus nur das Senden zu implementieren. Es wäre völlig überflüssig, wenn ich auch den Empfang implementieren müsste, obwohl ich nie etwas empfangen will



SOLID-Prinzipien

Zusammenstellung nach Robert C. Martin

Definition Dependency-Inversion-Prinzip

Module hoher Ebenen sollten nicht von Modulen niedriger Ebenen abhängen. Beide sollten von Abstraktionen abhängen. Abstraktionen sollten nicht von Details abhängen. Details sollten von Abstraktionen abhängen.[34]

2024-11-20

IT-Transformation

└─ Architektur

└─ SOLID-Prinzipien

SOLID-Prinzipien
Zusammenstellung nach Robert C. Martin

Definition Dependency-Inversion-Prinzip

Module hoher Ebenen sollten nicht von Modulen niedriger Ebenen abhängen. Beide sollten von Abstraktionen abhängen. Abstraktionen sollten nicht von Details abhängen. Details sollten von Abstraktionen abhängen [34]

- Dependency-Inversion-Prinzip nach Robert C. Martin: In einer geschichteten Architektur existieren Klassen und Module auf unterschiedlichen Abstraktionsebenen
- Klassen auf niedrigerer Ebene kapseln Implementierungsdetails wie z.B. Datenbankverbindungen
- Klassen auf höhere Ebenen kapseln abstrakte Geschäftslogik
- Höhere Schichten sollen dabei abstrakte Interfaces verwenden, statt auf die Klassen und somit Implementierungsdetails zuzugreifen



Schnittstellen

*Welche Schnittstellen ver-
netzen Komponenten und
Systeme?*

2024-11-20

IT-Transformation

└ Schnittstellen

└ Schnittstellen

- Schnittstellen sind wichtig, um eine performante, stabile und sichere Zusammenarbeit von verschiedenen Systemen zu erzielen



Schnittstelle

Definition

Definition Schnittstelle

Eine Schnittstelle (englisch Interface [...]) ist ein Teil eines Systems, das der Kommunikation dient.[35]

2024-11-20

IT-Transformation
└ Schnittstellen

└ Schnittstelle

Schnittstelle
Definition

Definition Schnittstelle

Eine Schnittstelle (englisch Interface [...]) ist ein Teil eines Systems, das der Kommunikation dient.[35]

- Der Begriff stammt aus der Naturwissenschaft und bezeichnet die physikalische Phasengrenze zweier Zustände eines Mediums
- Er beschreibt bildhaft die Eigenschaft eines Systems als Black Box, von der nur die „Oberfläche“ sichtbar ist; nur über diese ist eine Kommunikation möglich.
- Zwei benachbarte Black Boxes können nur miteinander kommunizieren, wenn ihre Oberflächen „zusammenpassen“.



Fragwürdige Schnittstellen

Schnittstellen, deren Nutzung kritisch hinterfragt werden sollte

- Fax
- SMS

2024-11-20

IT-Transformation

└ Schnittstellen

└ Fragwürdige Schnittstellen

Fragwürdige Schnittstellen
Schnittstellen, deren Nutzung kritisch hinterfragt werden sollte

- Fax
- SMS

- Fax ist der Treppenwitz der deutschen Digitalisierung. Wird immer noch eingesetzt, ist aber unsicher, verschlechtert durch Ausdruck, Scan, Ausdruck, Scan die Qualität erheblich und ist fehleranfällig
- SMS sind nur schwach auf Netzebene verschlüsselt. Passende SIM-Karte kann durch Social Engineering erlangt werden (SIM-Swapping). Gehen nach gewisser Zeit ohne Kontakt zum Empfangsgerät verloren



Web Schnittstellen

Schnittstellen für Kommunikation zwischen Systemen

- RESTful APIs
- GraphQL
- WebSocket
- Remote Procedure Calls
- SOAP (Simple Object Access Protocol)

2024-11-20

IT-Transformation
└─ Schnittstellen

└─ Web Schnittstellen

Web Schnittstellen
Schnittstellen für Kommunikation zwischen Systemen

- RESTful APIs
- GraphQL
- WebSocket
- Remote Procedure Calls
- SOAP (Simple Object Access Protocol)

- RESTful APIs sind Schnittstellen, die auf die Standardmethoden des HTTP Protocols setzen (GET, POST, PUT). Dabei werden Ressourcen über URLs angesprochen
- GraphQL ist eine moderne Datenabfrage- und Manipulationssprache. Mit ihr kann man angeben, welcher Teil von den Daten relevant ist und so das Datenvolumen reduzieren. 2012 von Facebook entwickelt und 2018 an eine Foundation übergeben
- WebSocket ist ein auf TCP basierendes Netzwerkprotokoll. Mit ihm ist im Gegensatz zu HTTP eine bidirektionale Kommunikation möglich
- Remote Procedure Calls stammen noch aus dem RFC 707 von 1976 und ermöglichen es Funktionen auf anderen Systemen aufzurufen
- SOAP (Simple Object Access Protocol) ist ein für die Informatik alter Standard, der bis auf 1999 zurückgeht. Er basiert auf den Austausch von XML, was im Hinblick auf Schnelligkeit und Sicherheit einige Probleme mitsich bringt. Wird bei neueren Projekten selten eingesetzt.



Cybersicherheit

Welche Rolle spielt Cybersicherheit in der digitalen Transformation?



2024-11-20

IT-Transformation

└─ Sicherheit

└─ Cybersicherheit

- Photo: Red Text Cyber Security between Blue Binary Code Data on Screen[36] von Marco Verch unter Creative Commons 2.0[1]

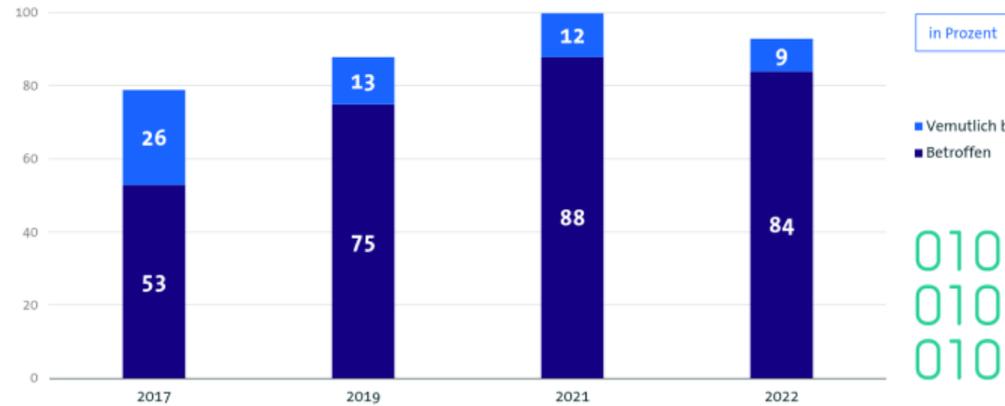


Cyberangriffe

Betroffene Unternehmen

Deutsche Wirtschaft in der Breite von Angriffen betroffen

War Ihr Unternehmen innerhalb der letzten 12 Monate (2017 und 2019: innerhalb der letzten zwei Jahre) von Diebstahl, Industriespionage oder Sabotage betroffen?



Basis: Alle befragten Unternehmen (n=1.066) | Quelle: Bitkom Research 2022



bitkom [20]

2024-11-20

IT-Transformation

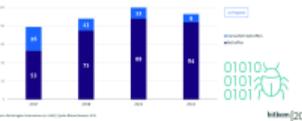
Sicherheit

Cyberangriffe

Cyberangriffe

Deutsche Wirtschaft in der Breite von Angriffen betroffen

War Ihr Unternehmen innerhalb der letzten 12 Monate (2017 und 2019: innerhalb der letzten zwei Jahre) von Diebstahl, Industriespionage oder Sabotage betroffen?



- Grafik aus der Präsentation Wirtschaftsschutz 2022 der bitkom

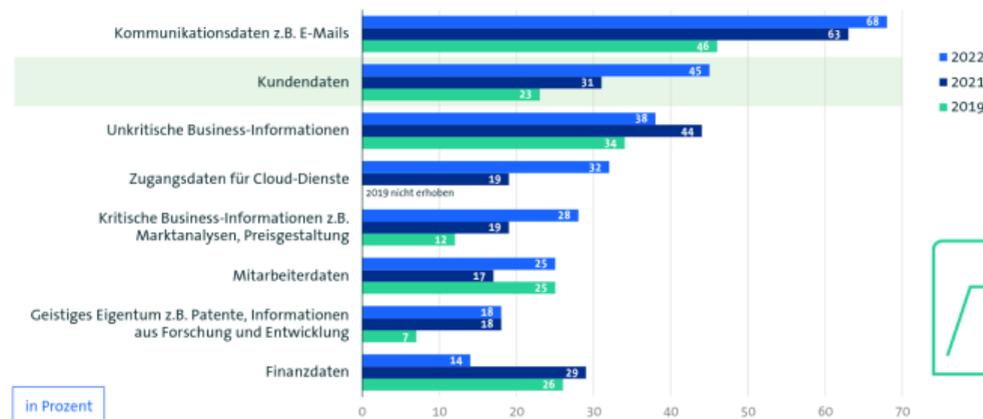


Cyberangriffe

Diebesgut

Datendiebstahl: Immer öfter sind Dritte betroffen

Welche der folgenden Arten von digitalen Daten wurden in Ihrem Unternehmen gestohlen?



in Prozent



bitkom [20]

Basis: Alle befragten Unternehmen, die in den letzten 12 Monaten (2019: 2 Jahren) von Diebstahl von sensiblen digitalen Daten betroffen waren (2022: n=383; 2021: n=330; 2019: n=229) | Mehrfachnennungen möglich | Quelle: Bitkom Research 2022

2024-11-20
IT-Transformation
└ Sicherheit
└ Cyberangriffe



- Grafik aus der Präsentation Wirtschaftsschutz 2022 der bitkom

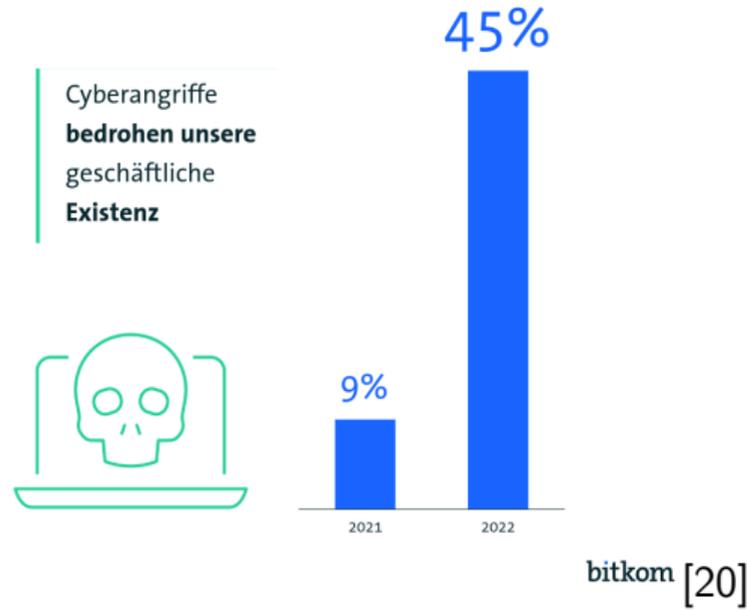


Cyberangriffe

Existenzbedrohend

Cyberattacken bedrohen Existenz vieler Unternehmen

Inwiefern stimmen Sie der Aussage zu bzw. nicht zu?



Basis: Alle befragten Unternehmen (n=1.066) | Quelle: Bitkom Research 2022

2024-11-20

IT-Transformation

└ Sicherheit

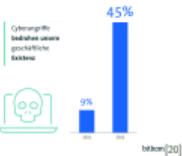
└ Cyberangriffe

Cyberangriffe

Existenzbedrohend

Cyberattacken bedrohen Existenz vieler Unternehmen

Inwiefern stimmen Sie der Aussage zu bzw. nicht zu?



- Grafik aus der Präsentation Wirtschaftsschutz 2022 der bitkom



Cyberangriffe

Schäden

202 Milliarden Euro Schaden pro Jahr

Wodurch sind Ihrem Unternehmen innerhalb der letzten 12 Monate Schäden im Zusammenhang mit Diebstahl, Industriespionage oder Sabotage entstanden?

Schaden durch...	Schadenssummen in Mrd. Euro (2022)	Schadenssummen in Mrd. Euro (2021)	Schadenssummen in Mrd. Euro (2019)	Schadenssummen in Mrd. Euro (2017)
Ausfall, Diebstahl oder Schädigung von Informations- und Produktionssystemen oder Betriebsabläufen	41,5	61,9	13,5	5,3
Erpressung mit gestohlenen Daten oder verschlüsselten Daten	10,7	24,3	5,3	0,7
Datenschutzrechtliche Maßnahmen (z.B. Information von Kunden)	18,3	17,1	4,4	3,2
Patentrechtsverletzungen (auch schon vor der Anmeldung)	18,8	30,5	14,3	7,7
Umsatzeinbußen durch Verlust von Wettbewerbsvorteilen	41,5	29	11,1	8,6
Umsatzeinbußen durch nachgemachte Produkte (Plagiate)	21,1	22,7	11,1	3,5
Imageschaden bei Kunden oder Lieferanten/ Negative Medienberichterstattung	23,6	12,3	9,3	7,7
Kosten für Ermittlungen und Ersatzmaßnahmen	10,1	13,3	18,3	10,6
Kosten für Rechtsstreitigkeiten	16,2	12,4	15,6	5,5
Höhere Mitarbeiterfluktuation/Abwerben von Mitarbeitern	-	-	-	2,2
Sonstige Schäden	0,9	0	<0,1	<0,1
Gesamtschaden pro Jahr	202,7	223,5	102,9	54,8

Basis: Alle befragten Unternehmen, die in den letzten 12 Monaten (2019 und 2017: 2 Jahren) von Diebstahl von Datendiebstahl, Industriespionage oder Sabotage betroffen waren (2022: n=899; 2021: n=935; 2019: n=801; 2017: n=571) | Mehrfachnennungen möglich | Quelle: Bitkom Research 2022



2024-11-20

IT-Transformation
↳ Sicherheit

↳ Cyberangriffe

Cyberangriffe

Schäden

202 Milliarden Euro Schaden pro Jahr



- Grafik aus der Präsentation Wirtschaftsschutz 2022 der bitkom
- Laut Studie auf Statista, die ich nicht einsehen kann, 18.700€ pro einzelnen Fall
- Aber was ist nun eigentlich Cybersicherheit



CIA Triade

Algemeine Schutzziele

- Vertraulichkeit (confidentiality)
- Integrität (integrity)
- Verfügbarkeit (availability)

2024-11-20

IT-Transformation

└─ Sicherheit

└─ CIA Triade

CIA Triade
Algemeine Schutzziele

- Vertraulichkeit (confidentiality)
- Integrität (integrity)
- Verfügbarkeit (availability)

- Siehe Wikipedia Artikel Informationssicherheit[37]
- Vertraulichkeit: Daten dürfen lediglich von autorisierten Benutzern gelesen werden, dies gilt sowohl beim Zugriff auf gespeicherte Daten wie auch während der Datenübertragung
- Integrität: Daten dürfen nicht unbemerkt verändert werden. Alle Änderungen müssen nachvollziehbar sein
- Verfügbarkeit: Verhinderung von Systemausfällen; der Zugriff auf Daten muss innerhalb eines vereinbarten Zeitrahmens gewährleistet sein
- Weitere Schutzziele: Nichtabstreitbarkeit
- Risikomanagement ist enorm wichtig. Nicht alles kann man verhindern



Risikobewältigung

Vier Strategien

- Risikovermeidung
- Risikoverminderung
- Risikoverlagerung
- Risikoakzeptanz

2024-11-20

IT-Transformation

└ Sicherheit

└ Risikobewältigung

Risikobewältigung

Vier Strategien

- Risikovermeidung
- Risikoverminderung
- Risikoverlagerung
- Risikoakzeptanz

- Wenn ich ein Risiko vermeiden kann, tritt es nicht auf. Z.B.: Wenn ich nicht fliegen kann, ich nicht mit dem Flugzeug abstürzen
- Risikoverminderung reduziert die Wahrscheinlichkeit des Eintritts oder die Folgen des Eintritts
- Risikoverlagerung wäre z.B. eine Versicherung. Kann in Kombination mit Risikoverminderung auftreten
- Ich kann auch das Risiko bewusst in Kauf nehmen und mit den Konsequenzen leben. Kann in Kombination mit Risikoverminderung oder -verlagerung auftreten



Datenschutz

Warum ist es Datenschutz und Data Governance so wichtig?



2024-11-20

IT-Transformation

└─ Datenschutz

└─ Datenschutz

- Datenschutz kann als Wettbewerbsvorteil angesehen werden
- Schon aus moralischen und rechtlichen Gründen sollte man sich an die Datenschutzgesetze halten
- Auch oder gerade Behörden, die von Strafen durch die DSGVO ausgenommen sind



Datenschutz

Definition

Definition Datenschutz

[Datenschutz ist] Schutz vor missbräuchlicher Datenverarbeitung, Schutz des Rechts auf informationelle Selbstbestimmung, Schutz des Persönlichkeitsrechts bei der Datenverarbeitung und auch Schutz der Privatsphäre[38]

2024-11-20

IT-Transformation
└─ Datenschutz

└─ Datenschutz

- Siehe Wikipedia Artikel Datenschutz[38]

Datenschutz
Definition

Definition Datenschutz

[Datenschutz ist] Schutz vor missbräuchlicher Datenverarbeitung, Schutz des Rechts auf informationelle Selbstbestimmung, Schutz des Persönlichkeitsrechts bei der Datenverarbeitung und auch Schutz der Privatsphäre[38]



Vorteile eines gelebten Datenschutzes

Mögliche Vorteile

- Startpunkt für ein Compliance-Management-System
- Voranbringen der Transformation
- IT-Sicherheit
- Vertrauen

2024-11-20

IT-Transformation
└─ Datenschutz

└─ Vorteile eines gelebten Datenschutzes

Vorteile eines gelebten Datenschutzes
Mögliche Vorteile

- Startpunkt für ein Compliance-Management-System
- Voranbringen der Transformation
- IT-Sicherheit
- Vertrauen

- Ein Compliance-Management-System muss nicht unbedingt ein Programm sein, sondern ist vielmehr ein Konzept, wo die Prozesse definiert und versioniert werden
- Gerade junge Unternehmen haben noch kein umfassendes System zur Einhaltung aller regulatorischen Anforderungen
- Datenschutz nach DSGVO fordert Implementierungen nach Stand der Technik
- Manchmal ist es einfacher, mit Sachzwängen durch Vorgaben zu argumentieren
- Einer Organisation, die transparent über Datenschutz, Prozesse und Vorfälle berichtet wird eher vertraut
- Deutsche Gesellschaft für Datenschutz (DGD) zitiert Studie: 92% der Unternehmen, die Maßnahmen zur Einhaltung der DSGVO ergriffen haben, [verzeichneten] einen erheblichen Anstieg des Kundenvertrauens und der Kundenbeteiligung[39]



Vorteile eines gelebten Datenschutzes

Mögliche Vorteile

- Geschäftsvorteil
- Risikominimierung
- Reduktion von Ressourcen
- Vereinheitlichung von Märkten und One-Stop-Shop-Verfahren

2024-11-20

IT-Transformation

└─Datenschutz

└─Vorteile eines gelebten Datenschutzes

Vorteile eines gelebten Datenschutzes

Mögliche Vorteile

- Geschäftsvorteil
- Risikominimierung
- Reduktion von Ressourcen
- Vereinheitlichung von Märkten und One-Stop-Shop-Verfahren

- Eine Organisation die transparent über Datenschutz, Prozesse und Vorfälle berichtet wird bei ähnlichen Angeboten bevorzugt als Geschäftspartner ausgewählt werden
- Nicht erhobene, korrekt anonymisierte oder gelöschte Daten können nicht gestohlen werden. Verstöße gegen die DSGVO können teuer werden
- Daten die nicht benötigt werden sollten nicht erhoben werden. Benötigte Daten sollten gelöscht werden, wenn ihr Zweck erfüllt ist. Das spart evtl. viele Ressourcen
- Vereinheitlichung des Datenschutzes Europaweit und nur noch eine Aufsichtsbehörde



Dos and Don'ts

Welche Dos and Don'ts gibt es bei der IT-Transformation?



2024-11-20

IT-Transformation
└ Dos and Don'ts

└ Dos and Don'ts

- Bild aus der Folge Deja Q aus Star Trek: The Next Generation[40]



Don'ts

Antipattern in der IT - Teil 1[41]

- Infrastrukturzoo
- Manuelle Prozesse und keine Automatisierung
- Mangelnde Ressourcen für die Transformation

2024-11-20

IT-Transformation
└─ Dos and Don'ts

└─ Don'ts

Don'ts
Antipattern in der IT - Teil 1[41]

- Infrastrukturzoo
- Manuelle Prozesse und keine Automatisierung
- Mangelnde Ressourcen für die Transformation

- Entnommen aus dem Buch IT-Operations in der Transformation
- Infrastrukturzoo: Gewachsene Infrastruktur. Unterschiedlichste Hard- und Software für die gleichen Probleme
- Manuelle Prozesse und keine Automatisierung: Änderungen werden aufwändig und riskant. Wissen in einzelnen Köpfen
- Mangelnde Ressourcen für die Transformation: Transformation kann man nicht nebenbei erledigen. Es wird Zeit, Personal und Geld für Hard- und Software benötigt



Don'ts

Antipattern in der IT - Teil 2

- Not invented here
- Kulturignoranz
- Ignorieren des Gesetzes von Larman[42]
- Technologische Schulden
- Technologische Schulden kosten Zinsen und die Zinslast erdrückt irgendwann jedes Projekt

2024-11-20

IT-Transformation

- └ Dos and Don'ts
 - └ Don'ts

Don'ts
Antipattern in der IT - Teil 2

- Not invented here
- Kulturignoranz
- Ignorieren des Gesetzes von Larman[42]
- Technologische Schulden
- Technologische Schulden kosten Zinsen und die Zinslast erdrückt irgendwann jedes Projekt

- Entnommen aus dem Buch IT-Operations in der Transformation
- Not invented here: Nur lokales Maximum an Leistung durch Versumpfen in selbst gebauten Lösungen ohne Blick auf verfügbare, erprobte Standardlösungen
- Kulturignoranz: IT-Transformation ist kein rein technisches Konstrukt, sondern erfordert ein Umdenken bei allen Beteiligten
- Gesetz von Larman: Organisationen sind implizit darauf optimiert, Veränderungen von Positionen und Macht des mittleren Managements zu verhindern => Veränderungen am Ablauf durch Einführung von DevOps muss geführt werden



Don'ts

Antipattern in der IT - Teil 3

- Software not Ready
- Planlosigkeit und Intransparenz
- Unterschätzung der Komplexität
- Appzwang

2024-11-20

IT-Transformation
└─ Dos and Don'ts

└─ Don'ts

- Teilweise entnommen aus dem Buch IT-Operations in der Transformation
- Software not Ready: Die Infrastruktur wird losgelöst von der darauf laufenden Anwendung modernisiert und diese kommt mit den Veränderungen nicht klar. Z.B.: Stateful Services in der Cloud
- Agil bedeutet eben nicht ohne Plan loszulaufen und auf Messwerte zu verzichten. Im Gegenteil. Durch Auswahl passender Key Performance Indicator (KPI) und regelmäßige Feedbackschleifen in Reviews kann eine deutlich bessere Sichtbarkeit erreicht werden
- Gerade Cloud Technologien und Automatisierung haben die technische Komplexität deutlich vergrößert und benötigen erfahrene Expert*innen

Don'ts
Antipattern in der IT - Teil 3

- Software not Ready
- Planlosigkeit und Intransparenz
- Unterschätzung der Komplexität
- Appzwang



Dos

Pattern für erfolgreiche IT-Transformation - Teil 1

- Passende Methodik wählen: Klassische Methoden für komplizierte Dinge, Agile für komplexe
- Ubiquitous Language (Domain Driven Design) für Kommunikation
- Datenschutz und IT-Sicherheit von Anfang an

2024-11-20

IT-Transformation

- └ Dos and Don'ts

- └ Dos

- Kompliziert: Wenn ich Fachwissen oder eine wie auch immer geartete Analyse brauche, um das Problem zu lösen, bin ich im Bereich der komplizierten Probleme.[43]
- Lösung: Erkennen, analysieren, handeln (sense, analyze, act)
- Komplex: Wenn die Beziehung zwischen Ursache und Wirkung erst im Nachhinein ermittelt werden kann, handelt es sich um komplexe Probleme.[43]
- Erkunden, erkennen, reagieren (probe, sense, react)
- Ubiquitous Language von Eric Evans in Domain Driven Design. Eine gemeinsame Fachsprache für alle Beteiligten inkl. Business, Dev, Ops, Nutzer*innen.[44]
- Ubiquitäre Sprache geht auch auf Deutsch[45]
- Datenschutz und Sicherheit werden umso teurer, umso später sie implementiert werden, da teils fundamentale Änderungen am System erforderlich sind

Dos

Pattern für erfolgreiche IT-Transformation - Teil 1

- Passende Methodik wählen: Klassische Methoden für komplizierte Dinge, Agile für komplexe
- Ubiquitous Language (Domain Driven Design) für Kommunikation
- Datenschutz und IT-Sicherheit von Anfang an



Dos

Pattern für erfolgreiche IT-Transformation - Teil 2

- Crossfunktionale Teams
- DevOps und DevSecOps
- Everything as Code
- Accessibility

2024-11-20

IT-Transformation

- └ Dos and Don'ts

- └ Dos

- In Crossfunktionalen Teams arbeiten Personen unterschiedlicher Fachrichtungen zusammen, um gemeinsam ein Produkt zu bearbeiten. Z.B.: Business, UX, Frontend, Backend, Ops
- Dadurch, dass alle wichtigen Personen von der Anforderung über Entwurf und Entwicklung bis zum Betrieb im Team sitzen, sind kurze Feedbacks und hohe Flexibilität möglich
- DevOps richtet sich nach diesem Prinzip. You Built It - Your Run It
- DevSecOps erweitert das Prinzip um die Rolle der Cybersecurity im Team
- Everything as Code garantiert reproduzierbare, testbare und wartbare Infrastrukturkonfiguration wie VMs oder Deployments in Kubernetes

Dos
Pattern für erfolgreiche IT-Transformation - Teil 2

- Crossfunktionale Teams
- DevOps und DevSecOps
- Everything as Code
- Accessibility



Dos

Pattern für erfolgreiche IT-Transformation - Teil 2

- OpenData
- OpenSource
- Kommunikation

2024-11-20

IT-Transformation
└ Dos and Don'ts

└ Dos

- Falls nicht-personenbezogene Daten veröffentlicht werden können, tut das über eine Schnittstelle im maschinenlesbaren Format wie XML oder JSON
- Die Kreativität von Menschen wird weitere Lösungen basierend auf den Daten schaffen. Z.B.: Fahrplanauskünfte oder Spielplatzkarten
- Nutzt OpenSource Produkte um einem Vendorlockin zu entgehen
- Tragt zu OpenSource Projekten bei, wenn ihr Änderungen braucht oder intern vornehmt
- Public Money - Public Code

Dos
Pattern für erfolgreiche IT-Transformation - Teil 2

- OpenData
- OpenSource
- Kommunikation



Vielen Dank!

Materialien verfügbar unter
<https://seism0saurus.de/posts/it-transformation/>

2024-11-20

IT-Transformation
└ Abschluss

Vielen Dank!

Materialien verfügbar unter
<https://seism0saurus.de/posts/it-transformation/>



References I

- [1] *CC BY 2.0*. URL: <https://creativecommons.org/licenses/by/2.0/> (visited on 09/30/2024).
- [2] Thomas Wiesböck Florian und Hess. "Digital innovations". In: *Electronic Markets* 30.1 (2020), pp. 75–86. ISSN: 1422-8890. DOI: 10.1007/s12525-019-00364-9. URL: <https://doi.org/10.1007/s12525-019-00364-9>.
- [3] Prof. Dr. Thomas Hess. *Digitalisierung*. [Online; Stand 1. November 2024]. 2019. URL: <https://wi-lex.de/index.php/lexikon/technologische-und-methodische-grundlagen/informatik-grundlagen/digitalisierung/>.

2024-11-20

IT-Transformation

└ Abschluss

└ References

References I

- [1] *CC BY 2.0*. URL: <https://creativecommons.org/licenses/by/2.0/> (visited on 09/30/2024).
- [2] Thomas Wiesböck Florian und Hess. "Digital innovations". In: *Electronic Markets* 30.1 (2020), pp. 75–86. ISSN: 1422-8890. DOI: 10.1007/s12525-019-00364-9. URL: <https://doi.org/10.1007/s12525-019-00364-9>.
- [3] Prof. Dr. Thomas Hess. *Digitalisierung*. [Online; Stand 1. November 2024]. 2019. URL: <https://wi-lex.de/index.php/lexikon/technologische-und-methodische-grundlagen/informatik-grundlagen/digitalisierung/>.



References II

- [4] John DeMattia Adam und McKnight. “How IT Transformation Maturity Drives IT Agility, Innovation, and Improved Business Outcomes”. In: (2017). URL: <https://www.digitaleschweiz.ch/wp-content/uploads/2017/04/esg-dellemc-it-transformation-maturity-report.pdf>.
- [5] *IT-Transformation*. URL: <https://it-service.network/blog/2019/07/23/it-transformation/> (visited on 11/14/2024).

2024-11-20

IT-Transformation
└ Abschluss

└ References

References II

- [4] John DeMattia Adam und McKnight. “How IT Transformation Maturity Drives IT Agility, Innovation, and Improved Business Outcomes”. In: (2017). URL: <https://www.digitaleschweiz.ch/wp-content/uploads/2017/04/esg-dellemc-it-transformation-maturity-report.pdf>.
- [5] *IT-Transformation*. URL: <https://it-service.network/blog/2019/07/23/it-transformation/> (visited on 11/14/2024).



References III

- [6] Bardo Jung Jürgen und Fraunholz. “Applying Frameworks”. In: *Masterclass Enterprise Architecture Management*. Cham: Springer International Publishing, 2021, pp. 181–206. ISBN: 978-3-030-78495-9. DOI: 10.1007/978-3-030-78495-9_6. URL: https://doi.org/10.1007/978-3-030-78495-9_6.
- [7] *Rolf Knoll Kommentar zu Genealogie EAM*. URL: https://www.linkedin.com/feed/update/urn:li:activity:7128821392514461696?commentUrn=urn%3Ali%3Acomment%3A%28activity%3A7128821392514461696%2C7134060087085543425%29&dashCommentUrn=urn%3Ali%3Afsd_comment%3A%287134060087085543425%2Curn%3Ali%3Aactivity%3A7128821392514461696%29 (visited on 11/19/2024).

2024-11-20

IT-Transformation

└ Abschluss

└ References

References III

- [6] Bardo Jung Jürgen und Fraunholz. “Applying Frameworks”. In: *Masterclass Enterprise Architecture Management*. Cham: Springer International Publishing, 2021, pp. 181–206. ISBN: 978-3-030-78495-9. DOI: 10.1007/978-3-030-78495-9_6. URL: https://doi.org/10.1007/978-3-030-78495-9_6.
- [7] *Rolf Knoll Kommentar zu Genealogie EAM*. URL: https://www.linkedin.com/feed/update/urn:li:activity:7128821392514461696?commentUrn=urn%3Ali%3Acomment%3A%28activity%3A7128821392514461696%2C7134060087085543425%29&dashCommentUrn=urn%3Ali%3Afsd_comment%3A%287134060087085543425%2Curn%3Ali%3Aactivity%3A7128821392514461696%29 (visited on 11/19/2024).



References IV

- [8] Marten Schönherr. *Erarbeitung von Blueprints mit Architekturframeworks*. URL: https://www.researchgate.net/publication/266212095_Erarbeitung_von_Blueprints_mit_Architekturframeworks.
- [9] *ITIL*. URL: <https://de.wikipedia.org/wiki/ITIL> (visited on 11/19/2024).
- [10] *ISO/IEC 20000*. URL: https://de.wikipedia.org/wiki/ISO/IEC_20000 (visited on 11/19/2024).
- [11] *COBIT*. URL: <https://de.wikipedia.org/wiki/COBIT> (visited on 11/19/2024).

2024-11-20

IT-Transformation
└ Abschluss

└ References

References IV

- [8] Marten Schönherr. *Erarbeitung von Blueprints mit Architekturframeworks*. URL: https://www.researchgate.net/publication/266212095_Erarbeitung_von_Blueprints_mit_Architekturframeworks.
- [9] *ITIL*. URL: <https://de.wikipedia.org/wiki/ITIL> (visited on 11/19/2024).
- [10] *ISO/IEC 20000*. URL: https://de.wikipedia.org/wiki/ISO/IEC_20000 (visited on 11/19/2024).
- [11] *COBIT*. URL: <https://de.wikipedia.org/wiki/COBIT> (visited on 11/19/2024).



References V

- [12] *Qualitätsmanagementnorm*. URL: <https://de.wikipedia.org/wiki/Qualit%C3%A4tsmanagementnorm> (visited on 11/19/2024).
- [13] *Big Ball of Mud*. URL: https://de.wikipedia.org/wiki/Big_Ball_of_Mud (visited on 11/08/2024).
- [14] *§ 2 - KONSENS-Gesetz (KONSENS-G)*. URL: https://www.buzer.de/2_KONSENS-Gesetz_KONSENS-G.htm (visited on 11/08/2024).
- [15] *Architektur (Informatik)*. URL: [https://de.wikipedia.org/wiki/Architektur_\(Informatik\)](https://de.wikipedia.org/wiki/Architektur_(Informatik)) (visited on 11/08/2024).

2024-11-20

IT-Transformation

└ Abschluss

└ References

References V

- [12] *Qualitätsmanagementnorm*. URL: <https://de.wikipedia.org/wiki/Qualit%C3%A4tsmanagementnorm> (visited on 11/19/2024).
- [13] *Big Ball of Mud*. URL: https://de.wikipedia.org/wiki/Big_Ball_of_Mud (visited on 11/08/2024).
- [14] *§ 2 - KONSENS-Gesetz (KONSENS-G)*. URL: https://www.buzer.de/2_KONSENS-Gesetz_KONSENS-G.htm (visited on 11/08/2024).
- [15] *Architektur (Informatik)*. URL: [https://de.wikipedia.org/wiki/Architektur_\(Informatik\)](https://de.wikipedia.org/wiki/Architektur_(Informatik)) (visited on 11/08/2024).



References VI

- [16] *IT-Architektur*. URL: <https://www.natuvion.com/de/it-architektur/> (visited on 11/08/2024).
- [17] *Systemarchitektur: Das Fundament einer erfolgreichen Software-Konzeption*. URL: <https://www.linkitsystems.de/article/systemarchitektur-erstellen/> (visited on 11/08/2024).
- [18] *Online Transaction Processing*. URL: https://de.wikipedia.org/wiki/Online_Transaction_Processing (visited on 11/19/2024).
- [19] *Online Analytical Processing*. URL: https://de.wikipedia.org/wiki/Online_Analytical_Processing (visited on 11/19/2024).



References VII

- [20] Achim Berg. *Wirtschaftsschutz 2022*. URL: https://www.bitkom.org/sites/main/files/2022-08/Bitkom-Charts_Wirtschaftsschutz_Cybercrime_31.08.2022.pdf.
- [21] *Architecture overview*. URL: <https://help.owasp-juice.shop/introduction/architecture.html> (visited on 11/19/2024).
- [22] *Softwarearchitektur*. URL: <https://de.wikipedia.org/wiki/Softwarearchitektur> (visited on 11/08/2024).
- [23] *Cloud Computing*. URL: https://de.wikipedia.org/wiki/Cloud_Computing (visited on 11/11/2024).

2024-11-20

IT-Transformation

└ Abschluss

└ References

References VII

- [20] Achim Berg. *Wirtschaftsschutz 2022*. URL: https://www.bitkom.org/sites/main/files/2022-08/Bitkom-Charts_Wirtschaftsschutz_Cybercrime_31.08.2022.pdf.
- [21] *Architecture overview*. URL: <https://help.owasp-juice.shop/introduction/architecture.html> (visited on 11/19/2024).
- [22] *Softwarearchitektur*. URL: <https://de.wikipedia.org/wiki/Softwarearchitektur> (visited on 11/08/2024).
- [23] *Cloud Computing*. URL: https://de.wikipedia.org/wiki/Cloud_Computing (visited on 11/11/2024).



References VIII

- [24] *On-Premises*. URL: <https://de.wikipedia.org/wiki/On-Premises> (visited on 11/11/2024).
- [25] *The NIST Definition of Cloud Computing*. URL: <http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf> (visited on 11/11/2024).
- [26] *CC BY-SA 3.0*. URL: <https://creativecommons.org/licenses/by-sa/3.0/> (visited on 09/30/2024).
- [27] *Monolithic application*. URL: https://en.wikipedia.org/wiki/Monolithic_application (visited on 11/13/2024).

2024-11-20

IT-Transformation

└ Abschluss

└ References

References VIII

- [24] *On-Premises*. URL: <https://de.wikipedia.org/wiki/On-Premises> (visited on 11/11/2024).
- [25] *The NIST Definition of Cloud Computing*. URL: <http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf> (visited on 11/11/2024).
- [26] *CC BY-SA 3.0*. URL: <https://creativecommons.org/licenses/by-sa/3.0/> (visited on 09/30/2024).
- [27] *Monolithic application*. URL: https://en.wikipedia.org/wiki/Monolithic_application (visited on 11/13/2024).



References IX

- [28] *Microservices*. URL: <https://de.wikipedia.org/wiki/Microservices> (visited on 11/13/2024).
- [29] *What are microservices?* URL: <https://about.gitlab.com/topics/microservices/> (visited on 11/19/2024).
- [30] *Single-Responsibility-Prinzip*. URL: <https://de.wikipedia.org/wiki/Single-Responsibility-Prinzip> (visited on 11/13/2024).
- [31] *Open-Closed-Prinzip*. URL: <https://de.wikipedia.org/wiki/Open-Closed-Prinzip> (visited on 11/13/2024).

2024-11-20

IT-Transformation

└ Abschluss

└ References

References IX

- [28] *Microservices*. URL: <https://de.wikipedia.org/wiki/Microservices> (visited on 11/13/2024).
- [29] *What are microservices?* URL: <https://about.gitlab.com/topics/microservices/> (visited on 11/19/2024).
- [30] *Single-Responsibility-Prinzip*. URL: <https://de.wikipedia.org/wiki/Single-Responsibility-Prinzip> (visited on 11/13/2024).
- [31] *Open-Closed-Prinzip*. URL: <https://de.wikipedia.org/wiki/Open-Closed-Prinzip> (visited on 11/13/2024).



References X

- [32] *Liskovsches Substitutionsprinzip*. URL: https://de.wikipedia.org/wiki/Liskovsches_Substitutionsprinzip (visited on 11/13/2024).
- [33] *Interface-Segregation-Prinzip*. URL: <https://de.wikipedia.org/wiki/Interface-Segregation-Prinzip> (visited on 11/13/2024).
- [34] *Dependency-Inversion-Prinzip*. URL: <https://de.wikipedia.org/wiki/Dependency-Inversion-Prinzip> (visited on 11/13/2024).
- [35] *Schnittstelle*. URL: <https://de.wikipedia.org/wiki/Schnittstelle> (visited on 11/13/2024).



References XI

- [36] *Red Text Cyber Security between Blue Binary Code Data on Screen.* URL: <https://foto.wuestenigel.com/red-text-cyber-security-between-blue-binary-code-data-on-screen/> (visited on 11/08/2024).
- [37] *Informationssicherheit - Motivation und Ziele der Informationssicherheit.* URL: https://de.wikipedia.org/wiki/Informationssicherheit#Motivation_und_Ziele_der_Informationssicherheit (visited on 11/08/2024).
- [38] *Datenschutz.* URL: <https://de.wikipedia.org/wiki/Datenschutz> (visited on 11/19/2024).
- [39] *Datenschutz als Wettbewerbsvorteil.* URL: https://dg-datenschutz.de/datenschutz_als_wettbewerbsvorteil/ (visited on 11/19/2024).

- [36] *Red Text Cyber Security between Blue Binary Code Data on Screen.* URL: <https://foto.wuestenigel.com/red-text-cyber-security-between-blue-binary-code-data-on-screen/> (visited on 11/08/2024).
- [37] *Informationssicherheit - Motivation und Ziele der Informationssicherheit.* URL: https://de.wikipedia.org/wiki/Informationssicherheit#Motivation_und_Ziele_der_Informationssicherheit (visited on 11/08/2024).
- [38] *Datenschutz.* URL: <https://de.wikipedia.org/wiki/Datenschutz> (visited on 11/19/2024).
- [39] *Datenschutz als Wettbewerbsvorteil.* URL: https://dg-datenschutz.de/datenschutz_als_wettbewerbsvorteil/ (visited on 11/19/2024).



References XII

- [40] *Deja Q*. URL: https://en.wikipedia.org/wiki/Deja_Q (visited on 11/15/2024).
- [41] *IT-Operations in der Transformation. Zukunftsweisende IT-Betriebsmodelle zwischen „Hey Joe“ und „NoOps“*. Ed. by Hans-Peter Froeschle. Ed. by Ralf Oestereich. Ed. by Nikolaus Schmidt. DOI: <https://doi.org/10.1007/978-3-658-37324-5>.
- [42] *Larman's Laws of Organizational Behavior*. URL: https://www.craiglarman.com/wiki/index.php?title=Larman%27s_Laws_of_Organizational_Behavior (visited on 11/18/2024).

2024-11-20

IT-Transformation
└ Abschluss

└ References

References XII

- [40] *Deja Q*. URL: https://en.wikipedia.org/wiki/Deja_Q (visited on 11/15/2024).
- [41] *IT-Operations in der Transformation. Zukunftsweisende IT-Betriebsmodelle zwischen „Hey Joe“ und „NoOps“*. Ed. by Hans-Peter Froeschle. Ed. by Ralf Oestereich. Ed. by Nikolaus Schmidt. DOI: <https://doi.org/10.1007/978-3-658-37324-5>.
- [42] *Larman's Laws of Organizational Behavior*. URL: https://www.craiglarman.com/wiki/index.php?title=Larman%27s_Laws_of_Organizational_Behavior (visited on 11/18/2024).



References XIII

- [43] *Komplex und kompliziert: Wo liegt da der Unterschied?* URL: <https://chaosverbesserer.de/blog/2017/02/25/komplex-und-kompliziert-wo-liegt-da-der-unterschied/> (visited on 11/18/2024).
- [44] *Domain-driven Design - Ubiquitäre Sprache.* URL: https://de.wikipedia.org/wiki/Domain-driven_Design#Ubiquit%C3%A4re_Sprache (visited on 11/18/2024).
- [45] *Ubiquitous Language in einer nicht-englischen Fachdomäne: Unsere Herangehensweise.* URL: <https://www.webfactory.de/blog/ubiquitous-language-in-nicht-englischer-domaene> (visited on 11/18/2024).

2024-11-20

IT-Transformation

└ Abschluss

└ References

References XIII

- [43] *Komplex und kompliziert: Wo liegt da der Unterschied?* URL: <https://chaosverbesserer.de/blog/2017/02/25/komplex-und-kompliziert-wo-liegt-da-der-unterschied/> (visited on 11/18/2024).
- [44] *Domain-driven Design - Ubiquitäre Sprache.* URL: https://de.wikipedia.org/wiki/Domain-driven_Design#Ubiquit%C3%A4re_Sprache (visited on 11/18/2024).
- [45] *Ubiquitous Language in einer nicht-englischen Fachdomäne: Unsere Herangehensweise.* URL: <https://www.webfactory.de/blog/ubiquitous-language-in-nicht-englischer-domaene> (visited on 11/18/2024).