

Prozessoptimierung und Architekturentwicklung am Beispiel "Fremdvertragserfassung" im Umfeld der SaaS CRM Plattform eines Versicherungsun- ternehmens

Projektarbeit II

für die Prüfung zum
Bachelor of Science

des Studiengangs Wirtschaftsinformatik – Application Management
an der
Dualen Hochschule Baden-Württemberg Lörrach

Jacob Ruhnau

30. September 2024

Kurs
Ausbildungsfirma
Betreuer der Ausbildungsfirma
Wissenschaftlicher Betreuer

WWI22-AM
Helvetia Versicherungen, Basel CH
Kevin Gadiant
Tim Hanack

Ehrenwörtliche Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit mit dem Thema
Prozessoptimierung und Architekturentwicklung am Beispiel "Fremdvertragserfassung" im Umfeld der SaaS CRM Plattform eines Versicherungsunternehmens

selbstständig und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel angefertigt habe.

Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten und nicht veröffentlichten Schriften entnommen wurden, sind als solche kenntlich gemacht.

Ich versichere zudem, dass die eingereichte elektronische Fassung mit der gedruckten Fassung übereinstimmt

Die Arbeit ist in gleicher oder ähnlicher Form oder auszugsweise im Rahmen einer anderen Prüfung noch nicht vorgelegt worden.

Grenzach-Wyhlen, 30. September 2024

Jacob Ruhnau

Hinweis zum Umfang der Arbeit

Der Textteil der vorliegenden Arbeit - beginnend mit der Einleitung bis abschließlich Quellenverzeichnis - umfasst 11 Seiten.

Freigabe der Arbeit

Die vorliegende Arbeit wurde durch das Ausbildungsunternehmen Helvetia Versicherungen, Basel CH inhaltlich geprüft und zur Vorlage an der DHBW Lörrach, Studiengang Wirtschaftsinformatik – Application Management, freigegeben.

Basel, 30. September 2024

Kevin Gadiant

Kurzfassung

tbd

Inhalt

Ehrenwörtliche Erklärung	II
Freigabe der Arbeit	III
Kurzfassung	IV
Inhalt	V
Abkürzungsverzeichnis	VII
1 Einleitung	1
1.1 Themenabgrenzung und Problemstellung	1
1.2 Zielsetzung der Arbeit	1
1.3 Vorgehensweise und Aufbau der Arbeit	2
2 Theoretische Grundlagen und Wissenschaftliche Vorgehensweise	4
2.1 Geschäftsprozessmanagement	4
2.1.1 Geschäftsprozess-Lebenszyklen	4
2.1.2 Geschäftsprozess Modellierung	5
2.2 Software Architektur	6
2.2.1 Event-Driven Architektur	7
2.2.2 Microservices-Architektur	8
2.2.3 Interaktion in verteilten Systemen	8
2.2.4 Event Plattform: Kafka	10
2.3 Vorgehensweise	11
3 Analyse des Ist-Prozesses "Fremdvertrag Erfassung"	12
3.1 Bestehender Prozess	12
3.2 Aktuelle Probleme	13
3.3 Ungenutzte Potenziale	14
3.4 Mögliche Verbesserungen	15
4 Konzeption und Technischer Entwurf des Soll-Prozesses "Fremdvertrag Erfassung"	17
4.1 Soll-Prozess Erarbeitung	17
4.1.1 Datenerfassung und Digitalisierung (0. – 1. in Abbildung 2)	18
4.1.2 Datenverarbeitung und Optimierung (2. – 3. in Abbildung 2)	18

4.1.3	Integration und Speicherung (4 – 5a in Abbildung 2)	18
4.1.4	Data-Analytics (5b in Abbildung 2)	19
4.1.5	Automatisierte Verkaufschancen-Generierung (6. – 9. In Abbildung 2)	19
4.2	Technische Architektursicht	19
4.2.1	Technische Anforderungen	20
4.2.2	Beteiligte Systeme und Komponenten	21
4.2.3	Event Verwaltung & Prozess Choreografie	22
4.2.4	Kafka Topics	24
4.3	Weiteres Projektvorgehen	26
5	Fazit	28
	Literaturverzeichnis	X
	Abbildungen	XII
	Anhang	XVII

Abkürzungsverzeichnis

BPM	Business Process Management
BPMN	Business Process Modelling Language
CRM	Customer Relationship Management
DWH	Data-Warehouse
EDA	Event-Driven Architecture
OCR	Optical Character Recognition
TAM	Technology Acceptance Model

1 Einleitung

1.1 Themenabgrenzung und Problemstellung

In einer zunehmend digitalisierten Welt sind mobile Endgeräte wie Smartphones und Tablets ständige Begleiter, sowohl im privaten als auch im geschäftlichen Kontext. Insbesondere im Vertrieb, wo Außendienstmitarbeiterinnen und -mitarbeiter direkt beim Kunden vor Ort agieren, spielt die effiziente Nutzung von IT-Systemen eine entscheidende Rolle (Müller & Thienen, 2016, S. 67). Die Möglichkeit, jederzeit und überall auf relevante Informationen zugreifen zu können, erhöht nicht nur die Produktivität, sondern ist auch essentiell, um den steigenden Anforderungen an Effizienz, Flexibilität und Kundenservice gerecht zu werden. Dennoch erweisen sich bestehende Prozesse oft als unzureichend, um diesen Anforderungen gerecht zu werden.

Der aktuelle Prozess bei Helvetia zur Erfassung von Fremdverträgen – das heißt von Versicherungsverträgen zwischen Helvetias Kundinnen und Kunden und Wettbewerbern – wird auf einer CRM-Plattform ausgeführt, die als SaaS betrieben wird. Die Funktionsnutzung der Fremdvertragserfassung erfolgt über die vom SaaS-Provider zur Verfügung gestellte mobile App auf den Smartphones der Vertriebsmitarbeitenden.

Die Situation der Außendienstmitarbeitenden beim Kunden vor Ort erfordert eine intuitive und effiziente Prozessgestaltung. Jedoch weist der Prozess deutliche Schwächen auf: Obgleich der skizzierte Prozess als funktionsfähig zu betrachten ist, bestehen signifikante Problembereiche in den Bereichen Datenverarbeitung, Systemintegration und Automatisierung. Vor allem die Digitalisierung von Dokumenten weist Schwachstellen auf, sodass eine maschinelle Verarbeitung nicht möglich ist. Dies führt zu einer deutlichen Einschränkung des effektiven Nutzens der vorhandenen IT-Landschaft.

Auf betriebswirtschaftlicher Ebene stellt dies nicht nur ein Hindernis für die Effizienz der internen Abläufe dar, sondern es gehen auch potenzielle Wettbewerbsvorteile, durch die nicht durchgeführte systematische und automatisierte Datenverwertung verloren.

1.2 Zielsetzung der Arbeit

Ziel dieser Arbeit ist es, den bestehenden Prozess der Erfassung und Verarbeitung von Fremdverträgen bei Helvetia grundlegend zu analysieren und eine optimierte Lösung auf prozessualer Ebene zu erarbeiten und auf technologischer Ebene

herzuleiten. Dabei steht die Schaffung eines zukunftsfähigen Prozesses im Mittelpunkt, der die bestehenden Kernprobleme adressiert und gleichzeitig eine nachhaltige Verbesserung der betrieblichen Effizienz ermöglicht. Der neue Prozess soll darauf abzielen, die vollständige Automatisierung und Optimierung der Datenverarbeitung zu erreichen. Dies umfasst insbesondere die Funktionalität, digitale Dokumente maschinell auszuwerten und Informationen daraus systematisch in die Systemlandschaft zu integrieren. Die dabei extrahierten Daten müssen konsistent verarbeitet und validiert werden, um sicherzustellen, dass sie zuverlässig in den bestehenden Geschäftsprozessen genutzt werden können.

Aktuell erfordert der Prozess einen hohen manuellen Aufwand, ohne dass ein klarer Nutzen für die Vertriebsmitarbeiter erkennbar ist. Gemäß dem Technology Acceptance Model (TAM) von Davis beeinflusst die wahrgenommene Nützlichkeit eines Systems maßgeblich die Akzeptanz und Bereitschaft zur Nutzung durch die Anwender (Davis, 1989, S. 331). Der neue Prozess soll so gestaltet werden, dass der wahrgenommene Aufwand durch einen erkennbaren Nutzen gerechtfertigt wird. Dabei ist es entscheidend, die Nützlichkeit des Systems zu verbessern, indem automatisierte Mechanismen integriert und manuelle Eingaben reduziert werden. Dies steigert den wahrgenommenen Mehrwert und fördert die aktive Nutzung des Systems.

Ein zentraler Aspekt dieser Optimierung liegt in der Integration der verschiedenen Systemkomponenten innerhalb der IT-Landschaft von Helvetia. Der neue Prozess muss in der Lage sein, Informationen nicht nur im CRM zu generieren, sondern auch mit anderen relevanten Unternehmenssystemen zu interagieren, um eine nahtlose Wertschöpfungskette zu gewährleisten. Dies erfordert die Schaffung einer flexiblen und anpassungsfähigen Architektur, welche zukünftige Erweiterungen und Anpassungen ermöglicht, ohne bestehende Funktionen signifikant zu beeinträchtigen.

Auf betriebswirtschaftlicher Ebene soll der optimierte Prozess nicht nur die Effizienz interner Abläufe steigern, sondern auch Wettbewerbsvorteile sichern, indem er Helvetia in die Lage versetzt, Kundeninformationen systematisch auszuwerten und Verkaufschancen proaktiv zu generieren. Langfristig wird dadurch die Positionierung von Helvetia als innovativer und datengetriebener Anbieter gestärkt.

1.3 Vorgehensweise und Aufbau der Arbeit

Die vorliegende Arbeit zielt darauf ab, den bestehenden Prozess der Fremdvertragserfassung bei Helvetia zu analysieren und auf dieser Grundlage einen optimierten Soll-

Einleitung

Prozess zu entwickeln. Um dieses Ziel zu erreichen, wird ein methodisches Vorgehen gewählt, das sowohl betriebswirtschaftliche als auch technologische Aspekte berücksichtigt. In Kapitel 3 erfolgt eine detaillierte Ist-Analyse des aktuellen Prozesses, um Schwachstellen und Ineffizienzen zu identifizieren. Auf Basis dieser Erkenntnisse wird in Kapitel 4 ein neuer Soll-Prozess konzipiert, der die identifizierten Defizite adressiert und die Prozessleistung nachhaltig steigern soll. Dabei werden in einem Architektur-Entwurf moderne Technologien und Methoden berücksichtigt, um eine effiziente Umsetzung zu gewährleisten.

Die theoretischen Grundlagen, die diesem Vorgehen zugrunde liegen, werden in Kapitel 2 dargelegt. Hier werden relevante Konzepte des Geschäftsprozessmanagements sowie der Softwarearchitektur vorgestellt, die als Basis für die Analyse und Konzeption dienen.

2 Theoretische Grundlagen und Wissenschaftliche Vorgehensweise

2.1 Geschäftsprozessmanagement

Im Rahmen dieser Arbeit wird der Optimierung des bestehenden IT-Prozesses durch einen fundierten theoretischen Ansatz des Geschäftsprozessmanagements (Business Process Management, BPM) begegnet. Geschäftsprozesse stellen eine zentrale Struktur innerhalb eines Unternehmens dar, welche die Geschäftsaktivitäten und deren Reihenfolge definieren, um ein Geschäftsziel zu erreichen (Weske, 2012, S. 4). Um einen strukturierten Ansatz zur Neugestaltung des IT-Prozesses der Helvetia zu gewährleisten, stützt sich diese Arbeit auf zwei wesentliche Konzepte des BPM: den Prozesslebenszyklus und die Prozessmodellierung. Dabei steht der Prozesslebenszyklus als übergeordneter Rahmen im hierarchischen Verhältnis zur Prozessmodellierung, welche die einzelnen Phasen des Lebenszyklus detailliert beschreibt und analysiert. Im Folgenden werden zunächst die Lebenszyklen eines Geschäftsprozesses erläutert.

2.1.1 Geschäftsprozess-Lebenszyklen

Der Prozesslebenszyklus beschreibt die Phasen, die ein Geschäftsprozess durchläuft, von der Konzeption bis zur kontinuierlichen Verbesserung. Dabei kann man zwischen folgenden Phasen unterscheiden:

1. **Design und Modellierung:** Diese Phase bildet die Grundlage für die Gestaltung eines neuen oder die Anpassung eines bestehenden Prozesses. Hierbei wird der Prozess zunächst definiert und in die Geschäftsprozesslandkarte eingeordnet, die Akteure und Systeme werden identifiziert und die Anforderungen präzisiert. Die Prozessmodellierung spielt in dieser Phase eine zentrale Rolle. Eine formale Darstellung des Prozesses, beispielsweise mithilfe von Business Process Model and Notation (BPMN), erleichtert das Verständnis und die Kommunikation zwischen den verschiedenen Stakeholdern (Weske, 2012, S. 11f.).
2. **Implementierung:** In der Implementierungsphase wird der entworfene Prozess in das IT-System integriert und technisch realisiert. Dabei spielt die Auswahl geeigneter Technologien, wie z. B. der Einsatz von automatisierten Workflows und die Integration bestehender Systeme, eine entscheidende Rolle (Weske, 2012, S. 13f.).

3. **Ausführung:** Nachdem der Prozess implementiert wurde, erfolgt in dieser Phase die praktische Ausführung in der realen Umgebung. Die modellierten und implementierten Geschäftsprozesse werden nun getestet, um ihre Effizienz und Funktionalität zu bewerten (Weske, 2012, S. 14 f.).
4. **Überwachung und Optimierung:** Nach der Ausführung wird der Prozess kontinuierlich überwacht, um sicherzustellen, dass er den Anforderungen entspricht und eventuelle Engpässe oder Optimierungspotenziale frühzeitig identifiziert werden (Weske, 2012, S. 15f.). Freund und Rücker betonen, dass der Geschäftsprozess als dynamisches Element in Unternehmen verstanden werden muss, das regelmäßig an neue Herausforderungen und Geschäftsanforderungen angepasst werden sollte (Freund & Rücker, 2019, S. 5).

Das Durchlaufen der vier Phasen ermöglicht eine ganzheitliche Betrachtung des Prozesslebenszyklus. Dadurch soll sichergestellt werden, dass ein Geschäftsprozess nicht nur kurzfristige Verbesserungen bringt, sondern auch langfristig anpassungsfähig bleibt und auf künftige betriebliche Anforderungen reagieren kann (Weske, 2012, S. 15).

Eine zentrale Voraussetzung für den Erfolg der skizzierten Prozesse ist die klare Modellierung derselben, welche in der ersten Phase gestaltet wird. Die Prozessmodellierung soll dazu beitragen, dass komplexe Abläufe verständlich und präzise erfasst werden. Sie bildet die Basis für die spätere technische Umsetzung und Optimierung. Im Folgenden wird mit Bezug auf die Geschäftsprozessmodellierung auf die wesentliche Rolle der Modellierung im Geschäftsprozessmanagement eingegangen.

2.1.2 Geschäftsprozess Modellierung

Die Geschäftsprozessmodellierung ist ein wesentliches Instrument im Geschäftsprozessmanagement, um komplexe Abläufe innerhalb eines Unternehmens transparent und verständlich darzustellen. Sie ermöglicht es, Prozesse zu analysieren, zu dokumentieren und Optimierungsmöglichkeiten zu identifizieren. Durch die visuelle Darstellung werden Abläufe für alle Beteiligten greifbar, was die Kommunikation zwischen Stakeholdern erleichtert und eine gemeinsame Basis für Verbesserungen schafft (Weske, 2012, S. 73).

Es lassen sich hierbei verschiedene Ebenen der Prozessmodellierung unterscheiden: die strategische Ebene, die organisatorische Ebene, die operative Ebene und die implementierte Ebene. In dieser Hierarchie bestimmt jeweils die höhere Ebene die

nächstuntere, während die untere Ebene die nächsthöhere umsetzt. Organisatorische Geschäftsprozesse repräsentieren dabei übergeordnete, grobgranulare Geschäftsaktivitäten eines Unternehmens. Sie setzen sich typischerweise aus mehreren operativen Geschäftsprozessen zusammen, die gemeinsam zur Erfüllung des organisatorischen Prozesses beitragen. In den operativen Geschäftsprozessen werden die spezifischen Aktivitäten und deren Beziehungen detailliert beschrieben, wobei Aspekte der technischen Implementierung zunächst unberücksichtigt bleiben. Diese operativen Prozesse werden mittels Geschäftsprozessmodellen visualisiert und dienen als fundamentale Basis für die Entwicklung implementierter Geschäftsprozesse. Implementierte Geschäftsprozesse wiederum enthalten umfassende Informationen über die konkrete Ausführung der Prozessaktivitäten sowie über die technische und organisatorische Umgebung, in der diese stattfinden (Weske, 2012, S. 17ff.).

Die Business Process Model and Notation (BPMN) gilt als die standardisierte Methode zur grafischen Darstellung von Prozessen. BPMN wird häufig verwendet, um technische Anforderungen auf der einen Seite und fachliche Anforderungen auf der anderen Seite in eine gemeinsame visuelle Sprache zu überführen. Dadurch können sowohl technische Entwickler als auch nicht-technische Stakeholder ein gemeinsames Verständnis über den Ablauf eines Geschäftsprozesses entwickeln (Freund & Rücker, 2019, S. 5).

Die Prozessmodellierung mit BPMN stellt somit einen systematischen Ansatz zur Neugestaltung und Optimierung eines Geschäftsprozesses dar. Sie schafft Transparenz, stellt sicher, dass die Prozessstrukturen klar und konsistent definiert sind und fördert eine präzise Umsetzung derer. Insbesondere bei der Entwicklung einer Softwarearchitektur ist die durchgängige Anwendung von Modellen entscheidend, um die Prozesse strukturiert und logisch in das IT-System zu integrieren: Während die Prozessmodellierung den Ablauf visualisiert, bildet die Softwarearchitektur die technische Umsetzung dieses Ablaufs ab (Weske, 2012, S. 120). Ansätze zur Gestaltung einer Software Architektur werden im folgenden Kapitel beschrieben.

2.2 Software Architektur

Die Software Architektur bildet die technologische Grundlage für die Umsetzung der modellierten Geschäftsprozesse. Eine gut strukturierte Architektur ermöglicht es, die Prozesse performant, skalierbar und flexibel zu gestalten, sodass sie den Anforderungen des Unternehmens gerecht werden (Fowler, 2013, S. 4). In diesem Kapitel werden

die Merkmale der Event-Driven Architektur, Microservice Architektur sowie die Bedeutung der Ereignissteuerung und Ereigniserzeugung mit Apache Kafka als Schlüsselkomponenten der modernen Software Architektur beleuchtet.

2.2.1 Event-Driven Architektur

Die Event-Driven Architektur (EDA) ist ein Ansatz in der Softwarearchitektur, bei dem Ereignisse als zentrales Element der Kommunikation und Interaktion zwischen verschiedenen Systemkomponenten dienen. Ein Ereignis stellt eine signifikante Änderung oder einen wichtigen Zustand in einem System dar, auf den andere Komponenten reagieren können. EDA fördert die lose Kopplung zwischen Komponenten, da der Erzeuger eines Ereignisses (Producer) nicht wissen muss, welche Komponenten das Ereignis konsumieren (Consumers) oder wie sie darauf reagieren werden (Wolff, 2018, S. 137).

Ein wesentliches Grundprinzip der EDA ist die Asynchronität in der Kommunikation: Ereignisse werden erzeugt und veröffentlicht, ohne dass der Erzeuger auf eine unmittelbare Antwort warten muss. Dies ermöglicht es Systemen, effizient zu skalieren und eine höhere Performance zu erzielen, da Komponenten unabhängig voneinander operieren können (Richards, 2015, S. 11f.).

Ereignisverarbeitung kann aber genauso auch zusätzlich in Parallelität (gleichzeitige Ausführung mehrerer Prozesse) eingesetzt werden, um die Leistungsfähigkeit weiter zu steigern. Obwohl Asynchronität und Parallelität unterschiedliche Konzepte sind können beide in einer EDA implementiert werden, um den spezifischen Anforderungen des Systems gerecht zu werden. Die Kommunikation erfolgt typischerweise über ein zentrales Nachrichtensystem oder einen Event-Bus, der die Ereignisse verteilt und somit die Entkopplung der Komponenten unterstützt (ebd.).

Die Vorteile der Event-Driven Architektur sind vielfältig:

- **Resilienz des Systems:** Da die Komponenten lose gekoppelt sind, führt ein Ausfall einer Komponente nicht zwangsläufig zum Zusammenbruch des gesamten Systems (Kleppmann, 2019, Abschnitt I.1.2).
- **Hohe Skalierbarkeit:** Systeme können besser mit variablen Lasten umgehen, indem sie Ereignisse asynchron verarbeiten und bei Bedarf parallele Verarbeitung nutzen (Kleppmann, 2019, Abschnitt I.1.7).
- **Wartbarkeit:** Neue Funktionen oder Dienste können hinzugefügt werden, indem sie auf bestehende Ereignisse reagieren oder neue Ereignisse erzeugen,

ohne dass bestehende Komponenten angepasst werden müssen (Kleppmann, 2019, Abschnitt I.1.11)

Gleichsam wird durch die Dezentralisierung eine erhöhte Komplexität hergestellt, insbesondere im Hinblick auf mögliche Ausfälle eines angesprochenen Services (Kratzke, 2024, S. 174). Das folgende Kapitel erläutert Herangehensweisen, um diese Herausforderungen zu adressieren.

2.2.2 Microservices-Architektur

Die Microservices-Architektur ist ein moderner Ansatz in der Softwareentwicklung, der darauf abzielt, komplexe Anwendungen durch die Aufteilung in kleine, unabhängige Dienste zu vereinfachen. Die Microservices-Architektur basiert auf der Idee, eine Anwendung als Sammlung von Diensten zu strukturieren. Jeder dieser Dienste erfüllt eine spezifische, klar abgegrenzte Geschäftsanforderung und kommuniziert über leichtgewichtige Schnittstellen mit anderen Diensten (Newman, 2021, Kapitel 1.1).

Ein wesentliches Merkmal der Microservices-Architektur ist die Unabhängigkeit der einzelnen Dienste: Diese können eigenständig entwickelt, bereitgestellt und skaliert werden, was die Agilität in der Softwareentwicklung fördert. Die klare Abgrenzung von Verantwortlichkeiten reduziert die Komplexität der einzelnen Dienste und verbessert dadurch deren Wartbarkeit und Erweiterbarkeit. Zudem ermöglicht diese Architektur den Einsatz verschiedener Technologien und Programmiersprachen, da die Dienste über definierte Schnittstellen miteinander interagieren. Darüber hinaus bietet die Microservices-Architektur eine individuelle Skalierung der Dienste, sodass Ressourcen effizient genutzt werden können (Richards, 2015, S. 34f.).

Die Aufteilung in Microservices ermöglicht eine unabhängige Aktualisierung und Erweiterung einzelner Dienste, was zu einer schnelleren Anpassung an neue Geschäftsanforderungen führt (Newman, 2021, Kapitel 1.2 Absatz 1).

Allerdings stellen Koordination und Interaktion dieser autonomen Dienste eine besondere Herausforderung dar (Wolff, 2018, S. 167). Das folgende Kapitel behandelt verschiedene Aspekte, welche hierbei zu beachten sind.

2.2.3 Interaktion in verteilten Systemen

Unterschiedliche Interaktionsmodelle bieten spezifische Vor- und Nachteile in Bezug auf *Kommunikation*, *Konsistenz*, *Koordination* und *Kopplung* welche maßgeblich die Eigenschaften des Systems prägen.

Ein wichtiger Aspekt ist die **Kommunikation** zwischen den Services, die entweder synchron oder asynchron erfolgen kann. Bei synchroner Kommunikation werden Services direkt aufgerufen und blockieren, bis eine Antwort eintrifft, was die Kontrolle über den Prozessfluss vereinfacht. Asynchrone Kommunikation, bei der Services auf Ereignisse reagieren und indirekt über Nachrichtenkanäle interagieren, fördert dagegen Skalierbarkeit und Resilienz, erfordert aber ein komplexeres Design und führt möglicherweise zu höherer Latenz (Ford et al., 2021, Kapitel 11.3).

Die Frage der **Konsistenz** stellt ein weiteres Spannungsfeld dar. Während atomare Konsistenz garantiert, dass Transaktionen entweder vollständig durchgeführt oder rückgängig gemacht werden, führt sie zu einer stärkeren Kopplung der Services. Eine weniger strikte Ausprägung, die sogenannte "eventual consistency", erlaubt temporäre Inkonsistenzen, was mehr Flexibilität und Verfügbarkeit ermöglicht, jedoch Mechanismen zur Fehlererkennung und Kompensation erforderlich macht (Ford et al., 2021, Kapitel 12.2).

Kopplung ist ein wesentlicher Aspekt, der darstellt, wie stark einzelne Services voneinander abhängig sind. Eine enge Kopplung führt zu mehr Kontrolle und einfacheren Integrationen, erschwert jedoch die Anpassung oder Skalierung einzelner Komponenten. Lose Kopplung, die durch asynchrone Kommunikation und dezentrale Koordination begünstigt wird, fördert die Unabhängigkeit der Services und erleichtert die Skalierbarkeit, bringt aber auch komplexere Mechanismen zur Koordination und Fehlerbehandlung mit sich (Ford et al., 2021, Kapitel 2.1.3).

Schliesslich stehen in Bezug auf **Koordination** zwei grundlegende Ansätze zur Verfügung: Orchestrierung und Choreografie. Bei der Orchestrierung steuert ein zentraler Orchestrator den Prozessablauf und kennt die Abfolge der Service-Interaktionen. Dies erleichtert das Monitoring und die Fehlerbehandlung, führt aber zu starker Kopplung und kann einen Single Point of Failure darstellen. Im Gegensatz dazu funktioniert die Choreografie dezentral, indem jeder Service autonom auf Ereignisse reagiert. Dies fördert lose Kopplung und Skalierbarkeit, macht jedoch die Nachverfolgbarkeit des Gesamtprozesses und die Fehlerbehandlung komplexer (Ford et al., 2021, Kapitel 12.2). Die diskutierten Ansätze der Kommunikation, Konsistenz, Koordination und Kopplung spielen eine zentrale Rolle in der Architektur verteilter Systeme. Um diese Konzepte effizient umzusetzen, sind robuste Technologien erforderlich, die die asynchrone Kommunikation, lose Kopplung und die Event-getriebene Verarbeitung unterstützen. Im

folgenden Kapitel wird Apache Kafka, zentrales Element, um die zuvor genannten Herausforderungen in verteilten Systemen zu bewältigen, behandelt.

2.2.4 Event Plattform: Kafka

Die Implementierung einer ereignisgesteuerten Architektur erfordert eine leistungsfähige Plattform zur Verarbeitung und Verwaltung von Ereignissen. Apache Kafka hat sich in diesem Kontext als De-facto-Standard etabliert und bietet die notwendigen Funktionen, um große Mengen an Daten in Echtzeit zu verarbeiten und zu verteilen (Zelenin & Kropp, 2022, S. 40).

Apache Kafka ist ein verteiltes Streaming-System, das ursprünglich von LinkedIn entwickelt und später als Open-Source-Projekt unter der *Apache Software Foundation* veröffentlicht wurde (*Apache Kafka Documentation*, o. J., Kapitel 1.2). Es dient als zentrales Rückgrat für die Verarbeitung von Ereignisströmen und ermöglicht es, Nachrichten in hoher Geschwindigkeit und mit niedriger Latenz zu veröffentlichen, zu konsumieren, zu speichern und zu verarbeiten (Zelenin & Kropp, 2022, S. XII).

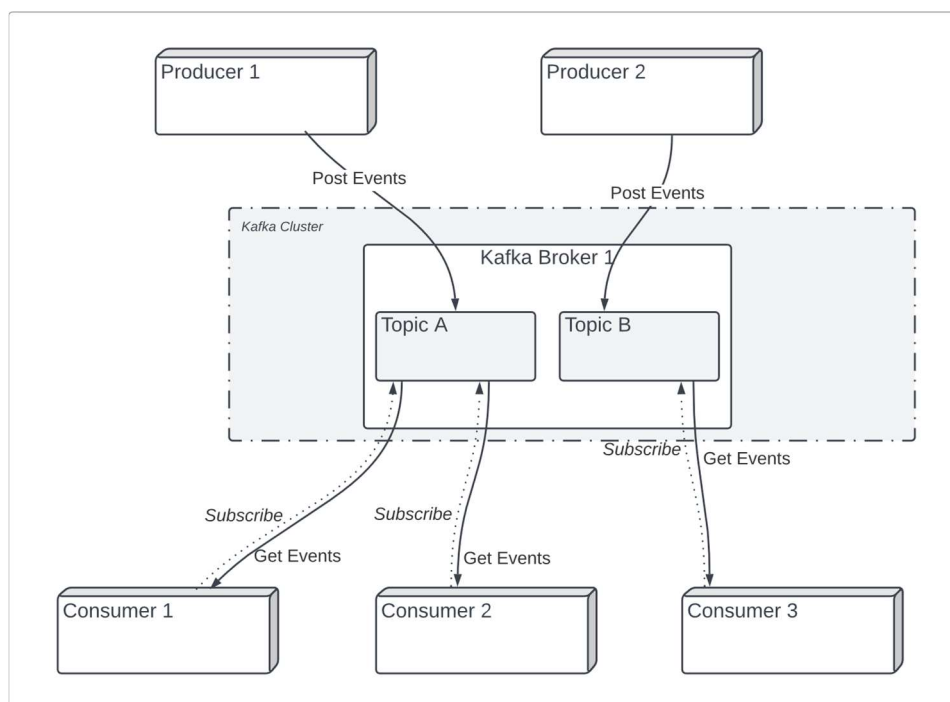


Abbildung 1: Schematische Darstellung der Funktion von Apache Kafka

Schematisch ist Kafka in Abbildung 1 skizziert wobei ein zentrales Konzept sogenannte *Topics* sind, die als Kategorien oder Feeds für Nachrichten dienen. *Producer* publizieren Nachrichten auf diese Topics, während *Consumer* diese Nachrichten abonnieren. Kafka zeichnet sich durch seine Fähigkeit aus, Nachrichten dauerhaft zu

speichern und sie somit für mehrere Consumer verfügbar zu machen. Dies unterstützt die Wiederherstellung und die Verarbeitung von Ereignissen zu einem späteren Zeitpunkt (Kleppmann, 2019, S. 23ff.).

Durch seine verteilte Architektur erreicht Kafka eine hohe Skalierbarkeit und Fehlertoleranz. Es ermöglicht die horizontale Skalierung durch Hinzufügen weiterer Broker, die zusammen einen Kafka-Cluster bilden (*Apache Kafka Documentation*, o. J., Kapitel 1.1). Dies ist besonders wichtig für Anwendungen, die eine hohe Datenrate und Zuverlässigkeit erfordern (Zelenin & Kropp, 2022, S. 31).

Im Kontext von Microservices ermöglicht Kafka eine asynchrone Kommunikation zwischen verschiedenen Komponenten eines Verteilens Systems. Dies fördert die lose Kopplung und Unabhängigkeit der Microservices, da sie nicht direkt miteinander kommunizieren müssen, sondern über Ereignisse interagieren können. Dadurch können Dienste unabhängig entwickelt, skaliert und gewartet werden, was die Flexibilität und Agilität des Systems erhöht (Wolff, 2018, S. 98f.).

2.3 Vorgehensweise

Die methodische Herangehensweise dieser Arbeit orientiert sich an den Prinzipien des Geschäftsprozessmanagements und stützt sich insbesondere auf den Geschäftsprozess-Lebenszyklus. Dieser Lebenszyklus beschreibt die Phasen, die ein Geschäftsprozess von der Konzeption bis zur kontinuierlichen Verbesserung durchläuft und umfasst typischerweise *Design und Modellierung*, *Implementierung*, *Ausführung*, *Überwachung und Optimierung* (vgl. Kapitel 2.1.1).

In dieser Arbeit wird dieser Lebenszyklus angewendet, um den Prozess der Fremdvertragserfassung systematisch zu analysieren und zu optimieren. Die Ist-Analyse in Kapitel 3 ist Teil der Phase der *Überwachung und Optimierung*, während die Konzeption des Soll-Prozesses in Kapitel 4 der Phase des *Designs und der Modellierung* entspricht. Die Entwicklung der Softwarearchitektur ist der *Implementierungsphase* zugeordnet.

Durch die Anwendung des Geschäftsprozess-Lebenszyklus wird eine strukturierte und wissenschaftlich fundierte Vorgehensweise angewandt, die es ermöglicht, den Prozess ganzheitlich zu betrachten und nachhaltige Verbesserungen zu erzielen. Diese Herangehensweise gewährleistet, dass sowohl betriebswirtschaftliche Ziele als auch technologische Anforderungen berücksichtigt werden, um eine effektive und effiziente Lösung für Helvetia zu entwickeln.

3 Analyse des Ist-Prozesses "Fremdvertrag Erfassung"

Die Erfassung von Fremdverträgen bei Helvetia dient dazu, Informationen über Versicherungsverträge zu sammeln, die Kundinnen und Kunden bei Wettbewerbern abgeschlossen haben. Diese Daten sind potenziell wertvoll für Cross-Selling-Strategien und zur Identifizierung neuer Verkaufschancen. Für die Analyse als Teil der Phase 4 *Überwachung und Optimierung* des Geschäftsprozess Lebenszyklus (vgl. Kapitel 2.1.1.) wird zunächst der bestehende Prozess der Fremdvertrag Erfassung beleuchtet.

3.1 Bestehender Prozess

Der Prozess, um einen Fremdvertrag zu erfassen, wird ausschließlich über die als SaaS betriebene CRM-Plattform abgewickelt. Vertriebsmitarbeitende greifen auf diese von Geschäftsgeräten (Laptops oder Smartphones) über eine Desktopanwendung oder mobile App zu.

Im typischen Ablauf beginnt der Prozess mit dem persönlichen Kontakt zwischen dem Vertriebsmitarbeitenden und dem Kunden. Während eines Beratungsgesprächs bittet der Vertriebsmitarbeitende den Kunden um Einsicht in dessen bestehende Versicherungsverträge bei anderen Anbietern. Der Kunde stellt daraufhin die physischen Dokumente der Fremdverträge zur Verfügung. Der Vertriebsmitarbeitende nutzt die mobile CRM-App, um diese Dokumente zu erfassen, wobei hierfür eine dedizierte Aktion mit Dialog auf einem Kundendatensatz zur Verfügung steht. Die Kopfdaten des Vertrags (Nummer, Fremdversicherer, Datum Inkrafttreten, Datum Außerkrafttreten, Verlängerungsvereinbarung, Produktbranche) werden hierbei manuell erfasst. Zusätzlich kann fakultativ eine digitalisierte Version des Dokuments abgelegt werden. Hierbei werden die einzelnen Seiten der Verträge mit der Kamera des Smartphones fotografiert. Die App speichert diese Bilder im JPEG-Format und lädt sie direkt in das CRM-System hoch. Der Fremdvertrag wird als eigenständiges Objekt angelegt und direkt mit dem Kundendatensatz verknüpft.

In der Praxis zeigt sich jedoch, dass viele Vertriebsmitarbeitende eine alternative Methode durchführen: Statt die Dokumente vor Ort zu digitalisieren, nehmen sie die Originaldokumente des Kunden mit. Anschließend werden die Dokumente an Mitarbeitende des Vertriebssupports im Büro übergeben. Der Vertriebssupport erhält den Auftrag, die Fremdverträge zu digitalisieren. Hierzu werden stationäre Scanner

verwendet, die eine PDF-Datei erzeugen. Der Fremdvertrag wird über dieselbe Funktionalität wie auf den mobilen Geräten analog an der Desktopanwendung als Objekt im CRM-System angelegt, mit der Datei angereichert und mit dem Kundendatensatz verknüpft. Das Originaldokument wird danach an den Kunden zurückgesandt, entweder per Post oder durch persönliche Übergabe beim nächsten Treffen mit dem Vertriebsmitarbeitenden.

Nach der Dokumentenerfassung kann der Vertriebsmitarbeitende optional manuell eine Verkaufschance im CRM-System anlegen. Dies erfolgt über die Funktion "Verkaufschance aus Fremdvertrag", mit der potenzielle Verkaufschancen systematisch erfasst und verfolgt werden können.

3.2 Aktuelle Probleme

Obwohl bei Helvetia ein formalisiertes Verfahren zur Erfassung von Fremdverträgen etabliert ist, hat die Auswertung von Nutzerfeedback und Nutzungsanalysen aufgetan, dass in der Praxis erhebliche Probleme auftreten, die die Effizienz und Effektivität dieses Prozesses beeinträchtigen. Ein zentrales Problem besteht darin, dass die erfassten Informationen zwar als Objekt und ggf. auch ergänzt um das digitalisierte Dokument im CRM-System abgelegt werden, jedoch **nicht systematisch konsumier- oder weiter verarbeitbar**. Dies ist daher nicht möglich, da die digitalisierten Fremdvertragsdokumente oftmals **in fragmentierter Form gespeichert** werden. Häufig werden einzelne Seiten eines Vertrags als separate JPEG-Dateien abgelegt, was zu einer unübersichtlichen und ineffizienten Datenstruktur führt: die konsolidierte Sicht auf einen Vertrag ist erschwert und erhöht den administrativen Aufwand bei der Weiterverarbeitung der Dokumente. So ist die Speicherung als Bilddatei ohne weitere Verarbeitungsschritte problematisch, da die enthaltenen Informationen nur bedingt maschinell auslesbar sind.

Darüber hinaus besteht eine mangelnde Integration der erfassten Daten in die übrige IT-Landschaft. Die **im CRM-System isolierte Ablage** der Vertragsdaten und digitalisierten Dokumente ohne Möglichkeit der systematischen Weiterverarbeitung verhindert die Nutzung der Informationen durch andere Systeme oder Anwendungen.

Ein weiterer kritischer Aspekt ist das fehlende Incentive für die Vertriebsmitarbeitenden, den vorgesehenen Prozess ordnungsgemäß zu durchlaufen. Die manuelle Eingabe der Vertragskopfdaten und das zusätzliche Digitalisieren der Dokumente werden als zeitaufwändig und wenig nutzbringend empfunden. Da die Vertriebsmitarbeitenden

keinen unmittelbaren Mehrwert aus diesem Prozess ziehen, sind sie wenig motiviert, die notwendigen Schritte sorgfältig auszuführen. Dies führt häufig dazu, dass alternative Methoden angewandt werden, wie das Mitnehmen der Originaldokumente und die spätere Digitalisierung durch den Vertriebssupport. Diese Vorgehensweise erhöht den administrativen Aufwand erheblich, verlängert die Prozessdauer und birgt Risiken hinsichtlich Datenschutzes und Datensicherheit.

Insgesamt führt die Kombination aus fragmentierter Datenablage, fehlender maschineller Lesbarkeit, mangelnder Integration in die IT-Landschaft und ein als geringfügig nutzenbringend wahrgenommene Funktionalität durch die Endanwender zu erheblichen Ineffizienzen im Prozess.

3.3 Ungenutzte Potenziale

Die bestehenden Defizite im Prozess der Fremdvertragserfassung bei Helvetia führen dazu, dass bedeutende Potenziale ungenutzt bleiben, die für die strategische Entwicklung des Unternehmens von großer Relevanz sind: So liegt ein zentrales ungenutztes Potenzial in der **großflächigen Auswertung** der erfassten Fremdverträge. Durch eine systematische Analyse dieser Daten könnte Helvetia wertvolle Erkenntnisse über das Marktumfeld gewinnen. Die Informationen aus den Fremdverträgen bieten Einblicke in die Produkte und Konditionen der Wettbewerber, wodurch Trends identifiziert und das eigene Produktportfolio gezielt verfeinert werden könnten. Dies würde nicht nur die Wettbewerbsfähigkeit stärken, sondern auch die Entwicklung innovativer Produkte ermöglichen, die besser auf die Bedürfnisse der Kundinnen und Kunden zugeschnitten sind.

Des Weiteren bietet die **systematische Generierung von Offerten** und Verkaufschancen ein erhebliches Potenzial zur Umsatzsteigerung. Die detaillierten Informationen aus den Fremdverträgen könnten genutzt werden, um individuelle Angebote zu erstellen und Cross-Selling-Möglichkeiten zu identifizieren. Durch eine proaktive Ansprache der Kundinnen und Kunden mit maßgeschneiderten Offerten könnte die Abschlussquote erhöht und die Kundenbindung vertieft werden.

Ein weiterer wichtiger Aspekt betrifft die Datenhaltung gemäß der Unternehmensstrategie. Aktuell werden die digitalen Dokumente der Fremdverträge im CRM-System abgelegt, was zu einer ineffizienten Datenverwaltung führt. Durch die Verlagerung der Dokumentenablage in das bei Helvetia bereits vorhandene **Dokumentenarchivierungssystem** könnten Speicherressourcen optimiert und Compliance-Anforderungen

besser erfüllt werden. Eine zentrale und strukturierte Datenhaltung würde zudem die Datenintegrität erhöhen und die Grundlage für eine effektive Informationsverarbeitung schaffen.

Schließlich eröffnen sich **Chancen für die Entwicklung neuer digitaler Geschäftsmodelle**. Durch die Kombination von Datenanalyse und kundenorientierten Services könnte Helvetia innovative Angebote schaffen, die einen echten Mehrwert bieten und sich vom Wettbewerb abheben.

Die Realisierung dieser Potenziale erfordert eine strategische Neuausrichtung des Prozesses der Fremdvertragserfassung sowie Investitionen in moderne Technologien und Systeme. Möglichkeiten dies umzusetzen, werden im nächsten Kapitel erörtert.

3.4 Mögliche Verbesserungen

Die identifizierten Defizite im Prozess der Fremdvertragserfassung bei Helvetia weisen auf mehrere Verbesserungsansätze hin, wobei der zentrale Fokus auf der Steigerung der Datenwertschöpfung aus Kundendokumenten liegt, um die Effizienz interner Prozesse zu erhöhen und strategische Potenziale zu realisieren. Um dies zu erreichen, können auf mehreren Ebenen Verbesserungen umgesetzt werden, die sich auf die Digitalisierung und Weiterverarbeitung der Daten beziehen.

Ein wesentlicher Schritt ist die **Sicherstellung maschinell lesbarer Dokumente**. Durch den Einsatz von Technologien zur automatisierten Texterkennung, wie Optical Character Recognition (OCR) in Kombination mit semantischer Analyse, können die Informationen aus den Fremdverträgen effektiv extrahiert werden.

Die **automatisierte Extraktion** von Attributwerten **reduziert manuelle Eingaben** und minimiert Fehlerquellen und die Rückführung der extrahierten Daten in die entsprechenden Datensätze im CRM-System ist essenziell für eine konsistente Datenhaltung und Weiterverarbeitung. Eine nahtlose Integration fördert die Aktualität und Verlässlichkeit der Kunden- und Vertragsinformationen.

Die **Auslagerung der digitalisierten Dokumente** in das bereits vorhandene zentrale Archivierungssystem ermöglicht zudem eine effiziente und sichere Datenhaltung, optimiert Speicherressourcen und erfüllt Compliance-Anforderungen gemäß der Unternehmensstrategie.

Die Integration von **Schnittstellen zu dem Offert-System** eröffnet zusätzliche Wertschöpfungsmöglichkeiten aus den Fremdvertragsdaten. Automatisierte Angebotserstellungen auf Basis der extrahierten Informationen können den Verkaufsprozess

beschleunigen und erhöhen die Reaktionsfähigkeit gegenüber Kundenanfragen. Die **Anbindung an Data-Analytics-Plattformen** ermöglicht darüber hinaus umfassende Marktanalysen, Trendidentifikation und Optimierung des Produktportfolios.

Es ist zu berücksichtigen, dass die Reduzierung des Aufwands für Endanwender oder umfangreiche Entwicklungen aufgrund technischer Restriktionen der SaaS-CRM-Plattform begrenzt ist. So ist die Anpassbarkeit der Software durch Vorgaben des Anbieters eingeschränkt, sodass umfassende Veränderungen in Benutzeroberfläche oder Funktionalität nicht ohne Weiteres realisierbar sind.

Im folgenden Kapitel wird eine detaillierte Konzeption des Soll-Prozesses diese Verbesserungsansätze aufgreifen und konkrete Lösungsvorschläge präsentieren, wobei sowohl technische als auch organisatorische Aspekte für eine nachhaltige Prozessoptimierung berücksichtigt werden.

4 Konzeption und Technischer Entwurf des Soll-Prozesses "Fremdvertrag Erfassung"

Auf Grundlage der in Kapitel 3 identifizierten Optimierungspotenziale des aktuellen Prozesses der Fremdvertragsfassung wird in diesem Kapitel im Sinne der 1. Phase *Design und Modellierung* des Geschäftsprozess Lebenszyklus der Soll-Prozess neu entworfen und modelliert. Anhand dieser Ergebnisse beginnt anschließend die 2. Phase *Implementierung* mit dem Entwurf der technischen Architektur. Zunächst wird der Soll-Prozess in seinen einzelnen Schritten detailliert beschrieben, wobei insbesondere auf die wichtigsten Aspekte der Datenverarbeitung, -validierung und -integration eingegangen wird.

4.1 Soll-Prozess Erarbeitung

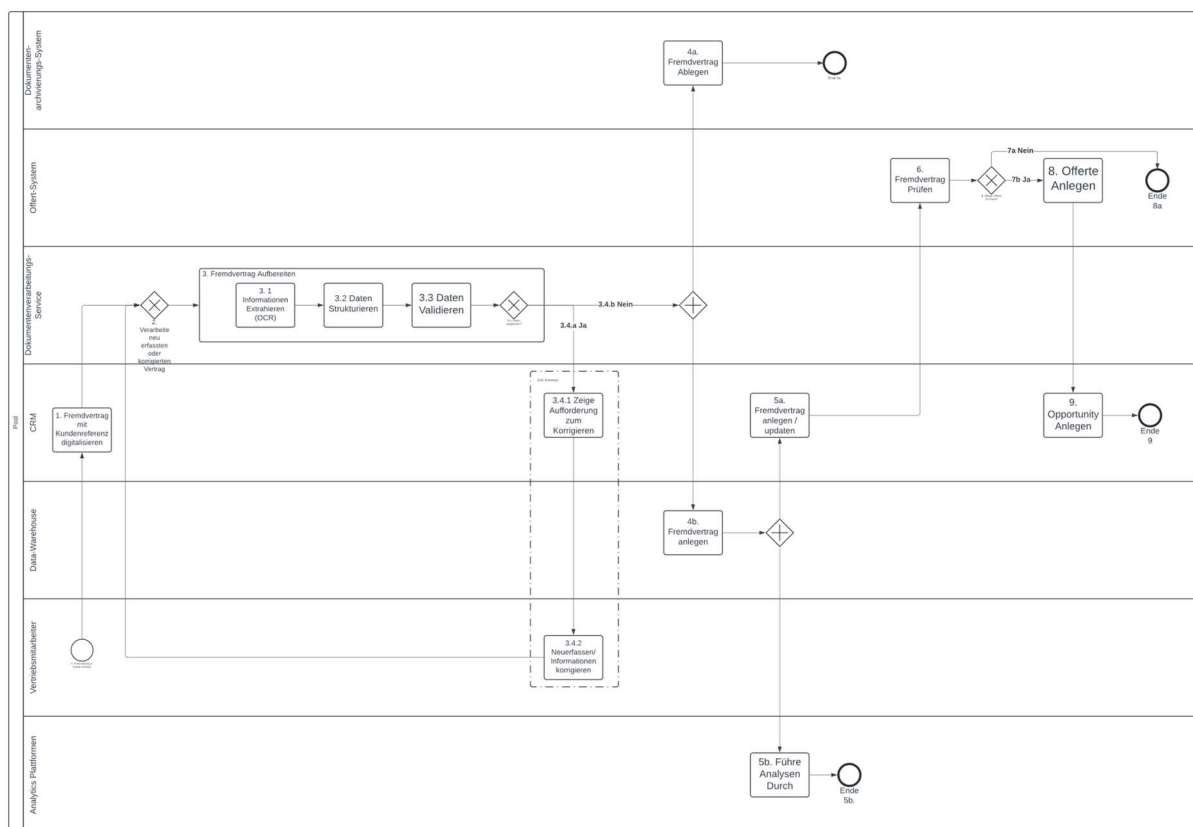


Abbildung 2: BPMN CHART "Fremdvertragsfassung" – Soll-Prozess

Der neue Prozess dient dazu, den bestehenden Fremdvertragsprozess bei der Helvetia zu optimieren, indem manuelle Aufgaben reduziert, Daten effizienter verarbeitet und die Integration mit bestehenden Systemen verbessert wird. Ziel ist es, die

Konzeption und Technischer Entwurf des Soll-Prozesses "Fremdvertrag Erfassung"

Erfassung und Verarbeitung von Fremdverträgen zu automatisieren und die extrahierten Informationen konsistent in das CRM-System zu integrieren. Dadurch sollen betriebliche Effizienz und die Wertschöpfung aus vorhandenen Daten gesteigert werden. Dieser ist in Abbildung 2 mit BPNM modelliert, wobei auf wesentliche Schwerpunkte folgend näher ausgeführt werden.

4.1.1 Datenerfassung und Digitalisierung (0. – 1. in Abbildung 2)

Im ersten Schritt des Soll-Prozesses erfolgt die Erfassung der Fremdverträge durch Vertriebsmitarbeiter über die mobile CRM-Anwendung, wobei hier die notwendigen manuellen Eingaben minimiert werden sollen. Die Digitalisierung der Verträge erfolgt unmittelbar im Kondenkontakt.

4.1.2 Datenverarbeitung und Optimierung (2. – 3. in Abbildung 2)

Nach der Erfassung und Digitalisierung werden die Daten in einer zentralen Datenverarbeitungsplattform weiter optimiert, wobei Dokumente in ein standardisiertes Format (PDF) konvertiert werden.

Um eine automatisierte Weiterverarbeitung der Dokumente zu ermöglichen, kommt im Anschluss eine Optical Character Recognition (OCR)-Technologie sowie eine semantische Analyse zum Einsatz, die die Inhalte der Verträge ausliest und strukturiert zur Verfügung stellt. Dabei spielt die Extraktion sämtliche relevante Vertragsinformationen, wie Kundendaten, Vertragslaufzeiten und Versicherungsinhalte eine Rolle.

Zusätzlich werden die Daten validiert, um Inkonsistenzen und Fehler zu erkennen und nach Möglichkeit automatisch zu korrigieren oder sonst dem Vertriebsmitarbeiter Zurückzuspielen, um das Dokument neu zu erfassen oder die Informationen manuell richtig zustellen.

4.1.3 Integration und Speicherung (4 – 5a in Abbildung 2)

Die Daten werden nach ihrer Optimierung in das DWH-System integriert sowie in die CRM-Plattform. Dabei erfolgt eine automatisierte Zuordnung der Daten zu bestehenden Kundenprofilen.

Neben der Speicherung der Datensätze im CRM wird die digitalisierte Vertragsdatei von dort in ein zentrales Archivierungssystem ausgelagert, das sicherstellt, dass alle Vertragsdokumente jederzeit verfügbar und revisionssicher gespeichert sind. Dies ermöglicht eine langfristige und systematische Archivierung der Verträge.

4.1.4 Data-Analytics (5b in Abbildung 2)

Ein entscheidender Schritt im Soll-Prozess ist die Anbindung der Vertragsdaten an Data-Analytics-Systeme, welche über das Data Warehouse Zugriff auf die Vertragsdaten haben. Durch die Integration mit Data-Analytics-Systemen können Vertragsdaten in Kombination mit anderen Kundendaten für Marktprognosen und Vorhersagemodelle genutzt werden.

4.1.5 Automatisierte Verkaufschancen-Generierung (6. – 9. In Abbildung 2)

Ein weiteres Ziel des Soll-Prozesses ist die automatisierte Generierung von Verkaufschancen auf Basis der verarbeiteten Vertragsdaten. Hierbei werden spezifische Vertragsmerkmale analysiert und mit vorhandenen Kundendaten abgeglichen, um Upselling- oder Cross-Selling-Möglichkeiten zu identifizieren. Dieser Schritt soll dazu beitragen, den Vertrieb effizienter zu gestalten und zusätzliche Umsatzpotenziale zu erschließen.

Mit der detaillierten Erarbeitung des gesamten Soll-Prozesses ist ein ganzheitlicher Ansatz für die Optimierung der Fremdvertragserfassung geschaffen worden. Die beschriebenen Prozessschritte legen den Grundstein für eine automatisierte, effiziente und datengetriebene Verarbeitung von Fremdverträgen bei der Helvetia.

Um diese Prozessoptimierungen in die Praxis umzusetzen, ist eine entsprechende technische Architektur erforderlich, die die definierten Anforderungen erfüllt und eine nahtlose Integration in die bestehende Systemlandschaft ermöglicht. Im folgenden Kapitel wird die technische Architektursicht näher beleuchtet.

4.2 Technische Architektursicht

Die technische Architektur des Soll-Prozesses zielt darauf ab, eine effiziente, skalierbare und flexible Lösung zu implementieren, welche die Erfassung, Verarbeitung und Integration von Fremdverträgen wie in Kapitel 4.1 beschrieben optimiert. Um dies zu gewährleisten, werden nun zuerst die technischen (sowohl funktionalen als auch die nicht-funktionalen) Anforderungen des Systems präzise definiert und passende technologische Komponenten ausgewählt.

4.2.1 Technische Anforderungen

Im Rahmen der Definition des Soll-Prozesses ergibt sich die Notwendigkeit einer tragfähigen technischen Grundlage, die die effiziente Umsetzung der beschriebenen Prozessverbesserungen ermöglicht. Die Auswahl der Technologien basiert auf den funktionalen und nicht-funktionalen Anforderungen, die durch die Prozessautomatisierung, Datenverarbeitung und Integration innerhalb des Helvetia-IT-Ökosystems gestellt werden. Die funktionalen Anforderungen (Tabelle 1) ergeben sich aus den in Kapitel 4.1 dargestellten Optimierungsansätzen.

Anforderung	Beschreibung
Automatisierte Erfassung und Verarbeitung	Die Digitalisierung und automatische Verarbeitung von Fremdverträgen mittels OCR und semantischer Analyse zur Texterkennung und Datenextraktion.
Asynchrone Ereignisverarbeitung	Ereignisse müssen in verarbeitet werden, ohne dass direkte Abhängigkeiten zwischen den Komponenten bestehen.
Datenintegration und -speicherung	Integration der extrahierten Vertragsdaten in das CRM sowie weitere Systeme in der IT-Landschaft. Die Ablage der digitalisierten Dokumente erfolgt im Dokumentenarchivierungssystem
Automatisierte Offert-/Verkaufschancen-Generierung	Automatische Erstellung von Verkaufschancen im CRM auf Basis der extrahierten Vertragsdaten zur Identifizierung von Cross-Selling- und Upselling-Potenzialen.

Tabelle 1: Funktionale Anforderungen der Fremdvertragserfassung

Neben den funktionalen Anforderungen ergeben sich nicht-funktionale Anforderungen (Tabelle 2), welche sich an den Qualitätsanforderungen der ISO/IEC 25010, die Kriterien wie Funktionalität, Zuverlässigkeit, Benutzbarkeit und Sicherheit an Software definiert (International Organization for Standardization, 2014) orientieren.

Anforderung	Beschreibung
Skalierbarkeit	Die Architektur muss in der Lage sein, steigende Lasten zu bewältigen. Kafka ermöglicht horizontale Skalierbarkeit durch Hinzufügen weiterer Knoten im Cluster.
Hohe Verfügbarkeit und Fehlertoleranz	Die Plattform muss ausfallsicher und fehlertolerant sein. Kafka bietet automatische Replikation und Wiederherstellung im Fehlerfall.
Konsistenz und Datenintegrität	Sicherstellung, dass Vertragsdaten korrekt und konsistent im CRM gespeichert und zur Verfügung gestellt werden. Fehler bei der Datenverarbeitung müssen automatisch erkannt und selbst oder vom Nutzer korrigiert werden.

Anforderung	Beschreibung
Flexibilität und Erweiterbarkeit	Die Architektur sollte flexibel und erweiterbar sein, um zukünftige Funktionalitäten integrieren zu können, ohne bestehende Systeme signifikant anzupassen.
Sicherheit und Datenschutz	Die sensiblen Kundendaten müssen DSGVO-konform verarbeitet und gespeichert werden, inklusive verschlüsselter Übertragung und sicherem Datenzugriff.

Tabelle 2: Nicht-Funktionale Anforderungen der Fremdvertragsfassung

4.2.2 Beteiligte Systeme und Komponenten

Um den beschriebenen Anforderungen gerecht zu werden, wird die Integration mehrerer Technologien und Systeme vorgeschlagen, die als Teil der gesamten Unternehmensarchitektur die Umsetzung der geschäftlichen und technischen Prozesse unterstützen.

Das **CRM-System "uno"** nimmt hierbei eine zentrale Rolle ein und ist mit Salesforce realisiert. Es fungiert als Hauptsystem für die Erfassung und Verwaltung von Fremdverträgen, indem es die digitalen Kopien der Verträge mit einer Referenzierung auf den jeweiligen Kunden und den Dokumenttyp "Fremdvertrag" speichert. Innerhalb der Helvetia ist das CRM-System der primäre Verwalter des Geschäftsobjekts "Fremdvertrag", wodurch es die Integrität und Verfügbarkeit der relevanten Informationen gewährleistet.

Zur Verarbeitung der erfassten Fremdverträge wird ein neuer **Dokumentenverarbeitungsservice "DPS"** (Document Processing Service) bereitgestellt. Dieser neue Microservice nutzt fortschrittliche Technologien wie OCR (Optical Character Recognition) zur automatisierten Texterkennung sowie eine semantische Analyse, um die extrahierten Informationen zu strukturieren. Ein wesentlicher Bestandteil dieses Services ist die Validierung der Daten auf Vollständigkeit und Konsistenz. Sollte es zu Inkonsistenzen oder fehlenden Informationen kommen, wird der Endanwender benachrichtigt, um die erforderlichen Korrekturen vorzunehmen. Diese automatisierte Datenverarbeitung reduziert den manuellen Aufwand und minimiert gleichzeitig potenzielle Fehlerquellen.

Die Bereitstellung und der Betrieb der Microservices erfolgt über die **Helvetia Container Plattform (HCP)**, die auf einer self-hosted OpenShift-Umgebung basiert. HCP stellt die zentrale Infrastruktur für das Management von Microservices, Applikationen

und Integrationen dar und bietet somit die nötige Flexibilität und Skalierbarkeit zur effizienten Verwaltung der technischen Komponenten.

Die digitalisierten Fremdverträge werden im **Dokumentenarchivierungssystem "Doxis"** gesichert. Nach der Erfassung im CRM-System und Aufbereitung durch den Dokumentenverarbeitungs-Service werden die entsprechenden Dokumente zur dauerhaften Archivierung an Doxis übergeben, wo sie bei Bedarf abgerufen werden können. Dies stellt sicher, dass alle relevanten Dokumente unter Einhaltung der Compliance-Anforderungen gespeichert werden.

Für die langfristige Speicherung und Konsolidierung der gesammelten Daten wird das **Data-Warehouse (DWH)** genutzt. Dieses System stellt eine zentrale Plattform dar, die Daten aus verschiedenen Quellen innerhalb der Systemlandschaft konsolidiert und speichert. Es ist auf Basis von Microsoft Azure Services aufgebaut und erlaubt nicht nur die langfristige Speicherung von Kundendaten und Referenzobjekten, sondern auch die Möglichkeit, Daten über längere Zeiträume hinweg für analytische Zwecke auszuwerten.

Zur Unterstützung analytischer Prozesse kommt die **Analytics-Plattform "DARTS"** zum Einsatz. Diese ermöglicht mit MS PowerBI die Durchführung multidimensionaler Analysen auf Basis der im DWH gespeicherten Daten. Mithilfe von DARTS können fundierte Einblicke in komplexe Geschäftsprozesse gewonnen und datenbasierte Entscheidungen getroffen werden, was einen wesentlichen Beitrag zur Optimierung der Unternehmensprozesse leistet.

Schließlich spielt das **Offert-System "Helix"** eine Rolle innerhalb der Systemlandschaft. Helix bietet die Funktionalitäten zur Erstellung, Berechnung und Verwaltung von Offerten und ist nahtlos in die bestehende Infrastruktur integriert. Dies ermöglicht eine effiziente und konsistente Angebotserstellung auf Basis der im CRM und DWH vorliegenden Daten.

Im folgenden Kapitel werden die Kommunikationswege zwischen den genannten Systemkomponenten sowie die zugrunde liegende Architektur detailliert erläutert.

4.2.3 Event Verwaltung & Prozess Choreografie

Die vorangegangene Darstellung der beteiligten Systemkomponenten in Kapitel 4.2.2 verdeutlicht die Komplexität der Interaktionen sowie die engen Abhängigkeiten zwischen den Systemen, die innerhalb der Prozesskette zur Erfassung, Verarbeitung und Archivierung von Fremdverträgen bestehen.

Konzeption und Technischer Entwurf des Soll-Prozesses "Fremdvertrag Erfassung"

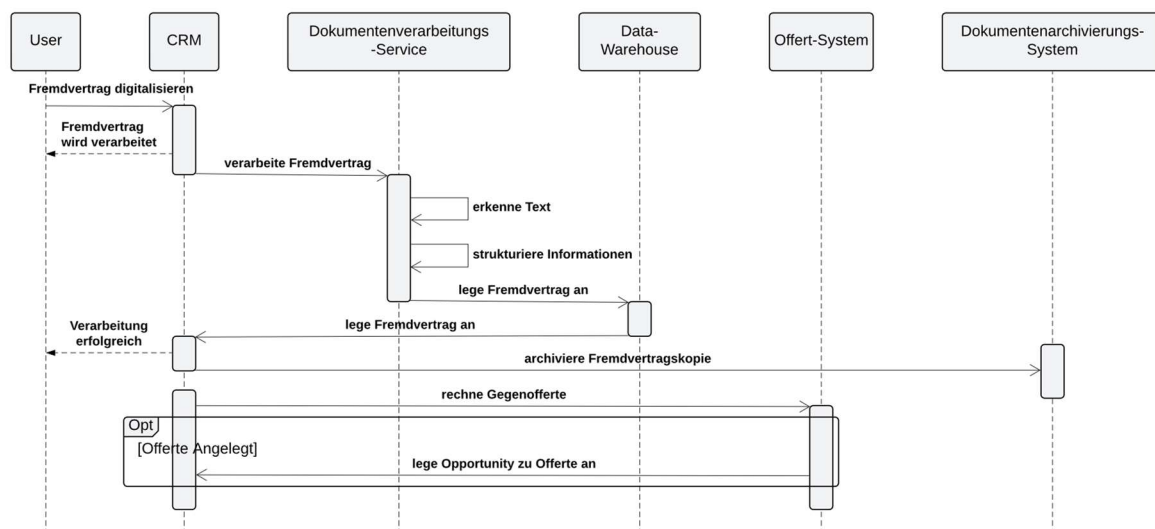


Abbildung 3: Nachrichten im Verteilten System

Dies wird insbesondere Abbildung 3 ersichtlich, welche die Kommunikation zwischen den Systemkomponenten visualisiert. Um Abhängigkeiten zu verringern und die Systemlandschaft flexibler und skalierbarer zu gestalten, wird eine Event-Driven Architecture (EDA) implementiert.

Die Einführung der Event-Driven Architecture adressiert diese Herausforderungen, indem die Kommunikation zwischen den Systemkomponenten auf der Grundlage von Ereignissen (Events) organisiert wird. Anstelle einer direkten, synchronen Interaktion erfolgt die Kommunikation asynchron über Events. Ein zentraler Bestandteil dieser Architektur ist der Einsatz von Apache Kafka als Messaging- und Event-Streaming-Plattform. Kafka fungiert als Event-Broker, der es ermöglicht, Ereignisse in Echtzeit zu verarbeiten und zu verteilen. Die relevanten Systemkomponenten können ihre Ereignisse in Kafka veröffentlichen und abonnieren, ohne direkt miteinander interagieren zu müssen. Dies reduziert die Kopplung der Systeme erheblich und erlaubt eine flexible Erweiterung der Architektur, ohne dass bestehende Prozesse beeinträchtigt werden müssen.

Darüber hinaus bietet Kafka die zentrale Umsetzung wichtiger nicht-funktionaler Anforderungen (vgl. Kapitel 4.2.1), die für die gesamte Systemarchitektur von entscheidender Bedeutung sind. Diese umfassen unter anderem Zuverlässigkeit, Fehlertoleranz und Skalierbarkeit. Durch die persistente Speicherung der Events stellt Kafka sicher, dass keine Daten verloren gehen, auch wenn es zu temporären Ausfällen

einzelner Komponenten kommt. Zudem unterstützt Kafka eine verteilte Verarbeitung von Ereignissen, was eine hohe Verfügbarkeit und Performance der Gesamtarchitektur gewährleistet.

Ein wesentlicher Vorteil der Event-Driven Architecture ist die Möglichkeit, eine Prozess-Choreographie zu implementieren, bei der es keinen zentralen Orchestrator gibt, der als potenzieller Single Point of Failure fungieren könnte. Stattdessen reagieren die Systeme eigenständig auf die relevanten Events, was die Ausfallsicherheit des Gesamtsystems erhöht.

Die technische Infrastruktur zur Bereitstellung und zum Betrieb von Kafka wird in die Helvetia Container Plattform (HCP) ausgelagert, wodurch der administrative Aufwand minimiert wird. Die HCP gewährleistet eine flexible und skalierbare Bereitstellung der Event-Streaming-Plattform und übernimmt die Verantwortung für den reibungslosen Betrieb.

Insgesamt stellt die Event-Driven Architecture einen essenziellen Schritt in Richtung einer resilienteren und zukunftssicheren Systemlandschaft dar, die auf die dynamischen Anforderungen des Unternehmens flexibel reagieren kann.

4.2.4 Kafka Topics

Die Implementierung einer Event-Driven Architecture erfordert die klare Definition von Kafka Topics, über die die Systemkomponenten asynchron miteinander kommunizieren können. Jedes Topic stellt einen spezifischen Ereignistyp dar, der von einer Komponente (Producer) erzeugt und von einer oder mehreren anderen Komponenten (Consumer) konsumiert wird. Diese Architektur erlaubt es, die Interaktionen zwischen den Systemen zu entkoppeln, wodurch eine flexible Skalierung und eine erhöhte Ausfallsicherheit gewährleistet werden.

In Tabelle 3 sind die für den Prozess der Fremdvertragserfassung relevanten Kafka Topics aufgelistet. Die Topics wurden nach einem Namensschema strukturiert, das die publizierende Anwendung, den Service und den jeweiligen Zweck widerspiegelt.

Kafka Topic Name	Zweck	Producer	Consumer
ch.uno.fremdvertrag.digitalized	Ein Fremdvertrag wurde digital erfasst.	uno	DPS
ch.dps.fremdvertrag.processed	Von einem digitalisierten Fremdvertrag wurden die Attribute strukturiert.	DPS	DWH

Kafka Topic Name	Zweck	Producer	Consumer
ch.dwh.fremdvertrag.created	Fremdvertrag wurde erfolgreich angelegt	DWH	uno, Darts
ch.dps.fremdvertrag.processing-error	Bei der Analyse des digitalisierten Fremdvertrags sind Fehler aufgetreten, welche manuell korrigiert werden müssen	DPS	uno
ch.helix.offer.created	Eine Offerte wurde erstellt	Helix	uno, DWH

Tabelle 3: Fremdvertragserfassung – Beteiligte Kafka-Topics

Ein zentrales Kafka Topic ist **ch.uno.fremdvertrag.digitalized**, das vom CRM-System (uno) erzeugt wird, sobald ein Fremdvertrag digital erfasst wurde. Dieses Event wird vom Dokumentenverarbeitungs-Service (DPS) konsumiert, der anschließend die Texterkennung und die Strukturierung der Vertragsdaten durchführt. Nach Abschluss dieser Verarbeitung wird das Ergebnis in Form des Events **ch.dps.fremdvertrag.processed** veröffentlicht, das wiederum vom Data-Warehouse (DWH) konsumiert wird. Hierdurch wird sichergestellt, dass die verarbeiteten und strukturierten Informationen langfristig gespeichert und für analytische Zwecke genutzt werden können.

Eine weitere wichtige Rolle spielt das Topic **ch.dwh.fremdvertrag.created**, das im DWH generiert wird, sobald ein Fremdvertrag erfolgreich angelegt wurde. Dieses Event wird sowohl vom CRM-System (uno) konsumiert, um die Referenz auf den verarbeiteten Fremdvertrag zu aktualisieren, als auch von DARTS, der Analytics-Plattform, um tiefere Analysen zu ermöglichen.

Für den Fall, dass bei der Verarbeitung eines Fremdvertrags im Dokumentenverarbeitungs-Service Fehler auftreten, wird das Topic **ch.dps.fremdvertrag.processing-error** erzeugt. Das CRM-System konsumiert dieses Event, um den Anwender über den Fehler zu informieren und gegebenenfalls manuelle Korrekturmaßnahmen einzuleiten. Zuletzt ist das Topic **ch.helix.offer.created** für die Erstellung einer Offerte von zentraler Bedeutung. Dieses Topic wird vom Offert-System (Helix) publiziert, wenn eine neue Offerte erstellt wird. Sowohl das CRM-System als auch das Data-Warehouse konsumieren dieses Event, um die Offerte im Kundenprofil zu referenzieren und die entsprechenden Daten langfristig für analytische Zwecke zu speichern.

4.3 Weiteres Projektvorgehen

Die erfolgreiche Umsetzung des Soll-Prozesses "Fremdvertragsfassung" erfordert eine schrittweise Implementierung, die sich auf priorisierte Komponenten konzentriert. Um einen schnellen Mehrwert zu erzielen und erste praktische Erfahrungen zu sammeln, wird ein Minimum Viable Product (MVP) definiert. Das MVP umfasst die essenziellen Funktionen, die notwendig sind, um den neuen Prozess in seiner Grundform operativ einzusetzen, eine Bausteinsicht dessen ist in Abbildung 4 dargestellt. Durch die Fokussierung auf Kernkomponenten können Risiken minimiert und der Entwicklungsaufwand überschaubar gehalten werden.

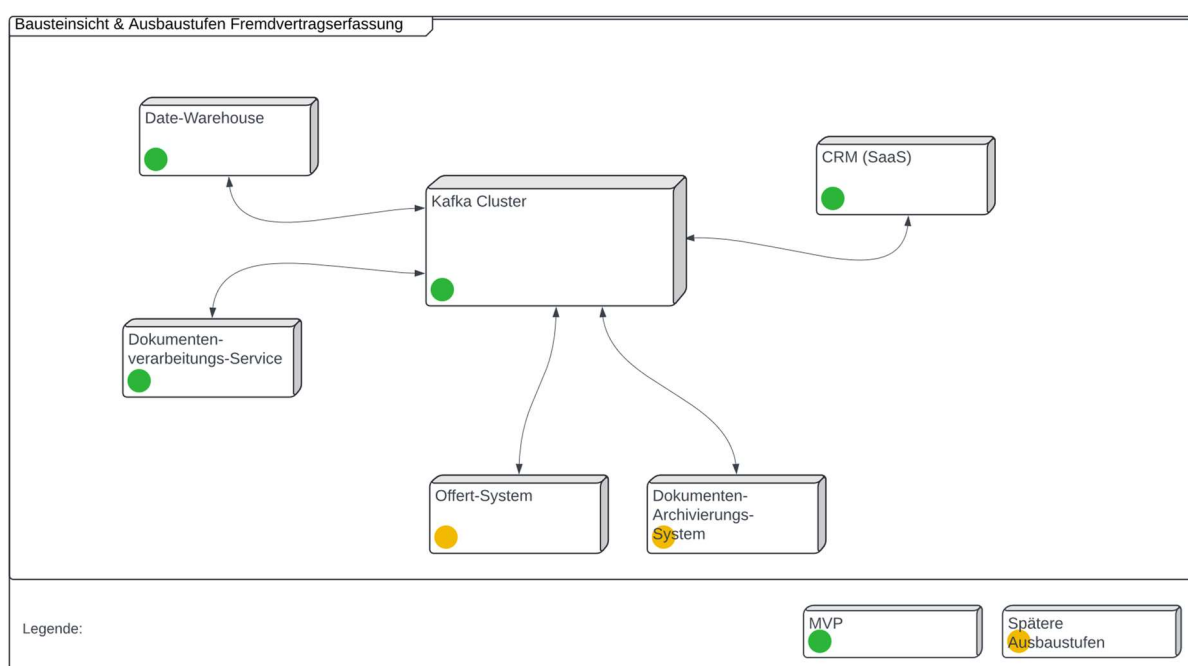


Abbildung 4: Bausteinsicht & Ausbaustufen Fremdvertragserfassung

Die Auswahl der Komponenten für das MVP basiert auf folgenden Überlegungen: Die Implementierung des Dokumentenverarbeitungsservices (DPS) sowie der Nutzeroberflächen im CRM stellt die grundlegende Funktionalität zur Verfügung und ist somit eine zwingende Maßnahme im MVP. Darüber hinaus ist auch die Integration des Data Warehouse (DWH) und die Einrichtung der Event-Driven Architecture (EDA) mit Apache Kafka notwendig für die Integration der Systeme. Durch die Konzentration auf die essenziellen Bausteine können potenzielle Risiken frühzeitig identifiziert und behoben werden, wodurch Fehlentwicklungen aufgrund unbekannter Problemvariablen vermieden werden. Funktionen, die keinen kritischen Mehrwert liefern oder ein höheres Risiko mit sich bringen, sollen erst später erfolgen. Gleichzeitig wird sichergestellt, dass

die benannten Funktionen den größten Mehrwert bei minimalem Aufwand liefern. Dies verringert die Komplexität und begünstigt einen erfolgreichen Teilabschluss, wodurch ein betriebswirtschaftlich sinnvoller Einsatz der Unternehmensressourcen sichergestellt wird.

Nach dem erfolgreichen Abschluss der Implementierungsphase des Geschäftsprozess-Lebenszyklus des MVP folgt die Ausführungsphase, in der der neue Prozess produktiv eingesetzt wird. In dieser Phase stehen verschiedene Aktivitäten im Vordergrund: Überwachung und Stabilisierung durch Monitoring-Tools zur Sicherstellung der Systemleistung und Verfügbarkeit sowie ein effektives Incident Management zur schnellen Behebung von Problemen. Gleichzeitig können Rückmeldungen der Anwender gesammelt werden, um möglichen Nachschulungsbedarf oder Anpassungen des neuen Prozesses zu erkennen. Dies ermöglicht die frühzeitige Identifikation von Optimierungspotenzialen.

Basierend auf den Erkenntnissen aus der Ausführungsphase werden in der kontinuierlichen Verbesserungsphase weitere Funktionen und Systeme integriert, wie die Anbindung des Offert-Systems "Helix" und die Integration des Dokumentenarchivierungssystems "Doxis".

Die Definition und Implementierung eines MVP legt den Grundstein für eine erfolgreiche Optimierung der Fremdvertragserfassung. Durch die fokussierte Auswahl der Komponenten werden die dringendsten Probleme adressiert und ein schneller Mehrwert erzielt. Die anschließende Ausführungsphase dient der Stabilisierung des neuen Prozesses, während die kontinuierliche Verbesserung weitere Ausbaustufen und eine flexible Anpassung an zukünftige Anforderungen ermöglicht.

5 Fazit

Im Rahmen dieser Arbeit wurde der bestehende Prozess der Fremdvertragserfassung bei der Helvetia analysiert und Optimierungspotenziale identifiziert. Durch eine detaillierte Untersuchung der aktuellen Abläufe konnten Schwachstellen aufgedeckt werden, insbesondere in den Bereichen der manuellen Datenerfassung, der ineffizienten Datenverarbeitung und der mangelnden Integration in die bestehende Systemlandschaft.

Auf Basis dieser Erkenntnisse wurde ein Soll-Prozess konzipiert, der auf Automatisierung und moderne Technologien setzt. Der entworfene Prozess sieht die Einführung eines Dokumentenverarbeitungs-Micro Services vor, der eine automatisierte Erfassung und Verarbeitung von Fremdverträgen ermöglicht. Die Integration in das bestehende CRM-System und das Data Warehouse gewährleistet eine konsistente Datenhaltung und bildet die Grundlage für erweiterte Datenanalysen.

Die technische Architektur wurde sorgfältig geplant, um den Anforderungen an Flexibilität, Skalierbarkeit und Zuverlässigkeit gerecht zu werden. Durch den Einsatz einer Event-Driven Architecture (EDA) und die Nutzung von Apache Kafka als Messaging-Plattform wird eine entkoppelte Kommunikation zwischen den Systemen ermöglicht. Dies fördert nicht nur die Performance, sondern auch die Erweiterbarkeit der Gesamtarchitektur.

Das weitere Vorgehen wurde in Form einer Roadmap skizziert, die die schrittweise Umsetzung der geplanten Maßnahmen vorsieht. Dabei wurde ein Minimal Viable Product (MVP) definiert, das die Kernfunktionen des neuen Prozesses umfasst. Die Implementierung des MVP ist jedoch zum Zeitpunkt dieser Arbeit noch nicht erfolgt und stellt einen Ausblick auf die nächste Projektphase dar.

Die Umsetzung des konzipierten Soll-Prozesses und die Implementierung des MVP bilden die nächsten Schritte im Projektverlauf. Die erfolgreiche Realisierung wird es der Helvetia ermöglichen, die Fremdvertragserfassung signifikant zu verbessern und die Effizienz der Geschäftsprozesse zu steigern. Insbesondere die Automatisierung zentraler Prozessschritte und die bessere Integration in die bestehende Systemlandschaft versprechen erhebliche Vorteile.

In der Ausführungsphase wird der Fokus auf der Implementierung der priorisierten Bausteine liegen. Durch die kontinuierliche Einbindung von Benutzerfeedback und die fortlaufende Optimierung der Systeme soll eine hohe Akzeptanz bei den Anwendern

Fazit

erreicht werden. Die geplanten Ausbaustufen, wie die Integration des Offert-Systems Helix und des Dokumentenarchivierungssystems Doxis, werden den Prozess weiter verbessern und zusätzliche Mehrwerte schaffen.

Langfristig bietet die entwickelte Lösung die Möglichkeit, auch weitere Prozesse und Geschäftsbereiche zu optimieren. Die flexible und skalierbare Architektur erlaubt es, auf zukünftige Anforderungen und technologische Entwicklungen zu reagieren. Damit soll erreicht werden, dass Helvetia sich erfolgreich den Herausforderungen der digitalen Transformation zu stellen und ihre Wettbewerbsfähigkeit nachhaltig zu stärken kann.

Die Arbeit hat gezeigt, dass eine ganzheitliche Betrachtung von Geschäftsprozessen und der dazugehörigen technischen Architekturen essenziell ist, um nachhaltige Verbesserungen zu erzielen. Durch die Kombination von Prozessanalyse, technischer Konzeption und strategischer Planung können erhebliche Effizienzsteigerungen erreicht werden.

Obwohl die Implementierung des MVP noch aussteht, bildet die vorliegende Konzeption eine solide Grundlage für die praktische Umsetzung.

Literaturverzeichnis

Apache Kafka Documentation. (o. J.). [Dokument]. Kafka 3.8 Documentation. Abgerufen 14. September 2024, von <https://kafka.apache.org/documentation/>

Davis, F. D. (1989). Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Quarterly*, 13(3), 319. <https://doi.org/10.2307/249008>

Ford, N., Richards, M., Sadalage, P., & Dehghani, Z. (2021). *Software Architecture: The Hard Parts : modern trade-off analyses for distributed architectures* (First edition). O'Reilly Media. <https://learning.oreilly.com/library/view/software-architecture-the/9781492086888>

Fowler, M. (2013). *Patterns of enterprise application architecture* (Nineteenth printing). Addison-Wesley.

Freund, J., & Rücker, B. (2019). *Praxishandbuch BPMN: Mit Einführung in DMN* (6., aktualisierte Auflage). Hanser.

International Organization for Standardization. (2014). *System und Software-Engineering—Qualitätskriterien und Bewertung von System- und Softwareprodukten* (No. ISO/IEC 25010). <https://www.iso.org/standard/64764.html>

Kleppmann, M. (2019). *Datenintensive Anwendungen designen* (F. Langenau, Übers.; 1. Auflage). O'Reilly. <https://learning.oreilly.com/library/view/datenintensive-anwendungen-designen/9781492070153>, abgerufen am 25.09.2024

Kratzke, N. (2024). *Cloud-native Computing: Software Engineering von Diensten und Applikationen für die Cloud* (2., überarbeitete Auflage). Hanser.

- Müller, L., & Thienen, L. (2016). Digitalisierung der Vertriebsprozesse – Framework zur erfolgreichen Transformation. In L. Binckebanck (Hrsg.), *Digitalisierung im Vertrieb: Strategien zum Einsatz neuer Technologien in Vertriebsorganisationen*. Springer Gabler.
- Newman, S. (2021). *Building microservices: Designing fine-grained systems* (Second edition).
- Richards, M. (2015). *Software architecture patterns*. O'Reilly Media.
- Weske, M. (2012). *Business process management: Concepts, languages, architectures* (2. ed). Springer.
- Wolff, E. (2018). *Microservices: Grundlagen flexibler Softwarearchitekturen* (2., aktualisierte Auflage). dpunkt.verlag.
- Zelenin, A., & Kropp, A. (2022). *Apache Kafka: Von den Grundlagen bis zum Produktiveinsatz*. Hanser.

Abbildungen

Abbildung 1: Schematische Darstellung der Funktion von Apache Kafka

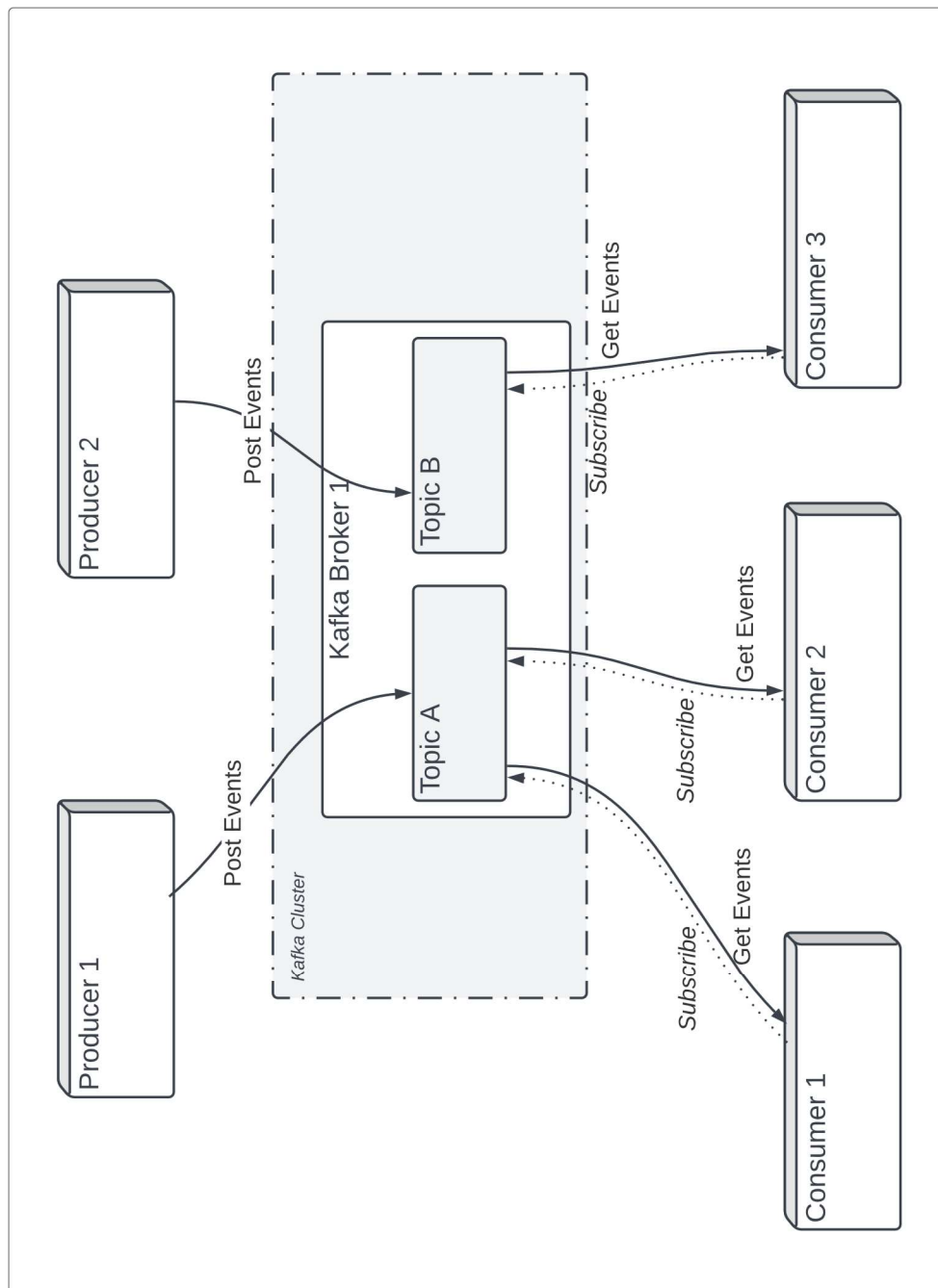


Abbildung 2: BPMN CHART "Fremdvertragserfassung" – Soll-Prozess

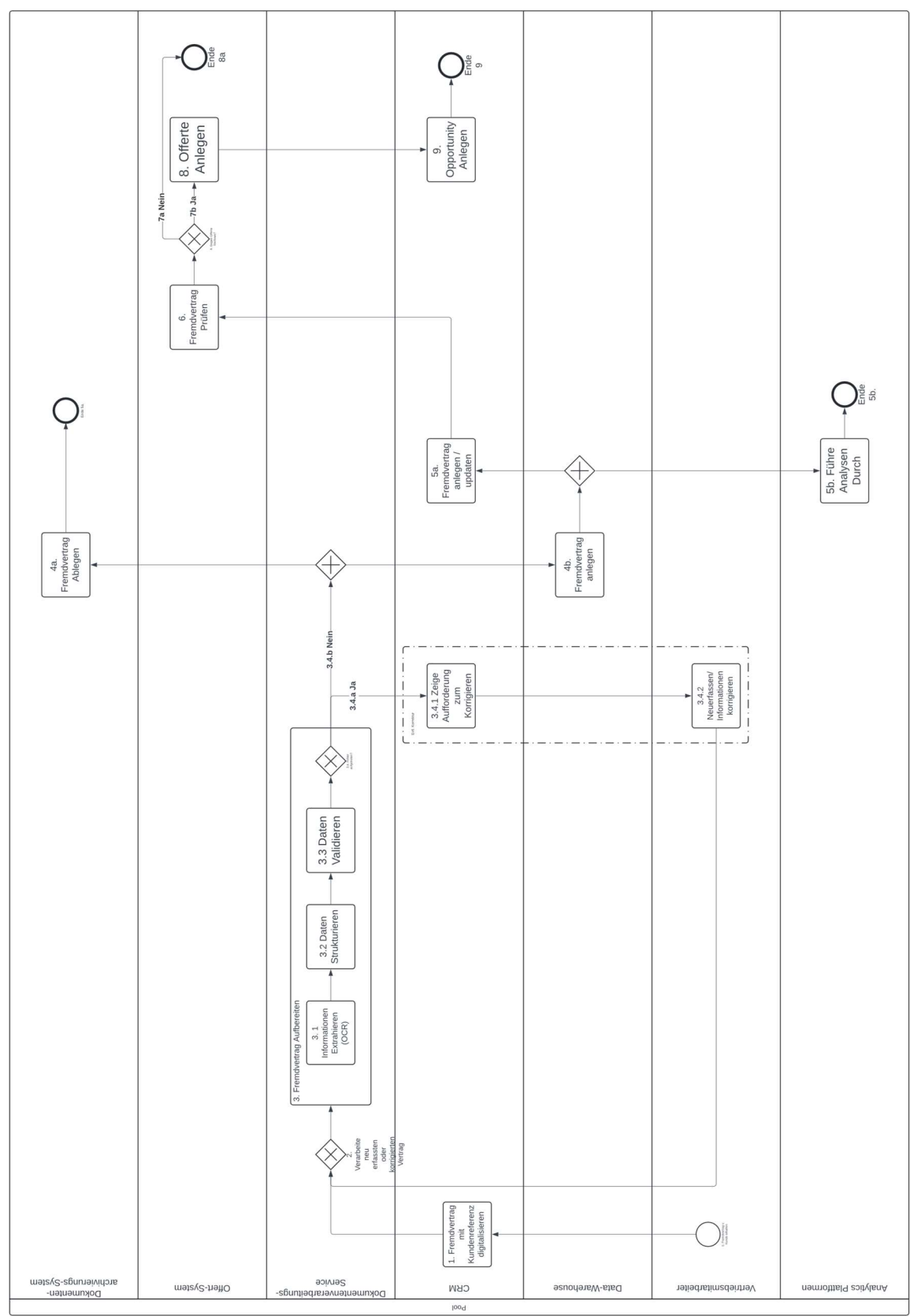


Abbildung 3: Nachrichten des verteilten Systems im Soll-Prozess

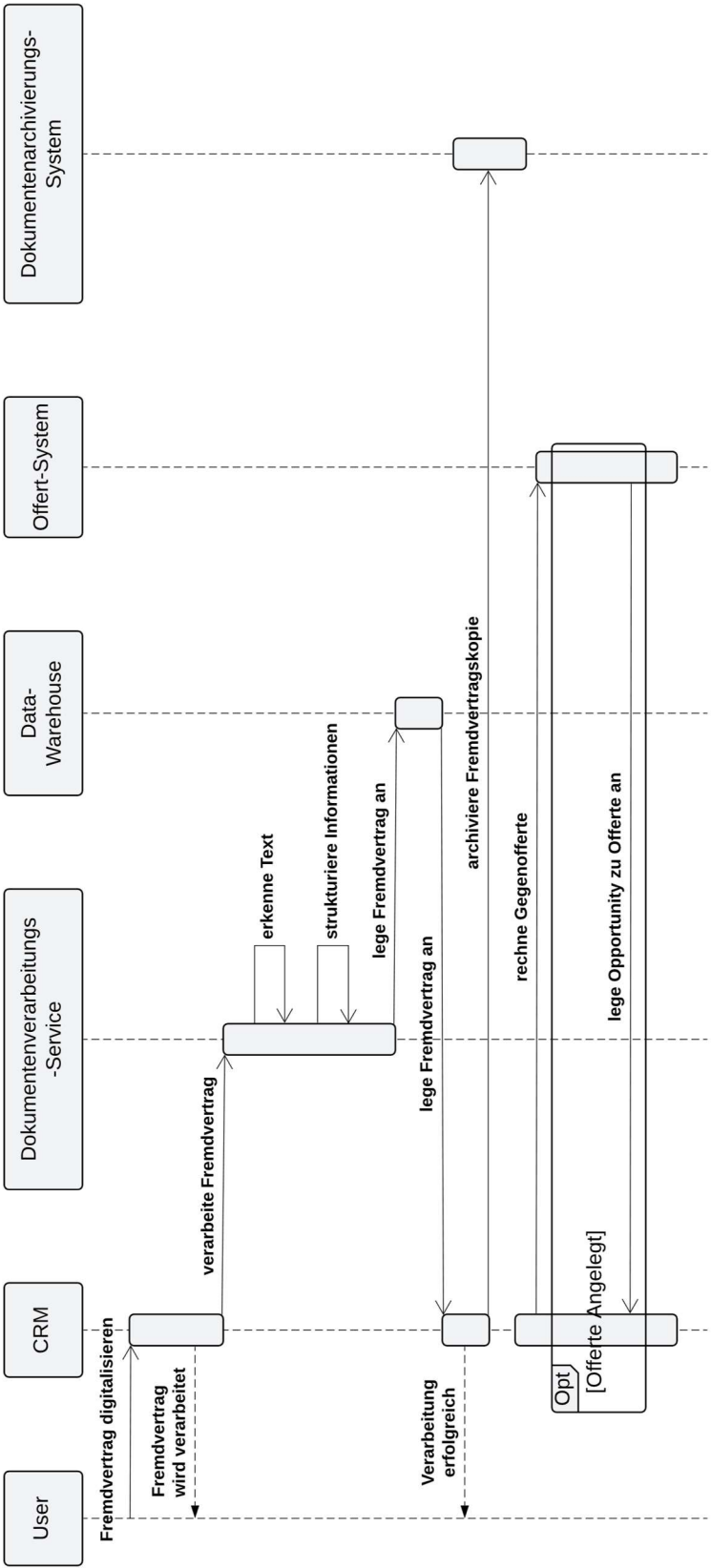


Abbildung 4: Bausteinsicht & Ausbaustufen Fremdvertragserfassung

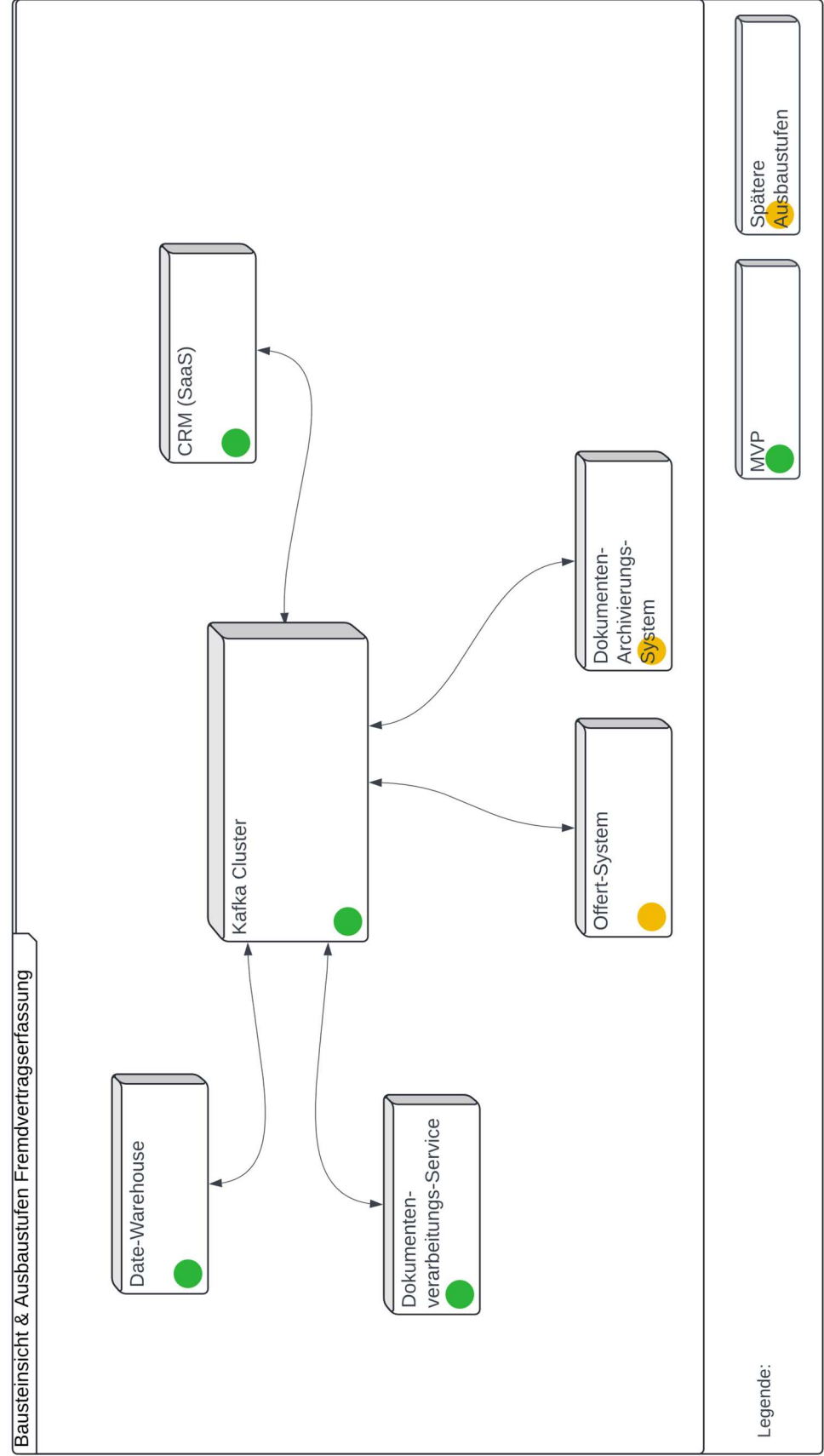
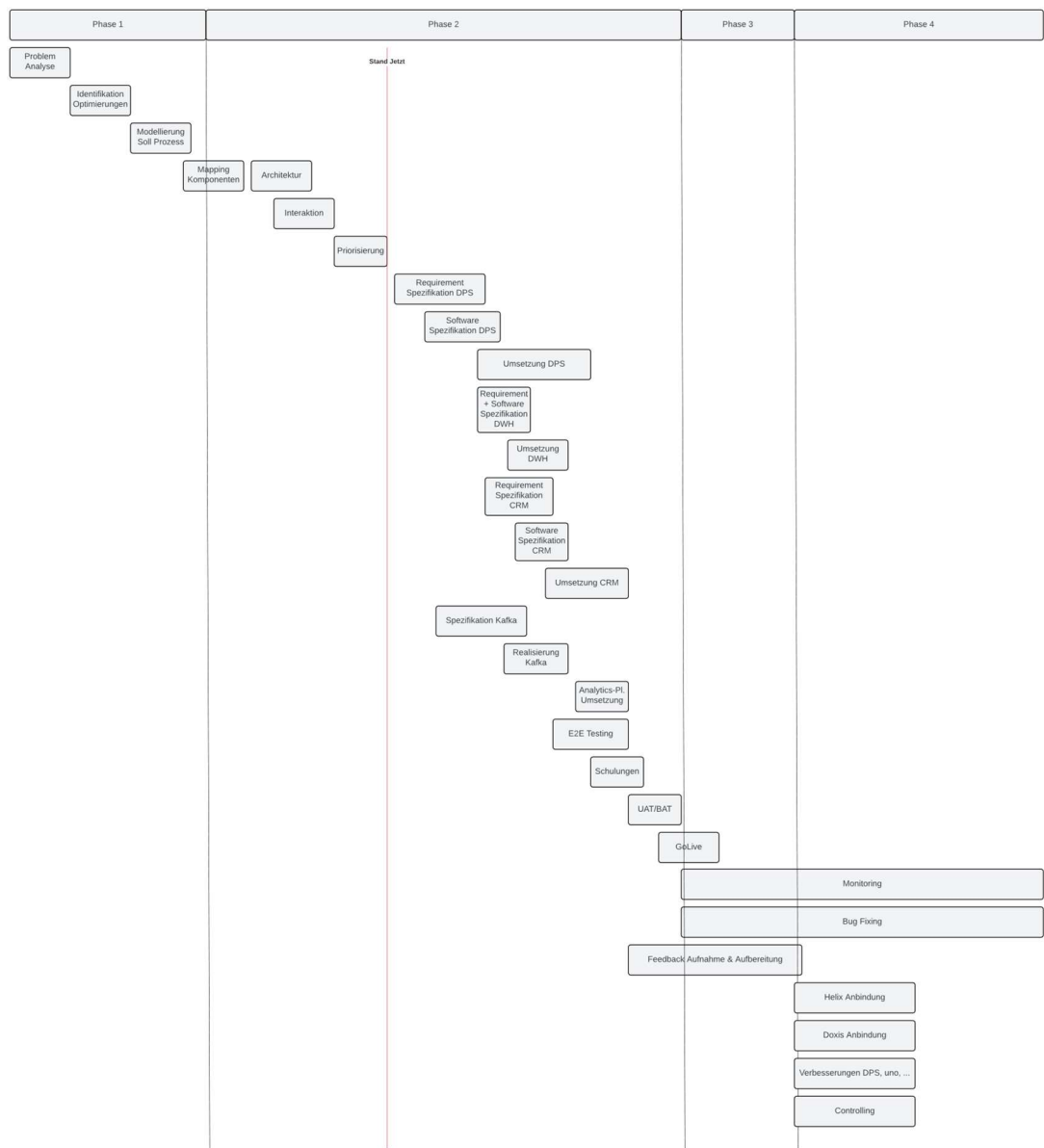


Abbildung 4: Projektablaufe exemplarisch als Gantt-Chart dargestellt



Anhang

Erklärung zur Verwendung von KI-Systemen

XXIII

Abbildungen.zip

Fehler! Textmarke nicht definiert.

Anhang X

Fehler! Textmarke nicht definiert.

Zusätzlich nur in der Digitalen Version enthalten:

Abb1 - Kafka Schema - PA2 Jacob Ruhnau.svg

Abb2 - BPNM Soll-Prozess Fremdvertragserfassung - PA2 Jacob Ruhnau.svg

Abb3 - Nachrichten im Verteilen System - PA2 Jacob Ruhnau.svg

Abb4- Bausteinsicht MVP- PA2 Jacob Ruhnau.svg

Abb5- Projektablauf - PA2 Jacob Ruhnau

Anhang2 - Kafka Topic Spezifikation - PA2 Jacob Ruhnau.json

Anhang 1: Spezifikation Kafka Topic *ch.dps.fremdvertrag.processed*

```
{
  "name": "data",
  "namespace": "ch.dps.fremdvertrag.processed",
  "type": "record",
  "doc": "https://wiki.helvetia.group/display/VTPL/API+Specification/ch+dps+fremdvertrag+processed",
  "fields": [
    {
      "name": "name",
      "type": "string",
      "doc": "Competitor Contract Name (Number) as given on the
Document"
    },
    {
      "name": "digitizationTimestamp",
      "type": {
        "type": "long",
        "logicalType": "timestamp-millis"
      },
      "doc": "The timestamp when the document was digitized"
    },
    {
      "name": "sourceSystem",
      "type": "string",
      "doc": "The source system from which the document was digit-
ized"
    },
    {
      "name": "digitilizedBy",
      "type": "string",
      "doc": "User-Helvetia-Identifrier who digitized the document"
    },
    {
      "name": "competitor",
      "type": "string",
      "doc": "Competitor Name"
    },
    {
      "name": "contractSignedDate",
      "type": "string",
      "doc": "Contract Signed Date in YYYY-MM-DD format"
    },
    {
      "name": "startDate",
      "type": "string",
    }
  ]
}
```

```

        "doc": "Contract Start Date in YYYY-MM-DD format"
    },
    {
        "name": "expirationDate",
        "type": "string",
        "doc": "Contract Expiration Date in YYYY-MM-DD format"
    },
    {
        "name": "contractHolder",
        "type": {
            "type": "record",
            "name": "ContractHolder",
            "namespace": "ch.dps.fremdvertrag.processed",
            "fields": [
                {
                    "name": "name",
                    "type": "string",
                    "doc": "Name of the Contract Holder"
                },
                {
                    "name": "zepasId",
                    "type": "string",
                    "doc": "Helvetia Business Partner Global Identifier",
                    "default": null
                },
                {
                    "name": "birthDate",
                    "type": "string",
                    "doc": "Birth Date of the Contract Holder",
                    "default": null
                },
                {
                    "name": "street",
                    "type": "string",
                    "doc": "Street of the Contract Holder as per Contract Document",
                    "default": null
                },
                {
                    "name": "houseNumber",
                    "type": "string",
                    "doc": "House Number of the Contract Holder as per Contract Document",
                    "default": null
                },
                {
                    "name": "zipCode",

```

```

        "type": "string",
        "doc": "Zip Code of the Contract Holder as per
Contract Document",
        "default": null
    },
    {
        "name": "city",
        "type": "string",
        "doc": "City of the Contract Holder as per Con-
tract Document",
        "default": null
    },
    {
        "name": "country",
        "type": "string",
        "doc": "Country of the Contract Holder as per
Contract Document",
        "default": null
    },
    {
        "name": "email",
        "type": "string",
        "doc": "Email of the Contract Holder as per Con-
tract Document",
        "default": null
    },
    {
        "name": "phone",
        "type": "string",
        "doc": "Phone of the Contract Holder as per Con-
tract Document",
        "default": null
    },
    {
        "name": "mobile",
        "type": "string",
        "doc": "Mobile of the Contract Holder as per Con-
tract Document",
        "default": null
    },
    {
        "name": "fax",
        "type": "string",
        "doc": "Fax of the Contract Holder as per Con-
tract Document",
        "default": null
    }
}
]

```

```

    },
    "doc": "Details of the Contract Holder"
  },
  {
    "name": "contractParticipants",
    "type": {
      "type": "array",
      "items": {
        "type": "record",
        "name": "Participant",
        "namespace": "ch.dps.fremdvertrag.processed",
        "fields": [
          {
            "name": "name",
            "type": "string",
            "doc": "Name of the Contract Participant",
            "default": null
          },
          {
            "name": "zepasId",
            "type": "string",
            "doc": "Helvetia Business Partner Global
Identifier",
            "default": null
          },
          {
            "name": "birthDate",
            "type": "string",
            "doc": "Birth Date of the Contract Partici-
pant",
            "default": null
          }
        ]
      }
    },
    "doc": "List of Contract Participants",
    "default": null
  },
  {
    "name": "policyType",
    "type": "string",
    "doc": "Type of the Policy",
    "default": null
  },
  {
    "name": "coverage",
    "type": "string",
    "doc": "Coverage details of the Policy",

```



```

        "default": null
    },
    {
        "name": "annualPremium",
        "type": "double",
        "doc": "Annual Premium Amount",
        "default": null
    },
    {
        "name": "currency",
        "type": "string",
        "doc": "Currency of the Premium Amount",
        "default": "CHF"
    },
    {
        "name": "status",
        "type": "string",
        "doc": "Status of the Contract",
        "default": "active"
    }
]
}

```

Anhang 2: Erklärung zur Verwendung von KI-Systemen

Erklärung zur Verwendung von KI-Systemen Ich erkläre, dass ich mich aktiv über die Leistungsfähigkeit und Beschränkungen der in meiner Arbeit eingesetzten KI-Systeme informiert habe; alle Inhalte aus wissenschaftlich anerkannten Quellen entnommen und entsprechend gekennzeichnet habe; alle Inhalte unter Anwendung wissenschaftlicher Methoden im Rahmen der vorliegenden Arbeit von mir selbst entwickelt wurden; mir bewusst bin, dass ich als Autor*in dieser Arbeit die Verantwortung für die in ihnen gemachten Angaben und Aussagen trage.

Bei der Erstellung der Arbeit habe ich die folgenden auf künstlicher Intelligenz (KI) basierten Systeme in der im Folgenden dargestellten Weise benutzt:

Arbeitsschritt	Eingesetzte(s) KI- System(e)	Beschreibung der Verwendungsweise
Generierung von Ideen und Konzeption der Arbeit	ChatGPT-4o	Brainstorming zu Möglichen Bearbeitungsthemen.
Literatursuche		
Literaturverwaltung und Zitationsmanagement		
Auswahl von Methoden und Modellen		
Datensammlung und analyse		
Generierung von Programmcodes		
Erstellung von Visualisierungen		
Interpretation und Validierung		
Strukturierung des Texts der Arbeit		
Formulierung des Texts der Arbeit	DeepL Write Pro ChatGPT 1o-Preview	Rechtschreibkontrolle und Sprachliche Umformulierungen
Übersetzung des Texts der Arbeit		

Arbeitsschritt	Eingesetzte(s) KI- System(e)	Beschreibung der Verwen- dungsweise
Redigieren des Texts		
Vorbereitung der Präsen- tation des Texts		
Sonstiges		

Grenzach-Wyhlen, 30. September 2024

Jacob Ruhnau