Smart Data Interoperability

edited by

Andreas Winter
Carl von Ossietzky University, Oldenburg

Christian Schönberg
Carl von Ossietzky University, Oldenburg

Oldenburg Lecture Notes on Software Engineering

OLNSE Number 6/2023
October 2023

Co-funded by
the European Union

Software Engineering

Carl von Ossietzky Universität Oldenburg

Smart Data Interoperability

edited by

Andreas Winter
Carl von Ossietzky University, Oldenburg

Christian Schönberg
Carl von Ossietzky University, Oldenburg

Oldenburg Lecture Notes on Software Engineering

OLNSE Number 6/2023
October 2023

Oldenburg Lecture Notes
on Software Engineering (OLNSE)
Carl von Ossietzky University Oldenburg
Department for Computer Science
Software Engineering Group
26111 Oldenburg, Germany

Content

Smart Data Interoperability

Im Interreg North Sea Projekt „Data for All“ (D4A)[1] werden innovative Daten-getriebene Ansätze zur Schaffung und Verbesserung von digitalen Diensten im Bereich von Smart Cities und Smart Regions erforscht und praktisch erprobt. Smarte Systeme – Smart Cities, Smart Health Applications, Smart Agriculture Systems, usw. – produzieren große Mengen an Daten. Diese Daten sind oft sehr unterschiedlich aufgebaut, selbst wenn sie ähnliche Informationen enthalten. Sie folgen oft keinem standardisierten Schema, oder interpretieren Standards auf unterschiedliche Weise. Sie verwenden unterschiedliche Formate, Formatierungen, Kodierungen, Maßeinheiten, Intervalle, Konventionen und Annahmen.

Beispielsweise kann ein Anbieter von E-Scootern seine Daten über die Gefährte nach deren Aufenthaltsort strukturieren, während ein anderer Anbieter die Daten nach dem genauen Typs des Scooters strukturiert. Dadurch ist eine gemeinsame Nutzung beider Datensätze deutlich erschwert. Aber nicht nur strukturell unterscheiden sich Daten Smarter Systeme, auch im Detail gibt es große Unterschiede: So speichert ein Scooter-Anbieter alle zwei Minuten den Aufenthaltsort der Geräte, ein anderer nur alle drei Minuten. Will man die Zeitreihen zusammenlegen, kommt es ggf. zu Inkonsistenzen. Auch unterschiedliche Maßeinheiten (mm/cm/in, usw.) oder andere technische Formate (JSON/XML/SQL/NoSQL) erschweren die gemeinsame interoperable Verwendung der Daten.

Im Seminar „Smart Data Interoperability“ wurde im Sommersemester 2023 im Rahmen von forschungsorientierter Lehre und im Kontext des D4A-Projekts zunächst die Domäne definiert: Was genau sind die Herausforderung bei der Interoperabilität von Daten aus Smarten Systemen? Und welche generellen Lösungsansätze existieren bisher?

Nach einer eigenständigen Recherche wurde die Domäne mithilfe von bewährten Kreativitätstechniken beschrieben. Als Herausforderungsbereiche wurden **Technische Herausforderungen**, **Datenkonflikte**, **Schemakonflikte** und **Rechtliche Herausforderungen** identifiziert. Drei dieser Teilaspekte der Domäne wurden anschließend von einzelnen Studierenden detailliert betrachtet und individuell aufgearbeitet. Die dort gewonnenen Erkenntnisse werden im Folgenden dokumentiert.

In the Interreg North Sea project “Data for All” (D4A)[1], innovative data-driven approaches to create and improve digital services in the field of Smart Cities and Smart Regions are researched and practically tested. Smart systems – smart cities, smart health applications, smart agriculture systems, etc. – produce large amounts of data. These data are often structured very differently, even if they contain similar information. They often do not follow a standardized schema, or they interpret standards in different ways. They use different formats, formatting, coding, units of measurement, intervals, conventions, and assumptions.

For example, one e-scooter provider may structure its vehicle data according to their location, while another provider may structure the data according to the exact type of scooter. This makes it much more difficult to use both sets of data together. But it is not only structurally that data of smart systems differ, there are also major differences in detail: for example, one scooter provider stores the whereabouts of the vehicles every two minutes, another only every three minutes. If one wants to combine the time series, inconsistencies may occur. Different units of measurement (mm/cm/in, etc.) or different technical formats (JSON/XML/SQL/NoSQL) also make it difficult to use the data in an interoperable manner.

In the seminar “Smart Data Interoperability” during the summer semester 2023 as part of research-oriented teaching and in the context of the D4A project, initially the domain was defined: What exactly are the challenges in interoperability of data from smart systems? And what general solution approaches exist so far?

After independent research, the domain was described using proven creativity techniques. **Technical challenges**, **data conflicts**, **schema conflicts**, and **legal challenges** were identified as challenge areas. Three of these sub-aspects of the domain were then considered in detail by individual students and worked through individually. The insights gained there are documented below.

[1] https://www.interregnorthsea.eu/dataforall

Technische Herausforderungen bei der Dateninteroperabilität

1st Clausing Tim
Carl von Ossietzky Universität Oldenburg, Department für Informatik
Abteilung Softwaretechnik
Oldenburg, Deutschland
tim.clausing@uni-oldenburg.de

***Zusammenfassung*—In Smart Regions stellt das Verarbeiten und Bereitstellen von Daten einen zentralen Aspekt dar. Hierbei muss oft auf unterschiedliche Datenquellen zugegriffen werden. Somit ist die Dateninteroperabilität eine zentrale Herausforderung bei der Entwicklung smarter Anwendungen, wie z.B. im Data for All (D4A) Projekt. Dabei umfasst die Dateninteroperabilität ein breites Problemfeld. In dieser Arbeit wird gezielt auf die technischen Herausforderungen eingegangen. Wobei zentrale Probleme genannt, spezifiziert und Lösungsansätze präsentiert werden.**

***Index Terms*—Data interoperability, technische Herausforderungen**

I. EINLEITUNG

Ein Bereich, in dem die Dateninteroperabilität eine große Herausforderung darstellt, sind Smart Cities bzw. Smart Regions. So beschreibt die DIN in einem Impulspapier zu Smart Cities die Interoperabilität als wichtigen Aspekt in der digitalen Entwicklung von Städten und Kommunen und erkennt einen hohen Standardisierungsbedarf in diesem Bereich an [1].
Ein Projekt, das sich unter anderem mit der Dateninteroperabilität in Smart Regions beschäftigt, ist *Data for All (D4A)*. D4A hat das Ziel durch das Einrichten von lokalen Datenökosystemen die Nordseeregion als Vorreiter für datengesteuerte Innovation zu positionieren [2], und stellt das Umfeld dieser Arbeit dar. Dabei ist Dateninteroperabilität definiert als „(...) die Funktion von Informationssystemen, Daten auszutauschen und die Weitergabe von Informationen zu ermöglichen. “ [3].

II. PROBLEMBESCHREIBUNG

Das D4A-Projekt erfordert für das Schaffen der angestrebten Datenökosysteme eine umfangreiche Interoperabilität der Daten. Um dies zu erreichen, muss zunächst geklärt werden, welche Herausforderung es bei dieser Interoperabilität gibt. Deshalb wurden in einer ausführlichen Literaturrecherche, welche der vorliegenden Arbeit vorangegangen ist, vier zentrale Herausforderungen in diesem Bereich ermittelt. Hierbei handelt es sich um:

- Herausforderungen, die aus Datenfehlern resultieren
- Technische Herausforderungen
- Herausforderungen, die aus Datenschemata resultieren
- Rechtliche Herausforderungen

Da jedes einzelne dieser Themengebiete viele Probleme umfasst, fokussiert sich diese Arbeit auf den Aspekt der technischen Herausforderungen. Der Einsatz von Technologien aus dem Bereich des *Internet of Things* (IoT) ist ein treibender Faktor für Smart Cities [4, S.1] und somit auch für Smart Regions. Allerdings werden in diesem Umfeld viele weitere Technologien für das Sammeln und Übertragen von Daten eingesetzt. So spielen z.B. auch Datenquellen, die sich im Bereich von Big Data bewegen eine wichtige Rolle [5, S. 30]. Die unterschiedlichen Datenquellen, deren Interoperabilität, wie bereits erwähnt, ein wichtiges Ziel im D4A-Projekt sind, basieren also auf unterschiedlichsten Technologien. Hieraus ergeben sich einige Herausforderungen, die es zu lösen gilt. Das Ziel dieser Arbeit ist es, durch eine ausführliche Literaturanalyse, Lösungsansätze zu ausgewählten technischen Herausforderungen der Dateninteroperabilität zu identifizieren. Dabei werden technische Herausforderungen definiert als: Herausforderungen, die aus dem Hard- und Softwaredesign der Datenquelle(n) resultieren. Dies umfasst jedoch nicht die Modellierung der Daten oder ihren Inhalt. Somit ergibt sich die folgende Forschungsfrage:

Wie lassen sich ausgewählte technische Herausforderungen bei der Dateninteroperabilität lösen?

A. Anwendungsbeispiel

Zur Veranschaulichung der konkreten Probleme, die aus den technischen Herausforderungen der Dateninteroperabilität in Smart Regions resultieren, wird an dieser Stelle ein Anwendungsbeispiel eingeführt. Hierbei werden drei Anbieter (A, B, und C) von E-Rollern betrachtet. Eine Smart Region möchte nun eine App entwickeln, um die Mobilität in der Region zu verbessern. Hierfür muss die Interoperabilität zwischen den Systemen der Anbieter A, B und C gewährleistet werden, da diese Daten für die App liefern sollen. Die drei Anbieter stellen zu diesem Zweck die aktuellen und historischen Positionen ihrer Fahrzeuge zur Verfügung. Somit ist es notwendig, die unterschiedlichen Systeme der Anbieter von E-Rollern interoperabel zu machen. In Tabelle I sind einige technischen Spezifikationen der Datenquellen dargestellt. Hieraus ergeben sich diverse technische Herausforderungen.
Ein Problem für die Dateninteroperabilität ergibt sich aus den unterschiedlichen Datenübertragungsprotokollen, die von den

Tabelle I
ANWENDUNGSBEISPIEL: ANBIETER VON E-ROLLERN IN EINER SMART CITY

	Anbieter A	**Anbieter B**	**Anbieter C**
Dateiformat	XML	XML	CSV
Zeichenencoding	UTF-8	ASCII	UTF-8
Übertragungsprotokoll	HTTP	HTTP	MQTT

Anbietern B und C eingesetzt werden (vgl. Tabelle I). Damit die Daten von beiden Anbietern verwendet werden könne ist die Berücksichtigung von HTTP und MQTT erforderlich. Des Weiteren verwenden die Anbieter B und C unterschiedliche Dateiformate (XML und CSV) für das Speichern der Fahrzeugpositionen (vgl. Tabelle I). Diese müssen ebenfalls berücksichtigt werden. Während die Anbieter A und B (vgl. Tabelle I) dieselben Dateiformate verwenden, sind diese jedoch einmal in UTF-8 und einmal in ASCII codiert. Auch dies muss unterstützt werden.
Da die Anbieter die aktuelle Position ihrer Fahrzeuge kontinuierlich veröffentlichen, kann die Smart Region diese Datenströme mit weiteren aktuellen Verkehrsdaten integrieren (z.B. von Verkehrskameras), um in Echtzeit ein Bild über die Verkehrslage in der Stadt zu generieren und über ihre App veröffentlichen. Des Weiteren könnte in diesem Anwendungsbeispiel die Smart Region über ihre App versuchen, eine, für die Bewohner optimale Verteilung der E-Roller zu gewährleisten. Um die ideale Anzahl und den passenden Ort für das Bereitstellen der Fahrzeuge zu ermitteln, könnte die Stadt vorliegende historische und aktuelle Verkehrsdaten analysieren. Dies erfordert die Interoperabilität vieler Datenquellen, die sich im Bereich von sehr großen und schnell wachsenden Datenmengen (Big Data) bewegen. Eine weitere Herausforderung kann sich aus der Zugriffsgeschwindigkeit auf die Position der E-Roller ergeben. Wenn die App z.B. jede Minute die Fahrzeugpositionen aktualisiert, die Anbieter jedoch jede Sekunde, arbeitet die App womöglich mit veralteten Daten.
Aus dem genannten Anwendungsbeispiel lassen sich somit sechs unterschiedliche technische Herausforderungen ableiten:

- Unterschiedliche Übertragungsprotokolle
- Unterschiedliche Dateiformate
- Unterschiedliches Zeichenencoding
- Integration von Datenströmen
- Zu große Datenquellen
- Unterschiedliche Zugriffsgeschwindigkeiten

Im Folgenden werden diese Herausforderungen genauer vorgestellt und mögliche Lösungsansätze aus der Literatur erläutert und diskutiert.

III. UNTERSCHIEDLICHE ÜBERTRAGUNGSPROTOKOLLE

Eine Herausforderung bei der Dateninteroperabilität ergibt sich aus den Übertragungsprotokollen, welche von den Datenquellen genutzt werden. So kann es z.B. bei der Integration von zwei Datenquellen vorkommen, dass die eine das Hypertext Transfer Protocol (HTTP) und die andere Quelle das Message Queuing Telemetry Transport (MQTT) Protokoll zur Datenübertragung verwendet. Ein Bereich in dem diese Problematik von hoher Relevanz ist, ist das IoT. Hier werden eine Vielzahl unterschiedlicher Protokolle eingesetzt [6]. Da das IoT eine Möglichkeit zur Datenerfassung und -integration insbesondere im Falle verteilter Systeme darstellt, ist es für Smart Cities von hoher Bedeutung [4, S.1].

A. Lösungsansätze

Das Problem der Interoperabilität im IoT ist eine durchaus bekannte Herausforderung in der Forschung. Hierbei ist es wichtig, verteilte Systeme so zu entwickeln, dass sie leicht erweiterbar sind, um die Interoperabilität, auch im Bereich der Protokolle, zu gewährleisten [6, S. 2363 f.]. Somit können schnell Gateways entwickelt werden, welche dann die unterschiedlichen Protokolle handhaben können. Der Einsatz solcher Gateways ist ein gängiger Lösungsansatz in der Literatur [4, S. 3 ff.].
Hierbei lässt sich das IoT als Architektur mit drei Ebenen betrachten. Die Anwendungsebene, welche für die Datenverarbeitung und das Anbieten des Services verantwortlich ist. Die Netzwerkebene, welche die Kommunikationsinfrastruktur darstellt. Und die Sensorebene, die die benötigten Daten erfasst. Ein IoT-Gateway dient als Bindeglied zwischen der Sensor- und der Netzwerkebene. Hierfür ist es auch in der Lage, die diversen Übertragungsprotokolle zu konvertieren [7].
Gil et al. [4] stellen einen weiteren Lösungsansatz vor. Hier dient ein zentraler Server als Schnittstelle für die unterschiedlichen Protokolle. Der Server stellt zeitgleich die IoT-Anwendung zur Verfügung. Für die Protokollintegration ist der Server mit Interfaces ausgestattet, welche die Handhabung der einzelnen Protokolle übernehmen. Die Interfaces stellen dann die Daten, die über die Protokolle übertragen wurden, der nächst höheren Schicht zur Verfügung. Hierbei handelt es sich um eine transparente Datenbrücke, welche alle Daten vereinheitlicht zusammenfasst [4, S. 8 f.].

B. Diskussion

Die ermittelten Ansätze zeigen, dass die Problematik der unterschiedlichen Übertragungsprotokolle generell lösbar ist. Allerdings ist es fragwürdig, ob alle Lösungsansätze, insbesondere im Bezug auf D4A geeignet sind. Die IoT-Gateways stellen zwar eine Lösung dar, jedoch werden sie auf Seite der Datenquellen installiert. Somit ist dieser Ansatz nur umsetzbar, wenn die Datenquellen diesen bereits unterstützen oder bereit sind, IoT-Gateways zu installieren. Grade im D4A-Projekt besteht nicht immer direkter Zugriff auf die Beschaffenheit der Datenquellen. Zudem sind die Gateways spezifisch für das IoT und eine Anwendung in anderen Bereichen ist nicht zwangsläufig möglich. Hingegen bietet der zentrale Server durch das Hinzufügen weiterer Schnittstellen den Vorteil einer leichten Erweiterbarkeit. Dieser Ansatz stellt allerdings einen höheren Entwicklungsaufwand dar, insbesondere wenn viele unterschiedliche Protokolle berücksichtigt werden müssen.

IV. UNTERSCHIEDLICHE DATEIFORMATE

Die selben Informationen lassen sich unterschiedlich darstellen. So ist es z.B. möglich identische personenbezogene Daten im XML oder im CSV-Dateiformat darzustellen, wie in den Abbildungen 1 und 2 dargestellt ist. Hierdurch ändert sich zwar die Struktur der Daten jedoch nicht deren Inhalt. Da im D4A-Projekt viele unterschiedliche Datenquellen integriert werden sollen, ist es möglich, dass diese ihre Daten in unterschiedlichen Dateiformaten angeben.

```
<?xml version="1.0" encoding="UTF-8"?>
<customer>
    <title>Mr</title>
    <name>Smith</name>
    <city>Ottawa</city>
    <state>ON</state>
</customer>
<customer>
    <title>Mrs</title>
    <name>Jones</name>
    <city>Winnipeg</city>
    <state>MB</state>
</customer>
```

Abbildung 1. Personenbezogene Daten im XML-Format [8]

```
Mr,Smith,Ottawa,ON
Mrs,Jones,Winnipeg,MB
```

Abbildung 2. Personenbezogene Daten im CSV-Format [8]

A. Lösungsansätze

Für die Interoperabilität unterschiedlicher Dateiformate gibt es sowohl in der Forschung [9] als auch in der Wirtschaft [8] einige Lösungen. Der *WebSphere Adapter for Flat Files* [8] von IBM arbeitet ohne die, in den Dateien enthaltenen, Daten zwischenzuspeichern. Hierfür unterteilt die Software zwischen einem Input- und einem Outputprozess. Der Inputprozess wird angestoßen, sobald eine neue Datei erstellt wird. Die Software setzt dann die, in der Datei enthaltenen, Daten in sogenannte *Events* um. Diese Events basieren auf einem systeminternen Datenmodell und können in einer Datenbank persistiert werden. Hierdurch werden die Daten unabhängig von ihrem ursprünglichen Dateiformat gemacht. Um die Daten dann einer Anwendung zur Verfügung zu stellen werden die *Events* in sogenannte *business objects* überführt und exportiert. Somit bleiben Metadaten der Originaldatei erhalten. Bei dem Outputprozess lassen sich die Daten dann in einer Datei mit beliebigem Dateiformat speichern [8].
Lopes et al. [9] stellen einen weiteren Ansatz vor. Dieser hat das Ziel die Dateninteroperabilität im Gesundheitswesen zu erleichtern. Dieser Ansatz basiert auf der Integration vieler Datenquellen in einer zentralen *Storage Engine*. Hierfür besitzt die vorgestellte Anwendung Adapter welche Daten aus CSV, XML, SQL und SPARQL Datenquellen einlesen können. Das überführen der Daten aus den unterschiedlichen Dateiformaten in die, von der *Storage Engine* benötigte Datenstruktur, geschieht anhand einer Konfigurationsdatei.

B. Diskussion

Die ermittelten Lösungsansätze zeigen, dass die Interoperabilität zwischen unterschiedlichen Dateiformaten generell möglich ist. Bei dem *WebSphere Adapter for Flat Files* ist es fragwürdig, ob dieser für die Anwendung in Smart Regions geeignet ist, da bei dem System ein starker Fokus auf Unternehmen in der freien Wirtschaft liegt. Die Beantwortung dieser Frage wird durch die sehr oberflächliche Beschreibung der Architektur weiter erschwert. Der Ansatz der zentralen *Storage Engine* ähnelt dem Ansatz des zentralen Servers aus Abschnitt III. Somit stellt er eine leicht zu erweiternde Lösung dar, die jedoch mit steigender Anzahl der zu berücksichtigenden Dateiformate immer komplexer umzusetzen wird.

V. UNTERSCHIEDLICHE ZEICHENENCODING

Neben den, in Abschnitt IV, vorgestellten Dateiformaten sind die einzelnen Zeichen in diesen Dateien nach einem bestimmten Standard encoded. Dies ist notwendig, da Computer grundlegend nur binäre und keinen komplexen Zeichen verarbeiten können. Es gibt viele unterschiedliche Zeichenencodings, wie z.B. UTF-8 oder ASCII. Somit ist es notwendig diese Zeichen, wie z.B. die Buchstaben eines Alphabets, in eine Darstellung zu mappen, die von einem Computer interpretiert werden kann [10, S.2].

A. Lösungsansatz

Um eine Interoperabilität zwischen verschiedenen Zeichenencodings zu gewährleisten, wird das oben bereits erwähnte Konzept des Mappings verwendet. Eine solches mapping wird vielfach von Programmiersprachen nativ unterstützt. So kann z.B. Java über einen eigenen Constructor ein Byte-Array als UTF-8 oder ASCII String darstellen. Dieses mapping lässt sich auch manuell umsetzen. Hierfür muss das Encoding der Datenquelle bekannt sein [11]. Ein Zeichenencoding besteht aus einem *Charset*. Dieses gibt an welche Zeichen in dem Encoding dargestellt werden können. Für jedes dieser Zeichen gibt es eine einzigartige Byte-Darstellung [12]. Für das Übersetzen eines Quell- in ein Zielencoding müssen nun die Bytes, welche ein einzelnes Zeichen darstellen, ausgelesen werden. Wie viele Bytes ein Zeichen darstellen ist dabei abhängig von dem jeweiligen Encoding [11]. Anschließend lässt sich diese binäre Darstellung so abändern, dass sie derselben Byte-Darstellung für dasselbe Zeichen im Zielencoding entspricht. Denkbare wäre hier ein ähnlicher Ansatz wie bei Gil et al. [4]. Also die Verwendung eines zentralen Servers mit einem einheitlichen Encoding, welcher mit Schnittstellen für die jeweiligen Zeichenencoding ausgestattet ist.

B. Diskussion

Bei der Recherche zu möglichen Lösungsansätzen für die Interoperabilität unterschiedlicher Zeichencodierungen konnten keine Systeme ermittelt werden, die das vorgestellte Konzept implementieren. Eine mögliche Erklärung ist, dass die

meisten modernen Systeme den UTF-8 Standard verwenden und das Interoperabilitätsproblem nur bei älteren Systemen auftritt [11]. Das vorgestellte Konzept zeigt jedoch, dass das Problem generell gelöst und diese Lösung dann im Kontext größerer Systeme umgesetzt werden können. Denkbar wäre hier ein zentraler Server mit Schnittstellen für die jeweiligen Encodings einzusetzen, wie es auch in den Abschnitten III und IV vorgestellt wurde.

VI. INTEGRATION VON DATENSTRÖMEN

Eine weitere technische Herausforderung ergibt sich aus der Aktualität von Daten. Wenn Daten von einer Zielanwendung in Echtzeit benötigt werden, kann es notwendig sein sogenannte Datenströme zu integrieren. Diese liefern einen kontinuierlichen Strom an hoch aktuellen Daten [5, S. 61 f.].

A. Lösungsansatz

Um die Interoperabilität von Datenströmen allgemein zu vereinfachen, ist es sinnvoll Standards zu verwenden. Als Beispiel könnten der *SWE Service Model Implementation Standard* des *Open Geospatial Consortium*, welche spezifisch für verteilte Sensornetze entwickelt wurde, verwendet werden [13].
Die Interoperabilität von Datenströmen kann durch den Einsatz eines sogenannten *Data Stream Management System (DSMS)* erreicht werden. Diese bestehen in der Regel aus sechs Komponenten. Der *Stream- and Query Catalog* fungiert als organisatorische Einheit, welche die Pipelines, also den Datenfluss, definiert. Hierbei hat er auch Einfluss darauf, welche Inputs als Datenstrom angenommen und mit welcher Priorität diese verarbeitet werden. *Monitoring and Planning Engine* dient zur Verwaltung der Hardwareressourcen. Der *Persistence Manager* ist eine optionale Komponente, die nur dann eingesetzt werden muss, wenn die verarbeiteten Daten permanent gespeichert werden sollen. Der *Stream Receiver* übernimmt das tatsächliche Empfangen der Datenströme und gibt diese an den *Processor* weiter. Des Weiteren besitzt die Komponente einen sogenannten *Load Shedder*. Das ist notwendig, um selektiv Daten zu verwerfen und diejenigen Daten zu behalten, die aufgrund ihrer Definition als wichtig angesehen werden. Dies dient z.B. dazu die, durch die Hardware gegebenen, limitierte Rechenleistung einzuhalten. Der *Processor* verarbeitet die ihm gelieferten Daten dann. Abschließend übernimmt der *Output Manager* die ausgehende Kommunikation der verarbeiteten Daten [5, S. 65 ff.]. In Abbildung 3 ist ein DSMS schematisch dargestellt.
Eine der größten Herausforderungen eines DSMS ist es die Menge der zu verarbeitenden Daten zu regulieren. Zum einen sollen die Daten aus den Datenströmen hoch aktuell sein, zum anderen stehen nur begrenzte Hardwareressourcen für deren Verarbeitung zu Verfügung. Um die Datenmenge zu regulieren, wird ein so genannter *Sliding Window Ansatz* verwendet. Hierbei werden die Daten, welche von einem Datenstrom geliefert werden, nur ausschnittsweise, also wie durch ein Fenster (Window), betrachtet. Alle Daten, die sich außerhalb dieses Fensters befinden, verfallen. Die Größe des Fensters kann anhand einer Zeitperiode (physical window), also z.B. der Zeit nachdem die Daten eingetroffen sind, oder anhand der Menge der eingetroffenen Daten (logical window) festgelegt werden [5, S. 65 ff.].
Anwendungsbeispiele für DSMS sind die SDSP Middleware [14] oder das Überwachungssystem zur Luftqualität von Samadzadegan et. al. [15]. Die SDSP Middleware dient zur Intergration von Datenströmen in Heimnetzwerken. Interressant hierbei ist, dass vor der Datenanalyse die eintreffenden Datenströme zu virtuellen Sensoren zusammengefasst werden. Dies ermöglicht eine vereinfachte Gruppierung einzelner Sensoren [14]. Das von Samadzadegan et. al. vorgestellte System dient zur Überwachung der Luftqualität in der Stadt Teheran [15].

B. Diskussion

Durch den Einsatz von DSMS lässt sich die Interoperabilität von Datenströmen gewährleisten. Dies zeigt auch das Anwendungsbeispiel der SDSP Middleware [14]. Allerdings handelt es sich bei *Data Stream Management Systems* um komplexe Anwendungen. Die Umsetzung einer solchen Lösung wird sich somit als sehr ressourcenintensiv gestalten.

VII. ZU GROSSE DATENQUELLEN

In Smart Cities werden eine hohe Anzahl an Sensoren und anderen Systemen eingesetzt, welche kontinuierlich eine große Menge an Daten produzieren [16, S. 2]. Dies führt zu sehr großen und strak wachsenden Datenquellen, die sich folglich im Bereich Big Data bewegen. Für das Entwickeln einer heterogenen Datenlandschaft in dem D4A-Projekt, kann es notwendig sein solche Datenquellen miteinander zu integrieren. Hierbei treten einige Herausforderungen auf, die gelöst werden müssen. In *Challenges of data integration and interoperability in big data* werden die Folgenden zentralen Herausforderungen identifiziert [17, S.39 f.]:

- Berücksichtigung großer Datenmengen
- Dateninkonsistenzen
- Anfrageoptimierung
- Unzureichende Ressourcen
- Skalierbarkeit
- Einrichten von Support Systemen
- ETL-Prozesse

A. Lösungsansätze

Als Lösungsansatz für die Interoperabilität in Big Data wurden eine Vielzahl an Interoperabilitäts-Frameworks entwickelt [18]. Zu diesen gehören unter anderem NIST Big Data Interoperability Framework [19], FAIR Data Points Supporting Big Data Interoperability (FDP) [20] oder das Smart City Interoperability Framework [21]. Diese Frameworks schaffen allgemeine Konzepte, um die Interoperabilität im Bereich Big Data zu gewährleisten. Jedoch gehen diese meistens nicht detailliert auf ihre Implementierung ein. Beispielsweise schlagen die FAIR Data Points eine konkrete Softwarearchitektur vor, verzichten aber bewusst auf das Nennen konkreter Technologien. FDPs fungieren als Softwareschnittstelle für den Zugriff auf die eigentlichen Datenquellen [20, S. 3]. Noch abstrakter

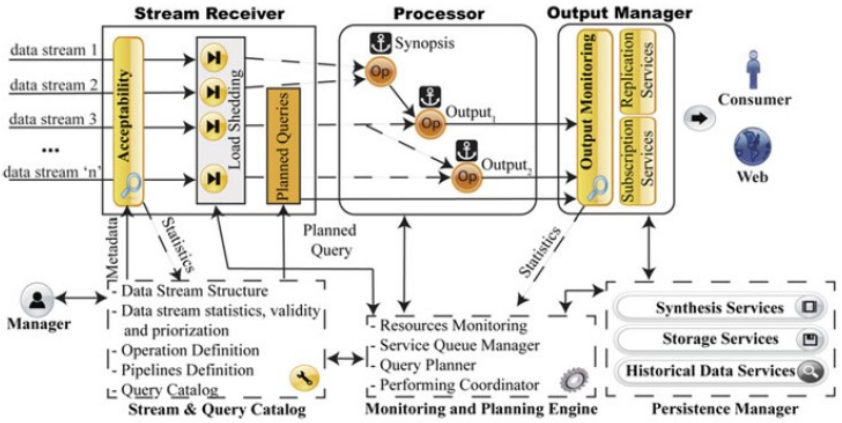

Abbildung 3. Schematische Darstellung eines DSMS (entnommen aus [5, S. 65])

arbeitet das Smart City Interoperability Framework. Dieses basiert auf dem sehr abstrakten Modell einer Smart City und gibt keine spezifische Softwarearchitektur vor [21].
Ein weiterer Ansatz für die Verbesserung der Interoperabilität im Bereich Big Data ist die Berücksichtigung von, in der Literatur sogenannten, Best Practices [17] [22]. Diese sind meist sehr abstrakt und gehen nicht über die konzeptionelle Ebene hinaus. So gibt Kadadi et. al. [17] grobe Lösungsansätze für die oben genannten Probleme an. Für die Berücksichtigung großer Datenmengen werden das Beschaffen schnellerer Hardware oder das Verwenden effizienterer Algorithmen genannt. Für das Verhindern von Dateninkonsistenzen können das Aufteilen der Daten in kleinere Einheiten oder das Anwenden geeigneter Methoden wie z.B. MapReduce verwendet werden. Die Anfrageoptimierung ist z.B. durch das Verwenden von verteilten Joins erreichbar. Für das Problem der unzureichenden Ressourcen schlagen Kadadi et. al. [17] die Weiterbildung von Personal oder die Umsetzung von großen Änderungen in kleineren Schritten vor. Eine hohe Skalierbarkeit kann durch den Einsatz von geeigneten Technologien, wie z.B. Hadoop, erreicht werden. Das Einrichten von Support Systemen, um z.B. Fehler schnell beheben zu können, wird ebenfalls als sinnvoll erachtet. ETL Prozesse werden von Kadadi et. al. allgemein als gut bewertet [17, S. 39 f.]. Scheerlinck et. al. [22] erweitert diese unter anderem um rechtliche Hinweise [22, S. 8 f.], das Schaffen einer einheitlichen Datenstruktur über die gesamte Organisation hinweg [22, S. 10] oder das Analysieren von Feedback, um sich an Änderungen in den Anforderungen an die Daten anpassen zu können [22, S.17].

B. Diskussion

Die Interoperabilität sehr großen Datenquellen kann von der Implementierung von Interoperabilitäts-Frameworks oder durch das Umsetzen der Best Practices erreicht werden. Hierbei handelt es sich jedoch um generelle Konzepte und nicht um konkrete Systeme zum Lösen der Problematik. Des Weiteren setzen diese Konzepte meist auf Seite der Datenquelle an. Sie können also auch nur von der Datenquelle umgesetzt werden. Wenn dies nicht der Fall ist, bleibt nur die Möglichkeit die Daten in einem eigenen System zu integrieren. Hiebei kann die Berücksichtigung von Best Practices und den Interoperabilitäts-Frameworks große Vorteile bieten.

VIII. UNTERSCHIEDLICHE ZUGRIFFSGESCHWINDIGKEITEN

Für die digitale Kommunikation können unterschiedlichste Technologien eingesetzt werden. Durch z.B. den Einsatz vieler Kommunikationsprotokolle gibt es ein breites Spektrum an Datenübertragungsraten [23, S. 183]. Hierdurch ist auch die Zugriffsgeschwindigkeit auf einzelne Datenquellen sehr breit gefächert. Diese kann z.B. dazu führen, dass Daten verloren gehen oder der Konsument ganz abstürzt. Da im D4A-Projekt diverse unterschiedliche Datenquellen integriert werden sollen, kann es notwendig sein, mit unterschiedlichen Zugriffsgeschwindigkeiten umgehen zu können.

A. Lösungsansätze

Ein in der Literatur genannter Lösungsansatz ist die Verwendung einer zentralen Kommunikationsinfrastruktur, welche die unterschiedlichen Datenübertragungsraten handhabt [24] [25]. Morse et. al. [24] verwenden hierfür ein Publisher/Subscriber-Konzept. Hierfür melden sich Datenproduzenten (Publisher) und Datenkonsumenten (Subscriber) bei der zentralen Kommunikationsinfrastruktur mit ihrer gewünschten Datenübertragungsrate an. Somit können die Datenproduzenten die Frequenz, mit der sie Nachrichten auf die Infrastruktur publishen, auf die maximal geforderte Übertragungsrate anpassen [24, S. 117]. Ayaida et. al. [25] verwenden einen ähnlichen Ansatz. Hier werden die Geräte von der Kommunikationsinfrastruktur (*C2A System*) erkannt und konfiguriert. Dabei werden relevante Informationen über die Geräte in einer *PerphTable* erfasst. Hierzu zählt unter anderem die Baud Rate (Datenübertragungsrate). Des Weiteren besitzt das System ein zentrales Datenverarbeitungsmodul. Dieses kann, nach der Verarbeitung der Daten, die korrekte Baud Rate für einen Datenkonsumenten aus der *PerphTable* auswählen und die Daten somit in der korrekten Geschwindigkeit korrekt übertragen [25].
Einen anderen Ansatz verfolgen Rahman et. al. [26]. Hier

werden Daten von langsamen Produzenten von einer Middleware gesammelt. Anschließend werden die Daten dann von der Middleware in einer höheren Geschwindigkeit an das Ziel gesendet [26, S. 3].

B. Diskussion

Zu dem Thema der Zugriffsgeschwindigkeiten konnten nur sehr wenige Arbeiten während der Literaturrecherche gefunden werden. Dies schränkt den Umfang der betrachteten Lösungsansätze stark ein. Zudem ist der Ansatz von Morse et. al. [24] sehr abstrakt gehalten und mehr als grobes Konzept anstatt eines Anwendungsbeispiels zu betrachten. Es stellt dennoch einen sinnvollen Lösungsansatz dar. Ayaida et. al. [25] eignet sich als Lösungsansatz gut. Es ist jedoch eingeschränkt, da sich die Lösung auf die Datenübertragung über einen physischen Bus beschränkt. Auch Rahman et. al. [26] zeigt, dass die Dateninteroperabilität bei unterschiedlichen Zugriffsgeschwindigkeiten gewährleistet werden kann.

IX. FAZIT

Allgemein ist festzuhalten, dass für alle sechs technischen Herausforderungen Lösungsansätze in der Literatur gefunden werden konnten. Die Anwendung einiger Konzept im Rahmen des D4A-Projekts ist allerdings fragwürdig. Z.B. stellen Lösungen, die von den Datenquellen umgesetzt werden müssen, wie z.B. IoT-Gateways, keine geeigneten Ansätze dar, da nicht immer Zugriff auf die Struktur der Datenquellen besteht. Zudem sind einige Lösungsansätze nicht primär für den Einsatz in Smart Regions entwickelt worden. Beispielsweise legt der *WebSphere Adapter for Flat Files* [8] einen starken Fokus auf Unternehmen. Hier ist eine Anwendung in Smart Regions ebenfalls fragwürdig. Auch konnten keine Anwendungsbeispiele oder Konzepte gefunden werden, die alle oder zumindest viele technische Herausforderungen auf einmal lösen. Es ist also empfehlenswert, die im D4A-Projekt auftretenden Dateninteroperabilitätsherausforderungen zu ermitteln und ein allgemeines Konzept zu schaffen, welches als Lösung fungiert.

LITERATUR

[1] "Impulspapier zu Normen und Standards – Smart City," 2017, visited on 05.06.2023. [Online]. Available: https://www.din.de/resource/blob/237630/4a7ee615d0ae296706f6a95705f584c1/smart-city-impulspapier-zu-normen-und-standards-data.pdf

[2] About us. Visited on 05.06.2023. [Online]. Available: https://www.interregnorthsea.eu/dataforall/about-us

[3] "Interoperability," o.j., visited on 11.07.2023. [Online]. Available: https://edps.europa.eu/data-protection/our-work/subjects/interoperability_en

[4] S. Gil, G. D. Zapata-Madrigal, R. García-Sierra, and L. A. Cruz Salazar, "Converging IoT protocols for the data integration of automation systems in the electrical industry," *Journal of Electrical Systems and Information Technology*, vol. 9, no. 1, pp. 1–21, 2022. [Online]. Available: https://jesit.springeropen.com/articles/10.1186/s43067-022-00043-4

[5] M. J. Diván and M. L. Sánchez Reynoso, "An Architecture for the Real-Time Data Stream Monitoring in IoT," in *Multimedia Big Data Computing for IoT Applications*. Springer, Singapore, 2020, pp. 59–100. [Online]. Available: https://link.springer.com/chapter/10.1007/978-981-13-8759-3_3

[6] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[7] H. Chen, X. Jia, and H. Li, "A brief introduction to IoT gateway," in *ICCTA 2011*. Stevenage, England: IET, 2012, pp. 610–613.

[8] (o.J.) IBM Business Automation Workflow: Overview. Visited on 22.06.2023. [Online]. Available: https://www.ibm.com/docs/en/baw/22.x?topic=files-overview

[9] P. Lopes and J. L. Oliveira, "A semantic web application framework for health systems interoperability," in *MIXHS 2011*, ser. ACM International Conference Proceeding Series, M.-M. Bouamrane and C. Tao, Eds. New York, N.Y.: ACM Press, 2011, pp. 87–90.

[10] A. M. McEnery and R. Z. Xiao, "Character encoding in corpus construction," o.J., visited on 22.06.2023. [Online]. Available: https://eprints.lancs.ac.uk/id/eprint/60/

[11] René Kiessling, "Character Encodings for Message Exchange with Legacy Systems Done Right," 22.02.2022, visited on 14.07.2023. [Online]. Available: https://medium.com/@rdkiessling/character-encodings-for-message-exchange-with-legacy-systems-done-right-3d1048a01e27

[12] K. Chandrakant, "Guide to Character Encoding," *Baeldung*, 01.12.2018, visited on 14.07.2023. [Online]. Available: https://www.baeldung.com/java-char-encoding

[13] Open Geospatial Consortium, "SWE Service Model Implementation Standard - Open Geospatial Consortium," 2023, visited on 17.07.2023. [Online]. Available: https://www.ogc.org/standard/swes/

[14] Y.-S. Noh, D.-O. Han, and Y.-C. Byun, "Real-time data stream processing for ubiquitous home network systems," in *Multimedia and Ubiquitous Engineering (MUE), 2010 4th International Conference on*. IEEE, 2010, pp. 1–4.

[15] F. Samadzadegan, H. Zahmatkesh, M. Saber, and H. J. Ghazi khanlou, "AN INTEROPERABLE ARCHITECTURE FOR AIR POLLUTION EARLY WARNING SYSTEM BASED ON SENSOR WEB," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XL-1/W3, pp. 459–462, 2013.

[16] I. A. T. Hashem, V. Chang, N. B. Anuar, K. Adewole, I. Yaqoob, A. Gani, E. Ahmed, and H. Chiroma, "The role of big data in smart city," *International Journal of Information Management*, vol. 36, no. 5, pp. 748–758, 2016, visited on 23.06.2023. [Online]. Available: https://www.researchgate.net/publication/301803005_The_Role_of_Big_Data_in_Smart_City

[17] A. Kadadi, R. Agrawal, C. Nyamful, and R. Atiq, "Challenges of data integration and interoperability in big data." IEEE, 2014.

[18] A. Bousdekis and G. Mentzas, "Enterprise integration and interoperability for big data-driven processes in the frame of industry 4.0," *Frontiers in big data*, vol. 4, p. 644651, 2021.

[19] "NIST Big Data Interoperability Framework," Gaithersburg, MD, 2019.

[20] Luiz Olavo Bonino da Silva Santos, Mark D. Wilkinson, Arnold Kuzniar, Rajaram Kaliyaperumal, and Kees Burger, "Fair data points supporting big data interoperability," in *Enterprise Interoperability in the Digitized and Networked Factory of the Future*, 2016, pp. 270–279. [Online]. Available: https://www.researchgate.net/publication/309468587_FAIR_Data_Points_Supporting_Big_Data_Interoperability

[21] J.-y. Ahn, J. S. Lee, H. J. Kim, and D. J. Hwang, "Smart city interoperability framework based on city infrastructure model and service prioritization," in *2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN)*. IEEE, 2016, pp. 337–342.

[22] J. Scheerlinck, F. Van Eeghem, and N. Loutas, "Big data interoperability analysis," 2018, visited on 18.07.2023. [Online]. Available: https://joinup.ec.europa.eu/sites/default/files/document/2018-05/SC508DI07171%20D05.02%20Big%20Data%20Interoperability%20Analysis_v1.00.pdf

[23] A. Aragues, J. Escayola, I. Martinez, P. del Valle, P. Munoz, J. Trigo, and J. Garcia, "Trends and challenges of the emerging technologies toward interoperability and standardization in e-health communications," *IEEE Communications Magazine*, vol. 49, no. 11, pp. 182–188, 2011.

[24] K. L. Morse, M. Lightner, R. Little, B. Lutz, and R. Scrudder, "Enabling simulation interoperability," *Computer*, vol. 39, no. 1, pp. 115–117, 2006.

[25] M. Ayaida, H. El Mehraz, L. Afilal, and H. Fouchal, "Communication interoperability model for embedded devices," in *GLOBECOM 2011 - 2011 IEEE Global Communications Conference*, I. Staff, Ed. [Place of publication not identified]: IEEE, 2011, pp. 1–5.

[26] T. Rahman and S. K. Chakraborty, "Provisioning technical interoperability within zigbee and ble in iot environment," in *IEMENTech*, I. o. E. Engineers and Electronics, Eds. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, 2018, pp. 1–4.

Identifizieren und Lösen von Datenkonflikten

1st Bothmann Kjell
Carl von Ossietzky Universität Oldenburg, Department für Informatik
Abteilung Softwaretechnik
Oldenburg, Deutschland
kjell.bothmann@uni-oldenburg.de

***Abstract*—Ein zentrales Problem in dem immer wichtiger werdenden Bereich der Dateninteroperabilität sind Datenkonflikte. Hierbei handelt es sich um Fälle, in denen Informationen von zwei Datenquellen unterschiedlich für das gleiche Realweltobjekt sind. Diese Arbeit sammelt Lösungsstrategien aus der Literatur für das Identifizieren und Lösen dieser Konflikte und bewertet, wie gut diese Strategien im Kontext des D4A Projektes für Smart Regions verwendet werden können.**

***Index Terms*—Smart Region, Record Linkage, Datenintegration, Datenkonflikte**

I. MOTIVATION

Dateninteroperabilität, ist die Fähigkeit von Systemen, Daten und Informationen auszutauschen [1]. Mit der steigenden Wichtigkeit von Dateninteroperabilität wird auch das Lösen von Datenkonflikten immer relevanter. Diese Konflikte entstehen z.B. wenn Daten aus unterschiedlichen Quellen widersprüchliche Informationen liefern. Mit steigender Anzahl und Komplexität von Datenquellen, welche integriert werden müssen, gewinnt das Lösen von Datenkonflikten stetig an Bedeutung, denn wenn diese Konflikte ungelöst bleiben, können sie zu einer schlechteren Datenqualität beitragen. Zusätzlich kann das Lösen von Datenkonflikten die Qualität der Daten oft sogar verbessern, denn es können widersprüchliche Daten entfernt und qualitativ schlechte Daten aufgebessert werden.

Ein Anwendungskontext, in dem das Integrieren von Daten durch die hohe Anzahl an Datenquellen besonders relevant ist, sind *Smart Regions*. Dort werden Daten aus vielen Quellen gesammelt und vereint, um den Bewohnern einer Region Dienstleistungen anbieten zu können, die ihr Leben in der Region erleichtern. Hierbei treten auch Datenkonflikte auf. Ein Projekt welches gezielt untersucht wie Daten effizient und zielgerichtet integriert werden können für die Anwendung in einem Smart Region Kontext ist das Data for all (D4A) Projekt [2]. Diese Arbeit konzentriert sich auf die Beischreibung von Lösungsansätzen für das Identifizieren und Lösen von Datenkonflikten für Smart Regions, im Rahmen des D4A Projektes.

II. PROBLEMSTELLUNG

Um die Herausforderungen, welche sich aus Datenkonflikten ergeben, zu verstehen, muss zunächst der Begriff geklärt werden. Datenkonflikte treten auf, wenn eine oder mehrere Quellen für das Selbe Realweltobjekt unterschiedliche Werte bei einem semantisch äquivalentem Attribut haben [3, S.7].

Ungelöste Datenkonflikte können zu vielen Problemen führen, dazu gehören:

- Widersprüchliche Informationen: Wenn Datenkonflikte nicht aufgelöst sind, dann werden bei der Integration von Datenquellen widersprüchliche Werte übernommen [3].
- Vertrauenswürdigeit: Widersprüchliche Informationen können auch dazu beitragen, dass Daten weniger vertrauenswürdig sind [4].
- Unbekannte Anzahl an referenzierten Objekten: Wenn Datenkonflikte nicht gelöst werden, dann kann das gleiche Realweltobjekt durch mehrere Tupel dargestellt werden, dies führt dazu, dass nicht durch einfaches Zählen der Tupel bestimmt werden kann, wie viele referenzierte Objekte vorliegen [4].
- Redundanz: Wenn ein Realweltobjekt durch mehrere Tupel referenziert wird und nur an einem Änderungen vorgenommen werden, entstehen weitere Konflikte [4].
- Erhöhter Speicheraufwand: Wenn ein Realweltobjekt durch mehr als ein Tupel dargestellt wird, dann müssen mehr Informationen abgespeichert werden.

Das Ziel dieser Arbeit ist es zu ermitteln, wie Datenkonflikte möglichst fehlerfrei und effizient identifiziert und gelöst werden können, um somit die oben genannten Probleme zu beheben. Hieraus ergibt sich die Forschungsfrage:

Wie können im Anwendungskontext des D4A Projektes Datenkonflikte möglichst fehlerfrei und effizient identifiziert und gelöst werden?

Hierzu wird eine Literaturrecherche durchgeführt, in der Lösungsvorschläge aus der Literatur für das Identifizieren und Lösen von Datenkonflikten gesammelt werden.

Im Folgenden wird die Problemdomäne zunächst an einem Anwendungsbeispiel illustriert. Daraufhin werden die Probleme und Lösungsansätze für das Identifizieren von Datenkonflikten betrachtet. Danach werden die Probleme und Lösungsansätze für das Lösen von Datenkonflikten betrachtet.

III. ANWENDUNGSBEISPIEL

Im Folgenden Abschnitt wird ein Anwendungsbeispiel gegeben. Es dient ausschließlich zur Illustrierung und spiegelt keine echten Werte wieder.

Im Anwendungsbeispiel sollen die Daten von zwei Rolleranbietern im Kontext einer Smart Region interoperabel zur Verfügung gestellt werden. Ziel hierbei ist es, das Mobilitätsangebot der Region zu verbessern, indem Daten von unterschiedlichen Unternehmen zusammengeführt werden, um diese den Benutzern einheitlich in einer Anwendung

ID	Name	Nachname	Wohnort	PLZ	Geburtsdatum	Geschlecht
1a	Julius	Müller	Oldenburg	26121	18.05.1990	M
2a	Julius	Müller	Oldenburg	26121	18.05.1990	M
3a	John	Doe	Hamburg	20148	04.02.2000	M

TABLE I
BEISPIELDATEN: ROLLERANBIETER A

ID	Name	Nachname	Wohnort	PLZ	Geburtsdatum	Geschlecht
1b	Julius	Müller	Odlenburg	26121	08.12.2000	M
2b	Erika	Mustermann	Oldenburg	26121	20.08.1985	F
3b	Julius	Müller	Bremen	28203	18.05.1990	M

TABLE II
BEISPIELDATEN: ROLLERANBIETER B

darzustellen. Im Rahmen dieses Projektes haben Rolleranbieter A und B sich entschlossen, ihre Kundendaten zur Verfügung zu stellen. Hierbei ergibt sich das Problem, dass einige der Benutzer ein Konto bei beiden Rolleranbietern haben. Wenn die Daten ohne weitere Verarbeitung zusammengeführt werden, dann wären einige Benutzer doppelt vertreten.

Im Rahmen dieses wurden die Daten der beiden Anbieter in ein semantisch Äquivalentes Schema überführt. Im nächsten Schritt sollen die Daten der beiden Anbieter zusammengeführt werden, wobei Duplikate entfernt und Datenkonflikte sinnvoll aufgelöst werden sollen.

Die Daten von Rolleranbieter A sind in Tabelle I und die von Rolleranbieter B sind in Tabelle II dargestellt. Es handelt sich hierbei um personenbezogene Daten von Benutzern der Rollerdienste.

Im Folgenden wird dieses Anwendungsbeispiel wiederholt aufgegriffen, um Konzepte des Identifizierens und Lösens von Datenkonflikten zu illustrieren.

IV. PROBLEME BEIM IDENTIFIZIEREN VON DATENKONFLIKTEN

In diesem Abschnitt werden die Probleme, die sich beim Identifizieren von Datenkonflikten ergeben, genauer betrachtet.

Damit Datenkonflikte identifiziert werden können, müssen die betrachteten Daten semantisch äquivalent sein (also die gleiche Bedeutung haben). Es existieren Verfahren, mit denen Schema angepasst werden können, sodass sie semantisch äquivalente Attribute aufweisen [3], [4]. Dieser Schritt wird oft Shema Mapping genannt, es ist jedoch nicht Fokus dieser Arbeit, daher wird im Folgenden angenommen, dass die Datenquellen bereits in einem äquivalenten Schema vorliegen.

Das Identifizieren von Datenkonflikten lässt sich in zwei Schritte unterteilen. In Schritt 1 muss ermittelt werden, ob Informationen aus Datenquellen sich auf das gleiche Realweltobjekt referenzieren, da dies eine Voraussetzung für das Vorhandensein von Datenkonflikten ist. Dieser Prozess wird im Rahmen dieser Arbeit als Duplikaterkennung bezeichnet. Duplikate sind dabei definiert als Tupel, die das gleiche Realweltobjekt referenzieren. Duplikate können dabei, aber müssen nicht, identische Werte für ihre Attribute aufweisen. Im zweiten Schritt kann durch Vergleichen der Attribute von identifizierten Duplikaten ermittelt werden ob Datenkonflikte vorliegen. Im Folgenden wird zunächst der erste Schritt, die Duplikaterkennung genauer betrachtet.

Beim Identifizieren von Duplikaten ergeben sich unter anderen zwei Probleme: *Effektivität* und *Effizienz* [3]. Effektivitätsprobleme entstehen dadurch, dass es oft keine Möglichkeit gibt Duplikate eindeutig zu identifizieren, hierdurch können Daten fälschlicherweise als Duplikat erkannt werden (*false positive*) oder Tupel die in Wirklichkeit keine Duplikate sind, können als Duplikate interpretiert (*false negative*) werden. Treten diese beiden Fehler häufig auf, kann dies die Datenqualität verschlechtern. Die Effizienz ist bei der Duplikaterkennung problematisch, wenn Duplikate in großen Datenquellen gesucht werden, da hierbei viele Tupel miteinander verglichen werden müssen.

Ein weiteres Problem bei der Duplikaterkennung sind Werte, die unterschiedlich formatiert sind, aber die gleiche Information abbilden (Synonyme) [5]. Dies kann z.B. beim Namensfeld für eine Person auftreten, wenn eine Datenquelle den Vornamen ganz aufnimmt, während die andere Datenquelle nur den Anfangsbuchstaben des Namens abbildet (z.B. ”Julius Müller” und ”J. Müller”). Dies erschwert das Erkennen, ob zwei Werte, die gleiche Information abbilden, da diese anders dargestellt werden kann.

Zusätzlich kann es Werte geben, die gleich sind, aber unterschiedliche Informationen abbilden (Homonyme). Wenn z.B. für eine Person der Wohnort ”Oldenburg” angegeben, dann ist nicht eindeutig, welche Stadt gemeint ist, da es in Deutschland zwei Städte mit dem Namen Oldenburg gibt. Wenn Homonyme in der Duplikaterkennung nicht richtig behandelt werden, dann ist es möglich, dass zwei Tupel fälschlicherweise als Duplikat identifiziert werden.

Die Probleme bei der Duplikaterkennung lassen dementsprechend zusammenfassen:

- Effektivität
- Effizienz
- Synonyme
- Homonyme

Im zweiten Schritt werden die Attribute von identifizierten Duplikaten miteinander verglichen. Dieser Schritt wurde in keiner der betrachteten Arbeiten genauer behandelt. Allerdings können wie bei der Duplikaterkennung auch Homonyme zum Problem werden. Homonyme können dazu führen, dass mögliche Datenkonflikte nicht identifiziert werden. Wenn z.B. zwei verglichene Attribute den Wert ”Oldenburg” aufweisen, kann es sein, dass hierbei nicht die gleiche Stadt gemeint

ist, obwohl die Felder identisch sind. Auf diese Weise können *false negatives* entstehen. Synonyme sind dabei nicht so problematisch. Wenn z.B. die Werte ”J. Müller” und ”Julius Müller” vorliegen, dann muss geklärt werden, welche Formatierung übernommen wird, der Konflikt muss dementsprechend trotzdem aufgelöst werden.

V. LÖSUNGSANSÄTZE FÜR DAS IDENTIFIZIEREN VON DATENKONFLIKTEN

In diesem Abschnitt werden Lösungsansätze aus der Forschung für das Identifizieren von Datenkonflikten diskutiert. Hierbei werden zunächst die Lösungsansätze der Duplikaterkennung betrachtet.

Für den Prozess der Duplikaterkennung haben alle betrachteten Quellen ein dreischrittiges Verfahren verfolgt. Diese Schritte sind:

1) Paare bilden
2) Attribute vergleichen
3) Tupel bewerten

Diese werden im Folgenden genauer betrachtet:

a) Paare bilden: Um Duplikate zu identifizieren müssen Tupel miteinander verglichen werden, hierfür werden zunächst Paare für den Vergleich gebildet. Eine mögliche Lösungsstrategie ist es alle möglichen Paare zu bilden, sodass jedes Tupel mit jedem anderem verglichen wird. Allerdings wächst dabei die Anzahl der Paare exponentiell zu der Anzahl der Tupel aus den Datenquellen. Deswegen ist diese Strategie oft mit zu viel Rechenaufwand verbunden [6].

Eine Strategie um die Anzahl der benötigten Vergleiche stark zu reduzieren ist Partitionierung, z.B. mit der *sorted Neighborhood Methode* [3], [4]. Hierbei wird ein Sortierschlüssel erzeugt, mit dem die Tupel vorsortiert werden, wodurch Duplikate mit hoher Wahrscheinlichkeit nah beieinander liegen. Auf diese Weise können Paare über Tupel gebildet werden, die in der Sortierung nah beieinander liegen [4]. Hierbei ist es jedoch möglich, dass durch die Wahl des Sortierschlüssels mögliche Duplikate auseinander sortiert werden. Daher kann es sinnvoll sein weitere Durchläufe der *sorted Neighborhood Methode* mit anderen Sortierschlüssel durchzuführen und die Liste der Paare um die zu erweitern, die beim neuem Durchlauf dazugekommen sind [4].

J. Asher et. al. [6] beschreibt, dass *Blocking* eine häufig verwendete Methode für das Reduzieren der zu betrachtenden Paare ist. Beim *Blocking* werden nur Paare gebildet über Tupel, die gleiche Werte für nur einen Teil ihrer Attribute haben [6]. Z.B. könnte in unserem Anwendungsbeispiel zunächst überprüft werden, welche Tupel den gleichen Nachnamen und die gleiche ZIP aufweisen, woraufhin Paare aus den hieraus entstandenen Gruppen gebildet werden können. Hierbei ist es auch möglich, dass mögliche Duplikate übersehen werden, genau wenn diese sich in den betrachteten Attributen unterscheiden. Aus diesem Grund kann es auch beim *Blocking* sinnvoll sein weitere Durchläufe mit anderen betrachteten Attributen durchzuführen.

Weitere Durchläufe können alle Tupel betrachten (*overlapping*) oder es ist möglich nur die Tupel zu betrachten, die noch nicht mit anderen verknüpft wurden (*sequential*) [6].

b) Attribute vergleichen: Nachdem Paare gebildet wurden werden die Attribute der Paare miteinander verglichen. Das Ziel beim vergleichen von Attributen ist es herauszufinden, ob zwei Tupel für ein Attribut die gleiche Information abbilden. Hierfür ist ein einfacher Vergleich, ob die Felder identisch sind nicht ausreichend. Ein Beispiel hierfür ist für Rolleranbieter B in Tabelle II zu sehen. Wird der Wohnort von Tupel 1b und 2b verglichen, lässt sich feststellen, dass diese nicht identisch sind, da Oldenburg in Tupel 1b, mit einem Buchstabendreher, falsch geschrieben ist. Allerdings stellen beide Felder die gleiche Information dar, nämlich, dass die betrachtete Person in Oldenburg wohn. Daher werden für den Vergleich von Attributen oft Ähnlichkeitsmaße verwendet [3], [4], [6].

J. Bleiholder 2009 et. al. [3] merken an, dass gute Ähnlichkeitsmaße oft domänenspezifisch sind. Allerdings ist es möglich jedes Feld als String zu interpretieren, wodurch *String-Distance Maße*, wie z.B. die *Levenshtein-distance*, für den Vergleich gewählt werden können. Die *Levenshtein-distance* gibt an, wie viele Zeichen in einem String gelöscht, getauscht und ergänzt werden müssen, um ihn in einen anderen String umzuwandeln. Der sich hieraus ergebenen Wert kann durch die Stringlänge des längeren Strings geteilt werden, um ein Maß für die Ähnlichkeit der beiden Strings zu bilden [7]. Es ist sinnvoll diese Verfahren um domänenspezifisches Wissen zu erweitern, sodass z.B. der Wert ”J. Müller” und ”Julius Müller” als ähnlich herausgestellt wird, obwohl die Strings sehr unterschiedlich sind [7]. Es scheint ebenfalls sinnvoll zu sein, mehrere Ähnlichkeitsmaße miteinander zu verknüpfen, um die Qualität der Abschätzung zu verbessern [8].

c) Tupel bewerten: Nachdem die Ähnlichkeit der Attribute für ein Paar bestimmt wurde, muss entschieden werden, ob es sich bei dem betrachtetem Paar um ein Duplikat handelt oder nicht. Dabei wird auf Tupelebene bewertet, wie ähnlich das Paar ist. Hierfür gibt es verschiedene Lösungsansätze, die im Folgenden genauer betrachtet werden.

Bei der Bewertung von Tupeln ist zu berücksichtigen, dass nicht alle Felder gleichermaßen aussagekräftig für die Ähnlichkeit zweier Tupel sind. Dass zwei unterschiedliche Personen das gleiche Geschlecht haben, kommt z.B. viel häufiger vor, als dass sie den gleichen Nachnamen haben. Daher ist der Nachname aussagekräftiger für die Übereinstimmung zweier Tupel. Daher ist es sinnvoll die Attribute unterschiedlich zu gewichten bei der Bewertung von Tupeln. J. Bleinholder 2009 et. al. [3] empfehlen dafür die Verwendung eines gewichteten Mittels im Zusammenhang mit einem Grenzwert, welcher festlegt, ab wann ein Tupel als Duplikat interpretiert wird. Zusätzlich kann es auch Felder geben, welche für die Duplikaterkennung nicht relevant sind. K. Hildebrand 2018 et. al. [4] geben an, dass nur Felder, die das Objekt eindeutig beschreiben, bei der Duplikaterkennung verwendet werden müssen.

ObjektID	ID	Name	Nachname	Wohnort	PLZ	Geburtsdatum	Geschlecht
1	1a	Julius	Müller	Oldenburg	26121	18.05.1990	M
1	2a	Julius	Müller	Oldenburg	26121	18.05.1990	M
2	3a	John	Doe	Hamburg	20148	04.02.2000	M
1	1b	Julius	Müller	Odlenburg	26121	08.12.2000	M
3	2b	Erika	Mustermann	Oldenburg	26121	20.08.1985	F
4	3b	Julius	Müller	Bremen	28203	18.05.1990	M

TABLE III
BEISPIELDATEN: ERGEBNISS DER DUPLIKATERKENNUNG

J. Asher et. al. unterscheiden dabei zwischen deterministischen und probabilistischen Verfahren. Deterministische Verfahren treffen die Entscheidung, ob es sich um ein Duplikat handelt, anhand von festgelegten Regeln. Probabilistische Verfahren bestimmen mithilfe eines Wahrscheinlichkeitsmodels die Wahrscheinlichkeit, dass es sich bei einem Paar um ein Duplikat handelt. [6]

Deterministische Verfahren können in vielen Fällen dazu führen, dass es viele *false positives* und *false negatives* gibt [6]. Sie sind aber üblicherweise durch ihre Einfachheit auch Ressourcen schonender.

Das wohl bekannteste und meist erforschte Model für probabilistesche Duplikaterkennung ist die Fellegi-Sunter Methode. Hierbei werden die Wahrscheinlichkeiten m (*matched*) und u (*unmatched*) bestimmt. Dabei ist m die Wahrscheinlichkeit, dass zwei verglichene Attribute den gleichen Wert haben unter der Annahme, dass sie ein Duplikat sind und u ist die Wahrscheinlichkeit, dass zwei verglichene Attribute den gleichen Wert aufweisen unter der Annahme, dass sie kein Duplikat sind. Durch subtrahieren von 1 können die Wahrscheinlichkeiten, dass sich die Werte unterscheiden bestimmt werden ($\bar{u}$ und $\bar{m}$). Daraufhin wird anhand der Ähnlichkeit der Attribute, die vorher berechnet wurde, eine Hypothese aufgestellt. Entweder die Felder sind sich einig, also bilden die gleiche Information ab, oder sie sind sich uneinig. Daraufhin wird entsprechend der Hypothese mit den Gewichten m und u oder mit $\bar{m}$ und $\bar{u}$ ein Wert für das Attribut berechnet. Die Werte die sich hieraus für jedes Attribut ergeben werden daraufhin addiert, woraus sich der Wert des Ähnlichkeitsmaßes für das Tupel ergibt. Je höher dieser Wert ist, desto wahrscheinlicher ist es, dass die Werte Duplikate sind. Daraufhin kann ein Grenzwert festgelegt werden. Der Algorithmus interpretiert Tupel, die diesen überschreiten als Duplikate.

Was die Fellegi-Sunter Methode nicht berücksichtigt ist, dass Attribute häufig nicht unabhängig voneinander sind. Es kann z.B. wahrscheinlicher sein, dass eine Person mit Vornamen ”Julius” männlich ist. Diese Abhängigkeiten können durch die Verwendung von *Machine Learning* Verfahren berücksichtigt werden. Für die Duplikaterkennung können dabei Klassifizierungsalgorithmen, wie z.B. Entscheidungsbäume verwendet werden. [6]

Es gibt auch Verfahren, die auf Erkenntnissen aus der Bayesschen Entscheidungstheorie beruhen. J. Asher et. al. [6] empfehlen jedoch diese Verfahren aufgrund ihrer Komplexität und noch mangelhaften Entwicklung zu meiden.

Wichtig für die betrachteten Verfahren ist, wie sie mit *null*/fehlenden Werten umgehen. Manche Ansätze setzen das Gewicht dieser Felder auf Null, sodass Vergleiche mit *null* nicht berücksichtigt werden. Andere Ansätze verteilen das Gewicht auf die anderen Attribute, um das Gewicht von Tupeln einheitlich zu halten. [6]

Das Ergebnis der Duplikaterkennung ist du Zuweisung einer ID für die Tupel, dabei haben alle Duplikatgruppen jeweils die gleiche ID. Für unser Anwendungsbeispiel könnte das, wie in Tabelle III dargestellt, aussehen. Hinzugekommen ist hierbei die *ObjektID*.

Nachdem die Duplikaterkeunnung abgeschlossen ist, kann durch Vergleich der Attribute festgelegt werden, ob ein Datenkonflikt vorliegt oder nicht. Hierbei können zwei Fälle auftreten:

1) Alle Attribute des betrachteten Paares sind identisch
2) Mindestens ein Attribut unterscheidet sich

Im ersten Fall kann das Duplikat einfach durch übernehmen eines Tupels gelöst werden. Im zweiten Fall wurde ein Datenkonflikt identifiziert, die Lösungsstrategien hierfür werden im Weiterem Verlauf dieser Arbeit behandelt.

VI. BEWERTUNG DER LÖSUNGSANSÄTZE FÜR DAS IDENTIFIZIEREN VON DATENKONFLIKTEN

In diesem Abschnitt wird bewertet, wie gut die gefundenen Lösungsansätze die vorher identifizierten Probleme lösen. Dabei werden zunächst die Probleme für die Duplikaterkennung betrachtet, diese waren: Effizienz, Effektivität, Synonyme und Homonyme.

Synonyme können in der Duplikaterkennung, wenn sie bekannt sind, mithilfe von domänenspezifischen Ähnlichkeitsmaßen für den Attributvergleich gelöst werden. So können z.B. die Werte ”J. Müller” und ”Julius Müller” als ähnlich herausgestellt werden.

Homonyme wurden in keiner der betrachteten Quellen genauer behandelt, aber bei Verfahren wie der Fellegi-Sunter Methode und Machiene Learning Verfahren werden Werten, die potentiell Homonyme sind, niedrigere Gewichte zugewiesen. Auf diese Weise werden Homonyme bereits berücksichtigt.

Eine weitere Methode für den Umgang von Homonymen ist, sofern dies möglich ist, die Homonyme aufzuklären. In unserem Anwendungsbeispiel wäre ein mögliches Homonym die Stadt Oldenburg (da es in Deutschland zwei gibt). Durch hinzuziehen der Postleitzahl ließen sich diese Homonyme jedoch eindeutig aufklären. Diese Methode lässt sich auch für den Attributvergleich, bei der Identifizierung von Datenkonflikten anwenden.

Effektivitätsprobleme, also das Vermeiden von *false positives* und *false negatives* können durch eine gute Wahl des Grenzwertes für die Bewertung von Tupeln vermindert werden. J. Bleinholder 2009 et. al. [3] geben an, dass die Wahl eines geeigneten Grenzwertes schwierig und sehr domänenspezifisch ist. I. Fellegi et. al. [9] teilen in ihrem Verfahren die Tupel in drei Gruppen auf: Duplikate, Ungewisse Tupel, Nicht-Duplikate. Auf diese Weise kann die Anzahl an falsch identifizierten Tupeln reduziert werden und ungewisse Tupel können genauer untersucht werden.

Effizienzprobleme beim Bilden von Paaren können wie erwähnt durch Partitionierungs- und Blockingverfahren gelöst werden. Effizienz ist sehr systemspezifisch und evtl. muss auf Effektivität verzichtet werden, um die Effizienz zu steigern (z.B. Verwendung eines simpleren Ähnlichkeitsmaßes).

Zusammenfassend lässt sich sagen, dass alle identifizierten Probleme in der Literatur berücksichtigt wurden und es existieren robuste Lösungen für die Probleme.

VII. Probleme beim Lösen von Datenkonflikten

Das Lösen von Datenkonflikten an sich ist nicht schwer (z.B. einfach einen zufälligen Wert übernehmen). Schwierig ist die Wahl eines passenden Verfahrens, was den benötigten Anforderungen gerecht wird.

Beim Lösen von Datenkonflikten ergibt sich das Problem, dass falsche Informationen übernommen oder erzeugt werden können, da es oft sehr schwer oder sogar unmöglich ist, mit absoluter Sicherheit zu erkennen, welcher Wert der richtige ist. Vergleicht man z.B. das Tupel 1a aus Tabelle I und das Tupel 1b aus Tabelle II, lässt sich nicht ohne zusätzliche Informationen ermitteln, ob bei einem das Geburtsdatum falsch ist.

Eine weitere Herausforderung ergibt sich dadurch, dass im Anwendungskontext des D4A keine genauen Anwendungsfälle für die Daten vorliegen. Denn in einer Smart Region müssen die Daten teils für sehr unterschiedlichen Anwendungen bereitgestellt werden. Daher müssen bereitgestellten Daten an unterschiedliche Anforderungen anpassbar sein. Dies gilt auch für das Lösen von Datenkonflikten.

VIII. Lösunsansätze für das Lösen von Datenkonflikten

In diesem Abschnitt werden Lösungsansätze aus der Literatur für das Lösen von Datenkonflikten dargestellt.

Bleiholder 2006 et. al. [10] teilen Lösungsansätze zunächst in drei Kategorien auf:

a) Konflikt ignorieren: Hierunter fallen Strategien, die den Konflikt nicht lösen, sondern weiter an das nächste System geben.

b) Konflikt vermeiden: Konfliktvermeidungsstrategien nehmen wahr, dass ein Konflikt vorliegt und treffen eine schnelle Entscheidung darüber, welcher Wert übernommen wird, aber sie lösen den Konflikt nicht auf.

c) Konflikt lösen: Diese Strategien lösen den Konflikt. Hierbei werden die Werte der Tupel betrachtet und entweder aufgrund der Werte entschieden welcher übernommen wird oder ein neuer Wert wird generiert.

Daraufhin stellen J. Bleiholder 2006 et. al. [10] eine Reihe von Lösungsverfahren für Datenkonflikte vor. Diese werden im weiteren kurz beschrieben.

Konflikt weiterleiten: Diese Strategie leitet den Konflikt, ohne ihn zu handhaben, an das nächste System weiter, sie gehört zu Konflikt ignorierenden Strategien.

Alle Möglichkeiten betrachten: Bei dieser Strategie werden alle möglichen Attribut Kombinationen an das nächste System weitergeleitet. Diese Strategie ist eine Konflikt ignorierende Strategie.

Erste passende Information nehmen: Diese Strategie nimmt für jedes Attribut den ersten Wert, der nicht fehlt/*null* ist. Es ist eine Konflikt vermeidende Strategie.

Keine Datenkonflikte weiterleiten: Diese Strategie identifiziert Datenkonflikte und filtert sie raus, sodass nur fehlerfreie Daten weitergegeben werden. Hierbei handelt es sich um eine Konflikt vermeidende Strategie.

Daten der vertrauenswürdigsten Quelle übernehmen: Bei dieser Strategie werden Daten von vertrauenswürdigeren Quellen bevorzugt. Es handelt sich hierbei um eine Konflikt vermeidende Strategie.

Häufigsten Wert übernehmen: Bei dieser Strategie wird der Wert übernommen, der am häufigsten vorkommt. Es ist eine Konflikt lösende Strategie.

Zufälligen Wert übernehmen: Diese Strategie übernimmt einen zufälligen Wert. Es ist eine Konflikt lösende Strategie.

Durchschnitt berechnen: Bei dieser Strategie wird der Durchschnitt aus den vorhandenen Werten berechnet. Es handelt sich hierbei um eine Konflikt lösende Strategie.

Aktuellere Daten übernehmen: Diese Strategie übernimmt die aktuelleren Daten. Bleiholder 2006 et. al. [10] merken hierbei an, dass für diese Strategie ein Zeitstempel oder ähnliches benötigt wird, um die Aktualität der Daten zu bestimmen, allerdings existieren auch Verfahren die versuchen, aus den Daten zu ermitteln, welche aktueller sind, ohne das ein Zeitstempel benötigt wird. Eine solches Verfahren wird in W. Fan 2014 et. al. [11] vorgestellt. Diese Strategie ist Konflikt lösend.

Zusätzlich wurden einige weitere Strategien identifiziert, die den obigen nicht eindeutig zugewiesen werden können. Hierzu gehören:

Probabilistische Verfahren: Es gibt Ansätze in der Forschung Datenkonflikte über unsichere Werte mithilfe von Konzepten aus der Wahrscheinlichkeitstheorie aufzulösen (siehe z.B. [12]). Zusätzlich ist es denkbar, dass Ansätze wie die Fellegi-Sunter Methode (siehe Abschnitt V) für das Lösen von Datenkonflikten adaptiert werden können, auch wenn keine Ansätze dafür in der Literatur gefunden wurden.

Keine widersprüchlichen Daten übernehmen: Diese Strategie präferiert für ein Tupel Werte, die sich nicht gegenseitig widersprechen.

Kombination aus Strategien: Es ist ggf. möglich die hier erwähnten Lösungsstrategien miteinander zu kombinieren, um sich den benötigten Anforderungen anzupassen. Ein Beispiel hierfür ist die von W. Fan 2014 et. al. [11] vorgestellte Lösungsstrategie. Hierbei werden aktuellere und nicht widersprüchliche Daten bevorzugt. Dabei ist es von der konkreten Kombination der Lösungsstrategien abhängig ob es sich um eine Konflikt vermeidende, oder lösende Strategie handelt.

Domänenspezifische Strategien: In manchen Fällen kann es sinnvoll sein, domänenspezifische Strategien einzusetzen, welche versuchen, die Datenqualität durch den Einsatz von domänenspezifischem Wissen zu verbessern. Ein Beispiel hierfür ist das Zusammenführen von Regenwassermessungen, um Messungenauigkeiten auszugleichen (siehe [13]).

Im Anschluss stellt sich die Frage, wie ein geeignetes Verfahren ausgewählt werden kann. Bleiholder 2006 et. al. [10] geben vier Kriterien für die Auswahl eines geeigneten Verfahrens an, diese sind:

- *Systemverfügbarkeit:* Kann das verfügbare System die Strategie umsetzen?
- *Informationsverfügbarkeit:* Liegen die für das Verfahren benötigten Informationen vor?
- *Kostenbedürfnisse:* Habe ich genug Geld/Zeit/Speicher, um die Strategie umzusetzen?
- *Qualitätsbedürfnisse:* Wie viele Informationen soll das Ergebnis beinhalten? Wie fehlerfrei muss es sein?

Hierbei handelt es sich jedoch um die einzige Quelle, die identifiziert wurde, die sich mit der Auswahl eines geeigneten Verfahrens beschäftigt.

IX. Bewertung der Lösungsansätze für das Lösen von Datenkonflikten

In diesem Abschnitt werden die Lösungsstrategien für das Lösen von Datenkonflikten bewertet. Hierbei wird betrachtet, wie gut die in Abschnitt VII identifizierten Probleme gelöst werden, diese waren:

- Die Wahl eines geeigneten Verfahrens
- Das Erzeugen oder Übernehmen falscher Werte
- Die Anforderungsunspezifität im D4A Projekt.

Die Wahl eines geeigneten Verfahrens ist weiterhin schwierig. Die hier aufgelisteten Lösungsansätze für das Lösen von Datenkonflikten sind gewiss nur eine Teilmenge der möglichen Strategien. Zusätzlich lassen sich viele dieser Strategien auf unterschiedliche Weisen umsetzen (z.B. einen Zufälligen Wert wählen mit oder ohne Extremwerte). Dementsprechend gibt es sehr viele Verfahren für das Lösen von Datenkonflikten, was die Auswahl eines geeigneten Verfahrens erschwert.

Die Auswahlkriterien von Bleiholder 2006 et. al. [10] sind recht unpräzise und grob. Insbesondere die Frage, wie die Qualität einer Strategie bewertet werden kann, ist unbeantwortet. Es wurden keine Arbeiten gefunden, in denen die Qualität der Strategien miteinander verglichen wurden und auch wenn Strategien an realistischen Szenarien getestet sind, wurden selten Qualitätskriterien bei der Bewertung der Ergebnisse berücksichtigt.

Es liegen kaum Information vor, welchen Einfluss Strategien auf das Erzeugen oder Übernehmen falscher Werte haben, da die hier aufgelisteten Verfahren nur in wenigen Fällen anhand von realistischen Szenarien getestet sind. Zusätzlich fehlt auch hier wieder ein Vergleich mehrerer Verfahren zueinander.

Im D4A Projekt macht es Sinn, eine geeignete Lösungsstrategie je nach Anforderungen der jeweiligen Anwendung zu wählen, allerdings sind viele der Eigenschaften der Verfahren (wie z.B. die Qualität) nicht gut untersucht, daher ist es auch im Kontext des D4A Projektes schwierig zu entscheiden, welche Verfahren gut geeignet sind.

X. Konklusion

Diese Arbeit hat, nachdem der Begriff des Datenkonfliktes eingeführt wurde, existierende Lösungsansätze aus der Literatur für das Identifizieren und Lösen von Datenkonflikten zusammengefasst.

Zusammenfassend lässt sich sagen, dass die Forschungsabdeckung für das Identifizieren von Datenkonflikten gut ist. Es gibt viele robuste Verfahren, die gut erforscht sind. Alle hier identifizierten Probleme wurden in der Wissenschaft bereits behandelt.

Für das Lösen von Datenkonflikten ist die Forschungsabdeckung jedoch mangelhaft. Viele wichtige Aspekte der Lösungsstrategien sind noch unbekannt und insbesondere die Wahl eines geeigneten Verfahrens gestaltet sich schwierig. Bereiche in welchen die Forschung hier ausgebaut werden sollte sind insbesondere der Vergleich der hier identifizierten Verfahren und Bewerten ihrer Auswirkung auf die Datenqualität.

References

[1] European Data Protection Supervisor, “Interoperability,” 05.06.2023. [Online]. Available: https://edps.europa.eu/data-protection/our-work/subjects/interoperability_en

[2] atene KOM, “Data for All. Data-driven Public Service Delivery in the North Sea Region - atene KOM,” 23.05.2023. [Online]. Available: https://atenekom.eu/project/data-for-all-data-driven-public-service-delivery-in-the-north-sea-region/?lang=en

[3] J. Bleiholder and F. Naumann, “Data fusion,” *ACM Computing Surveys*, vol. 41, no. 1, pp. 1–41, 2009.

[4] K. Hildebrand, M. Gebauer, H. Hinrichs, and M. Mielke, Eds., *Daten- und Informationsqualität: Auf dem Weg zur Information Excellence*, 4th ed. Wiesbaden: Springer Vieweg, 2018. [Online]. Available: https://ebookcentral.proquest.com/lib/kxp/detail.action?docID=5428760

[5] ScienceDirect, “Principles of Data Integration,” 24.06.2023. [Online]. Available: https://www-sciencedirect-com.proxy02a.bis.uni-oldenburg.de/book/9780124160446/principles-of-data-integration?via=ihub=

[6] J. Asher, D. Resnick, J. Brite, R. Brackbill, and J. Cone, “An Introduction to Probabilistic Record Linkage with a Focus on Linkage Processing for WTC Registries,” *International journal of environmental research and public health*, vol. 17, no. 18, 2020.

[7] J. Bleiholder and J. Schmid, “Datenintegration und Deduplizierung, pages = 123–142, publisher = MORGAN KAUFMANN, isbn = 978-3-658-30990-9, editor = Mielke, Michael, booktitle = DATEN- UND INFORMATIONSQUALITT, year = 2020, address = [S.l.], doi = 10.1007/978-3-658-30991-6_7, file = Bleiholder, Schmid 2021 - Datenintegration und Deduplizierung:Attachments/Bleiholder, Schmid 2021 - Datenintegration und Deduplizierung.pdf:application/pdf.”

[8] J. Martinez-Gil, “CoTO: A novel approach for fuzzy aggregation of semantic similarity measures,” *Cognitive Systems Research*, vol. 40, pp. 8–17, 2016.

[9] I. P. Fellegi and A. B. Sunter, "A Theory for Record Linkage," *Journal of the American Statistical Association*, vol. 64, no. 328, p. 1183, 1969.
[10] J. Bleiholder and F. Naumann, *Conflict Handling Strategies in an Integrated Information System*, 2006. [Online]. Available: https://www.researchgate.net/publication/238121998_Conflict_Handling_Strategies_in_an_Integrated_Information_System
[11] W. Fan, F. Geerts, N. Tang, and W. Yu, "Conflict resolution with data currency and consistency," *Journal of Data and Information Quality*, vol. 5, no. 1-2, pp. 1–37, 2014.
[12] E.-P. Lim, J. Srivastava, and S. Shekhar, "An evidential reasoning approach to attribute value conflict resolution in database integration," *IEEE Transactions on Knowledge and Data Engineering*, vol. 8, no. 5, pp. 707–723, 1996.
[13] S. Ochoa-Rodriguez, L.-P. Wang, P. Willems, and C. Onof, "A Review of Radar–Rain Gauge Data Merging Methods and Their Potential for Urban Hydrological Applications," *Water Resources Research*, vol. 55, no. 8, pp. 6356–6391, 2019.

Rechtliche Rahmenbedingungen für die Verarbeitung smarter Daten

Malte Südema
Department für Informatik - Software Engineering
Carl von Ossietzky Universität Oldenburg
Oldenburg, Deutschland
malte.suedema@uol.de

***Abstract*—In dieser Arbeit wird zunächst die Relevanz der rechtlichen Rahmenbedingungen im Bereich der Softwareentwicklung aufgezeigt und das Data-For-All (D4A)-Projekt vorgestellt. Darauf folgt die Betrachtung der für diese Arbeit relevanten Rechtsgebiete. Zunächst wird die Urheberschaft der Daten näher beleuchtet, indem festgestellt wird, wann Daten überhaupt schutzwürdig sind. Zudem wird die Urheberschaft der Daten geklärt, welche durch Dritte in die D4A-Software eingespeist werden und wem die Urheberschaft der durch die D4A-Software erstellten rekombinierten Daten obliegt. In diesem Kontext werden auch mögliche Lösungsansätze bezüglich der Nutzung von Daten vorgestellt. Weiter werden Datenschutzaspekte untersucht, welche für die Nutzung von Daten von Bedeutung sind. Hierbei werden personen-bezogene und nicht-personen-bezogene Daten unterschieden und die mit den jeweiligen Daten einhergehenden Pflichten und Maßnahmen aufgezeigt. Abschließend erfolgt die Behandlung von Haftungsfragen. Dafür wird ein Überblick über die verschiedene Haftungsarten gegeben und mögliche Ausschlusskriterien behandelt. Abschließend zu jedem Bereich werden Handlungsempfehlungen für das D4A-Projekt gegeben, welche sich auf die Gegebenheiten des Projektes und der Bereiche beziehen, um die rechtlichen Rahmenbedingungen einzuhalten.**

***Schlüsselwörter*—Smart Region, Smart Data, Datenschutz, Speicherung von Daten, Datenverarbeitung, Urheberrecht, Haftung, AGB**

I. MOTIVATION

Die Strafen für Unternehmen bei Verletzung von Gesetzen, welche aufgrund von Softwarefehlern oder Datenschutzmängel verhängt werden, können bis zu 1,2 Milliarden Euro betragen [1]. Der kalifornische Internetgigant *Meta Platforms* ist dabei kein Einzelfall. Wie ein Statista Bericht [2] zeigt, wurden seit der Einführung der Datenschutz-Grundverordnung (DSGVO) ungefähr 1600 Strafzahlungen für insgesamt über 4 Milliarden Euro aus unterschiedlichsten Gründen veranlasst [3]. Nicht nur der Datenschutz spielt bei der Einhaltung von rechtlichen Rahmenbedingungen bei der Verwendung von Software eine Rolle, wie ein Beispiel des Autoherstellers BMW zeigt. BMW wurde aufgrund einer fahrlässigen Aufsichtspflichtverletzung, durch eine fehlerhaften Bedatung in der Motorensteuerungssoftware, zu einer Strafe von 8,5 Millionen Euro verurteilt [4].

Um diese Strafen zu verhindern, ist es notwendig bereits, vor der Entwicklung und der Inbetriebnahme der Software den rechtlichen Rahmen zu kennen und gesetzliche Vorgaben und Bestimmungen zu berücksichtigen. Hierbei spielen Gesetze aus dem öffentlichen Recht, Zivilrecht, Vertragsrecht und Deliktrecht eine maßgebliche Rolle. [5]

Diese Voraussetzungen gelten auch für das durch die europäische Union angestoßene Projekt **Data For All (D4A)** [6]. Bei dem Projekt, an dem 19 Städte der Nordseeregion beteiligt sind, sollen smarte heterogene Daten aus verschiedenen Quellen rund um das Themenfeld Smart Regions interoperabel gemacht werden. Hierfür ist es notwendig eine Software zu entwickeln, welche Daten empfängt und diese weiterverarbeitet, sodass die Daten gebündelt in einem neuen Schema nutzbar vorliegen. Um eine rechtskonforme Umsetzung zu gewährleisten, ist es erforderlich, entsprechende rechtliche Rahmenbedingungen für die eingespeisten Daten sowie die zusammengesetzten Daten zu eruieren und mögliche Lösungsstrategien zu erörtern.

II. PROBLEMBESCHREIBUNG

Im Rahmen des D4A-Projektes müssen insbesondere rechtliche Problemstellungen zu den Daten berücksichtigt werden. Hierbei spielen Fragestellungen bezüglich der Urheberschaft, des Datenschutzes und der Haftung eine wichtige Rolle.

Die Urheberschaft der Daten stellt in diesem Kontext ein wesentliches Problemfeld dar, da die Daten hinsichtlich ihres Ursprunges unterschiedlich betrachtet werden müssen. Hier muss zwischen den Daten von Dritten, welche der D4A-Software bereitgestellt werden, und den durch die D4A-Software rekombinierten Daten unterschieden werden. Besonders bei der Betrachtung der rekombinierten Daten ist eine klare Urheberschaft nicht direkt zu erkennen und bedarf einer tiefer gehenden Einarbeitung. Ebenso muss die Frage geklärt werden, auf welcher rechtlichen Grundlage Daten von Dritten oder durch Dritte verwendet werden dürfen.

Der zweite Aspekt betrifft den Datenschutz, da im Rahmen des D4A-Projektes personen-bezogene und nicht-personenbezogene sowie gemischte Daten verarbeitet werden können. Durch diese Unterscheidung werden auch unterschiedliche rechtliche Bedingungen an die Nutzung der Daten geknüpft, welche im Kontext des D4A-Projektes betrachtet werden müssen. Besonders die mit der Verarbeitung einhergehenden Pflichten in Hinsicht auf den Datenschutz sind zu erarbeiten.

Im zuletzt genannten Punkt, stehen Fragen bezüglich der Haftung im Vordergrund. In diesem Zusammenhang ist zu klären, welche Partei für Fehler haftbar gemacht werden

kann. Im Speziellen inwiefern das D4A-Projekt bei fehlerhaft eingespeisten Daten und bei fehlerhaften rekombinierten Daten haftbar ist und wie die Haftung rechtswirksam ausgeschlossen werden kann beziehungsweise ob dies überhaupt, über beispielsweise Allgemeine Geschäftsbedingungen (AGB) oder Terms of Service (ToS), möglich ist.

Diese einzelnen Bereiche können zur folgenden allgemeinen Forschungsfrage zusammengefasst werden.

Welche rechtlichen Rahmenbedingungen in Bezug auf die Eigentümerschaft und Speicherung der Daten, des Datenschutzes und der Haftung bestehen im Kontext des D4A-Projektes und durch welche Ansätze können diese eingehalten werden?

Diese soll im Zuge diese Arbeit erarbeitet und mögliche Lösungsansätze für die einzelnen Probleme skizziert werden.

III. GRUNDLAGEN

Im folgendem Kapitel wird zunächst das grundlegende Konzept des D4A-Projektes beziehungsweise der Software erklärt. Außerdem werden die wesentliche Konzepte des Rechtssystems in Deutschland erläutert, um ein allgemeines Verständnis über Rechtsnormen und Rechtsbereiche zu erhalten.

A. Data-for-All Konzept

Das D4A-Projekt möchte es ermöglichen, Daten aus Smart Regions interoperabel nutzbar zu machen. Hierfür ist eine Software vorgesehen, die heterogene Daten aus verschiedenen Quellen kombiniert. Dabei soll es keine Rolle spielen welche Daten bereitgestellt werden oder welche Form sie besitzen. Durch eine Schnittstelle soll es Dritten ermöglicht werden, Daten in die D4A-Software einzuspeisen und diese so der Software für die weitere Verarbeitung zur Verfügung zu stellen. Mögliche Bereitsteller von Daten können beispielsweise E-Roller Betreiber, Fahrradverleiher oder kommunale Data Hubs sein.

Die D4A-Software hat in der Folge die Aufgabe, die Daten je nach Anwendungsfall sinnvoll miteinander zu kombinieren und diese wiederum Dritten über eine Schnittstelle zur Verfügung zu stellen. So soll eine Interoperabilität der heterogenen Daten aus verschiedenen Datenquellen erzeugt werden.

B. Rechtsnormen

Als Rechtsnorm werden gesetzliche Regelungen bezeichnet, welche eine Vorschrift auf abstrakte Weise darstellt. Rechtsnormen folgen dadurch dem eigentlichen materiellen Gesetz, diese sind beispielsweise in Gesetzbüchern festgehalten, und beschreiben in der Regel einen Tatbestand und eine Rechtsfolge. Falls diese Wenn-Dann-Syntax nicht gegeben ist, kann auch eine Legaldefinition vorliegen. Diese beschreibt lediglich, ob eine Handlung erlaubt ist, ohne eine Rechtsfolge zu benennen. Allgemein lassen sich vier Typen voneinander abgrenzen: Verbot, Gebot, Erlaubnis und Freistellung. Das Verbot stellt eine Unterlassungspflicht dar. Dahingegen ist unter dem Gebot eine Handlungspflicht zu verstehen, also die Notwendigkeit eine bestimmte Handlung durchzuführen, wie die Ersthilfe bei Unfällen zu leisten. Die Erlaubnis beschreibt wiederum lediglich ein Handlungsrecht. Diese kennzeichnet sich dadurch, dass einer Person die Möglichkeit zur Handlung gegeben wird, jedoch diese nicht verpflichtend ist. Abschließend wurde die Freistellung genannt, welche ein Unterlassungsrecht einräumt. Diese verhält sich ähnlich zur Erlaubnis, da einer Person lediglich die Möglichkeit gegeben wird eine Handlung nicht zu tätigen.

C. Rechtsbereiche

Die deutsche Rechtsprechung teilt sich in zwei voneinander getrennte Rechtsbereiche auf. Zum einen in das öffentlichen Recht und zum anderen in das Privatrecht. Ersteres bezieht Bereiche wie das Strafrecht, Grundrechte und Steuerrechte mit ein. Dieser Rechtsbereich wird maßgeblich dadurch charakterisiert, dass eine Partei, Kläger oder Beklagter, der Staat ist. Außerdem werden Klagen einem Verwaltungsgericht vorgetragen und verhandelt. Dabei muss angemerkt werden, dass das Strafrecht formal unter das öffentliche Recht fällt. Allerdings wird während der Juristenausbildung in Deutschland dieser gesondert gelehrt, wodurch er oftmals als eigenständiger Rechtsbereich genannt wird.

Das Privatrecht stellt wiederum den anderen elementaren Rechtsbereich dar. Dieser charakterisiert sich dadurch, dass beide Konfliktparteien Privatpersonen sind. Ferner wird dieser Bereich in das allgemeine Privatrecht und das Sonderprivatrecht aufgeteilt. Ersterer wird oftmals mit dem bürgerlichen Recht synonym verwendet, allerdings ist das bürgerliche Gesetzbuch (BGB) Teil des allgemeinen Privatrechts. Das Sonderprivatrecht wird auch als Privatrecht der Handelsleute bezeichnet und umfasst eine Vielzahl an Gesetzbüchern, wie das Handelsgesetzbuch (HGB), Urhebergesetze, die Datenschutzgrundverordnung (DSGVO) und das Vertragsrecht.

Insbesondere der Bereich des Privatrechts spielt für die Bearbeitung der Forschungsfrage eine erhebliche Rolle.

IV. RECHTLICHE RAHMENBEDINGUNGEN

Im Folgenden werden die drei Themengebiete, welche sich aus der Problemstellung und der Forschungsfrage ergeben, näher untersucht. Dadurch wird ein profunder Einblick über die in dem Themen relevanten Fragestellungen erarbeitet. Zudem werden allgemeine Informationen zu den rechtlichen Rahmenbedingungen gegeben, um ein besseres Verständnis zu erhalten. Abschließend zu jedem Gebiet werden die Fragestellungen beantwortet und es werden mögliche Handlungsempfehlungen gegeben, welche für den Kontext des D4A-Projektes ausgelegt sind.

A. Urheberrecht

Das Urheberrecht schützt den Urheber in seinen geistigen und persönlichen Beziehungen zum Werk und in der Nutzung des Werkes (Art. 11 UrhG). Diesem Artikel liegen zweierlei Erkenntnisse zu Grunde. Zum einen muss der Urheber eine

geistige oder persönliche Beziehung zu einem Werk besitzen. Dies heißt, dass ein Werk in irgendeiner Form einen Schaffensprozess widerspiegelt. Zum anderem hat der Urheber zur Nutzung seines Werkes sogenannte Hoheitsrechte und kann somit Verwertungs- und Nutzungsrechte erteilen. [7]

Daraus erschließen sich die Fragestellungen, welche sich in diesem Kontext für das D4A-Projekt ergeben. Die Fragen wem die eingespeisten und wem die rekombinierten Daten gehören und wie diese Verwendet werden dürfen, stehen im Vordergrund.

Zur Beantwortung der Fragen muss zunächst betrachtet werden, ob Daten überhaupt unter das Urheberrecht fallen. Generell sind Daten urheberrechtlich schutzlos, da sie keine körperlichen Gegenstände nach Artikel 90 BGB sind und in der Regel nicht eine relevante Schöpfungshöhe erreicht worden ist [8]. Dies gilt für jegliche Daten, also auch für personen-bezogene-, Forschungs- sowie Geschäftsdaten. Jedoch sieht das Gesetz Ausnahmen für die Nutzung vor. Dadurch können schutzlose Daten durch Gesetze als schutzwürdig betrachtet werden. Als bekanntes Beispiel dienen personen-bezogene Daten, dessen Nutzung durch die DSGVO massiv eingeschränkt wird. Weitere Gesetze, wie das Geschäftsgeheimnisschutzgesetz (GeschGehG) und auch das Urheberrecht, können unter bestimmten Umständen die Nutzung einschränken. Nur rechtliche Vorgaben dienen zur Bestimmung, ob Daten urheberrechtlich schutzwürdig sind. Dadurch hat beispielsweise das Copyright-Symbol ”©” keinerlei rechtliche Wirksamkeit in Deutschland [9].

Werden jedoch Sammelwerke (Art. 4 Abs. 1 des UrhG) oder Datenbankwerke (Art.4 Abs. 2 UrhG) betrachtet, liegt ein Urheberrechtsschutz vor. Damit Daten in diese Definition fallen, müssen mehrere Daten oder Elemente zusammengefasst und angeordnet werden. Dadurch wird ein urheberrechtlich relevanter Grad an Individualität beziehungsweise geistiger Schöpfung erzeugt. Bei einem Datenbankwerk muss zudem die Möglichkeit bestehen, dass diese elektronisch zugänglich sind und die Daten einer Systematik oder Methodik folgen müssen [8], [9]. Wichtig ist hierbei jedoch, dass nicht die Daten selbst geschützt sind, sondern die Datenbank als Ganzes. So fallen einzelne Datenpunkte nicht unter das Urheberrecht, aber Daten, welche wesentliche Teile der Datenbank widerspiegeln, schon. Falls keine Merkmale einer geistigen Schöpfung vorliegen, kann eine Datenbank auch durch das Datenbankschutzgesetz (Art. 87a ff. UrhG) geschützt sein. Hierfür ist eine wesentliche Investition für den Betrieb der Datenbank zu tätigen (Art. 87a Abs. 2 UrhG) [8], [10]. Abschießend sind angereicherte Datensätze zu nennen. Diese enthalten bereits urheberrechtlich geschützte Werke, wie Musik oder Bilder, und sind infolgedessen ebenfalls urheberrechtlich schützenswert.

Da davon auszugehen ist, dass die Daten, welche durch Dritte eingespeist werden, von einer Datenbank stammen, sind sie urheberrechtlich geschützt. Wodurch sich die erste Fragestellung, wem die eingespeisten Daten gehören, beantwortet. Werden jedoch die rekombinierten Daten betrachtet, dann ist die Urheberschaft nicht klar erkenntlich. Die Rekombination der Daten durch das D4A-Projekt spiegelt eine geistige Schöpfung wider und ist somit auch urheberrechtlich schützenswert. Dadurch würden sich die Urheber der eingespeisten Daten und das D4A-Projekt sich die Urheberschaft teilen. Falls jedoch nur Daten verwendet werden, welche keinem Urheberrechtsschutz unterliegen, hätte D4A das alleinige Urheberrecht.

Um rechtliche Sicherheit für die Nutzung der Daten von Dritten, wie auch durch Dritte, zu erhalten, gibt es verschiedene Möglichkeiten. Nutzungsrechte können durch die einspeisenden Unternehmen vergeben werden. Dies würde jedoch eine vertragliche Vereinbarung mit jedem Unternehmen, welches potenziell Daten in das D4A-System einspeist, voraussetzen. Alternativ könnte durch das D4A-Projekt ein AGB oder ToS formuliert werden, welches die einspeisenden Unternehmen über weitere Nutzung der Daten durch das D4A-Projekt in Kenntnis setzt und sich dadurch die Nutzungsrechte der Daten zusichert. Dieses Modell wäre auch für die Nutzung der rekombinierten Daten durch Dritte sinnvoll, da so allgemein Dritten ermöglicht wird, die Daten zu nutzen. Im Falle der rekombinierten Daten besteht auch die Möglichkeit, dass die anderen Urheber ihre Verwertungsrechte an das D4A-Projekt abgeben. Dadurch wäre es einfacher, Nutzungsrechte für die rekombinierten Daten zu vergeben. Allerdings ist dieses Vorgehen eher unüblich. Eine Besonderheit stellt die Verwendung von Daten dar, welche unter das GeschGehG fallen, da hier neben einer Verschwiegenheitsverpflichtungserklärung auch Maßnahmen zur Wahrung der Geheimhaltung vorgenommen werden müssen.

B. Datenschutz

Durch die unterschiedlichen rechtlichen Rahmenbedingungen im Bezug auf die zu schützenden Daten, insbesondere durch die DSGVO, sind die personen-bezogenen und die nicht-personen-bezogenen Daten getrennt voneinander zu betrachten. [11] Die Trennung kann durch den Art. 4 Nr. 1 DSGVO veranschaulicht werden. Dieser beschreibt, dass personen-bezogene Daten alle Informationen sind, die sich auf eine identifizierte oder identifizierbare Person beziehen. Daraus ergibt sich, dass alle Daten, welche nicht unmittelbar zur Identifikation einer Person genutzt werden können oder auch anonymisierte Daten, nicht in der Nutzung durch die DSGVO eingeschränkt sind. Im Rahmen des D4A-Projektes ergeben sich aus der Trennung jedoch zweierlei Fragestellungen. Zum einen, wo die Grenze zwischen personen-bezogenen und nicht-personen-bezogenen Daten gezogen wird. Zum anderem, welche Pflichten bei der Verarbeitung von personen-bezogenen Daten betrachtet werden müssen und welche Partei bei nicht Einhaltung der Pflichten verantwortlich ist. Zur Beantwortung der Fragen werden die im folgendem betrachteten rechtlichen Rahmenbedingungen getrennt voneinander erarbeitet.

Durch die eingangs erwähnte Definition von personenbezogenen Daten, wird deutlich, dass zwischen Daten, welche eine Person direkt identifizieren oder durch die eine Person identifizierbar wird, unterschieden werden kann. Bei den direkt identifizierbaren Daten handelt es sich um eine Kombination aus Name, Geburtsdatum und Adresse, mit derer eine Person in der Regel genau zugeordnet werden kann. Informationen, welche eine Person identifizierbar machen, können hingegen unterschiedlichen Ursprungs sein. Darunter fallen beispielsweise Online-Kennungen, Standort- und Bewegungsdaten sowie auch Dienstpläne. Durch eine Kombination verschiedener Daten ist es somit möglich eine Person zu identifizieren, wodurch ein wesentliches Kriterium erfüllt ist, dass es sich bei den Daten um personen-bezogene Daten handelt und die Nutzung durch die DSGVO eingeschränkt wird.

Um personen-bezogene Daten verwenden zu dürfen, muss einerseits die Erlaubnis der betreffenden Person eingeholt werden und andererseits muss die Erhebung an einen Zweck gebunden sein. Dieser Zweck muss bereits vorher festgelegt, eindeutig und legitim sein (Art. 5 DSGVO). Dadurch wird sichergestellt, dass die Daten nicht zweckentfremdet werden oder zu nicht legalen Zwecken verwendet werden. Eine Weiterverarbeitung der Daten kann zwar erfolgen, muss aber mit dem ursprünglichen Zweck vereinbar sein, sodass eine Zweckkompatibilität zwischen der ursprünglichen Erhebung und der Weiterverarbeitung besteht. [12]

Der Gesetzgeber formuliert zudem Grundsätze, wie mit den Daten umgegangen werden muss. Durch den Grundsatz der Tranzparenz (Art. 15 Abs. 1 DSGVO) muss es der Person, deren Daten erhoben wurden, möglich sein, in Erfahrung zu bringen, welche Informationen genau gespeichert und zu welchem Zwecke diese erhoben wurden. Ferner ergibt sich auch, dass die betroffene Person das Recht auf Veränderung und Löschung der gespeicherten Daten hat (Art. 16 & 17 DSGVO). Zudem dürfen auch nur für den Zweck relevante Daten erhoben werden, dies wird durch den Grundsatz der Datenminimierung beschrieben (Art. 5 Abs. 1c DSGVO). Die Speicherbegrenzung schreibt zudem vor, dass die Daten lediglich solange gespeichert werden dürfen, solange der Zweck nicht erfüllt ist (Art. 5 Abs. 1e). Ein weiterer relevanter Grundsatz bezieht sich auf die Integrität und Vertraulichkeit (Art. 5 Abs. 1f DSGVO), welcher die Pflicht beschreibt, die Daten gesichert zu speichern und zu schützen. Die Grundsätze der Richtigkeit und Rechenschaftspflicht sind zwar zu nennen, allerdings für den Kontext des D4A-Projektes von untergeordneter Wichtigkeit, da diese für die Umsetzung geringfügige Auswirkungen haben.

Durch die sichere Speicherung der Daten sind Schutzmaßnahmen zu treffen, welche durch eine vorherige Schutzbedarfsanalyse identifiziert werden müssen. Die DSGVO beschreibt, dass diese dem aktuellem Stand der Technik entsprechen müssen (Art. 32 Abs. 1 DSGVO, Art. 13 Abs. 7 Telemediengesetz). Dieser Stand ist jedoch nicht genau definiert und umschreibt die wirkungsvollste am Markt verfügbare Technik für IT-Sicherheit [13]. Der Bundesverband IT-Sicherheit e.V. (TeleTrusT) gibt jedoch einen jährlichen Bericht heraus, der einen objektiven technischen Überblick über aktuelle Technik gibt, der als Argumentationsgrundlage dienen kann. Jedoch ist auch eine Unterschreitung des aktuellen Standes der Technik möglich, wenn subjektive Gründe nachvollziehbar vorgelegt werden können. Darunter fallen monetäre Gründe, wie etwa dass die Implementierungskosten nicht im Verhältnis zum Projekt stehen oder, dass wirtschaftliche Ressourcen begrenzt sind (Art. 32 DSGVO).

Für die Feststellung der Verantwortlichkeit sieht der Gesetzgeber drei Modelle vor. Die Auftragsverarbeitung (Art. 28 DSGVO), eine gemeinsame Verantwortung (Art. 26 DSGVO) und die getrennte Verantwortung (Art. 24 Abs. 1 DSGVO). Erstere bezieht sich darauf, dass dieser die Daten lediglich im Auftrag weiterverarbeitet, sodass dieser von jeglicher Verantwortung ausgeschlossen ist. Bei der gemeinsamen Verantwortung wird jedoch davon ausgegangen, dass die Parteien einen gemeinsamen Zweck verfolgen und auch gemeinsam die Mittel festlegen, um den Zweck zu erreichen. Bei diesem Modell, welches auch als Joint Controllership bezeichnet wird, teilen sich alle Parteien die Verantwortung und es Bedarf einer eigenen Rechtsgrundlage. Bei dem letzten Modell wird wiederum entweder nur ein gemeinsamer Zweck oder nur die Mittel oder keines von Beidem festgelegt. Dadurch ist jede Partei für sich selbst verantwortlich.

Im Gegensatz zu den personen-bezogenen Daten ist die Verarbeitung von nicht-personen-bezogenen Daten wesentlich weniger reglementiert oder mit Pflichten versehen. In diesem Falle wird die Nutzung der Daten nicht durch die DSGVO eingeschränkt und können, wie in Unterabschnitt IV-A bereits beschrieben, als schutzlos angesehen werden. Durch eine Anonymisierung der personen-bezogenen Daten besteht außerdem die Möglichkeit, die Daten nachträglich in diesen Zustand zu konvertieren. Insbesondere die Zweckbindung und die Zustimmung der betreffenden Personen, auch zur Weiterverarbeitung, entfällt somit. Allerdings sind weitere Ausnahmen zu betrachtet. Denn Daten unterliegen trotzdem zumeist einem urheberrechtlichen Schutz, da es sich, wie bereits erwähnt, um Sammelwerke beziehungsweise Datenbankwerke handelt. Zudem sind bei Daten, welche der Geheimhaltung unterliegen, zusätzliche Schutzmaßnahmen nach aktuellem Stand der Technik vorzunehmen.

Abschießend müssen auch gemischte Datensätze betrachtet werden, da es sich hierbei um eine häufige Darstellung von Daten handelt. Darunter fallen Datensätze, wie Steuerregistereinträge, welche Firmenanschrift und Namen des Inhabers beinhalten, oder auch Daten, welche durch Internet of Things (IoT)-Geräte erhoben werden, da hier durch Standortdaten oder Nutzungsmuster auf Personen rückgeschlossen werden kann. Handelt es sich um einen gemischten Datensatz, kann, laut EU-Verordnung 2018/1807, auf drei unterschiedliche Arten damit umgegangen werden. Falls sich die Daten in dem Datensatz zwischen personenbezogenen Daten und nicht personen-bezogenen Daten trennen lassen, dann gelten die DSGVO Bestimmungen für personenbezogene Daten und für die nicht-personen-bezogenen Daten

gelten die allgemeinen Bestimmungen. Sollten die Daten innerhalb des Datensatzes jedoch untrennbar miteinander verbunden sein, dann greifen auf den gesamten Datensatz die DSGVO Bestimmungen.

Durch die vorherigen Ausführungen lässt sich auch die Fragestellung beantworten, ob es zwischen den Daten unterschiedliche rechtliche Rahmenbedingungen gibt. Durchaus müssen verschiedene Aspekte berücksichtigt werden. Zunächst muss ausgemacht werden, ob es sich um personen-bezogene oder nicht-personen-bezogene Daten handelt, welche verschiedenen Restriktionen unterliegen. Im Falle von gemischten Datensätzen muss wiederum geprüft werden, ob sich diese in die genannten Daten aufteilen lassen oder die Daten unzertrennlich sind. Die zweite Fragestellung, die zu Beginn des Abschnittes formuliert wurde, kann unter Berücksichtigung mehrerer Aspekte beantwortet werden. Um die Anforderungen zur Verarbeitung von personen-bezogenen Daten zu erfüllen, müssen eine Vielzahl an Sicherheitsmaßnahmen erfüllt werden, welche zudem dem aktuellen Stand der Technik entsprechen müssen. Ebenso ist es notwendig die Grundsätze in Bezug auf Speicherung, Zweckbindung und Auskunft zu erfüllen. Abschließend müssen die Daten zudem löschbar sein, da der Gesetzgeber jeder natürlichen Person das Recht auf vergessen werden einräumt. Welches allerdings auch rückwirkend auf anonymisierte Daten geltend ist. Das bedeutet, dass auch Daten, die nicht unter die DSGVO fallen, löschbar sein müssen.

Um die rechtlichen Notwendigkeiten richtig einzuschätzen, ist es notwendig seitens des D4A-Projektes frühzeitig einzuschätzen, ob personen-bezogene Daten verwendet werden. Im Falle einer Verwendung personen-bezogener Daten ist es sinnvoll, entsprechende Sicherheitsmaßnahmen ordentlich zu dokumentieren, sodass diese klar nachzuvollziehen sind. Zur rechtlichen Absicherung kann auch ein Datenschutzsiegel (Art. 42 DSGVO) oder ein zukünftiges europäisches Datenschutzsiegel in Betracht gezogen werden [14]. Ebenso müssen sich die beteiligten Parteien auf ein Verantwortungsmodell einigen und ggf. eine Joint-Controllership- wie auch eine Auftragsverarbeitungs-Vereinbarung treffen. Unabhängig von der Art der Daten sollte zudem ein Löschkonzept entwickelt werden, um das Löschen von einzelnen Datenpunkten zu ermöglichen. Grundsätzlich ist die Nutzung von personenbezogenen Daten im Rahmen des D4A-Projektes weniger empfehlenswert, da damit wesentlich mehr Pflichten und Anforderungen einher gehen.

C. Haftung

Der letzte Aspekt der in der Forschungsfrage ausgemachten Kernthematiken, bezieht sich auf Fragen im Bereich der Haftung. Im Vordergrund steht dabei, welche Partei für Fehler und dadurch entstandene Schäden verantwortlich gemacht werden kann. Diese Fragestellung lässt sich weiter spezifizieren. Daraus folgen die Fragen, ob das D4A-Projekt für Fehler in den eingespeisten Daten haftbar ist und ob das D4A-Projekt für mögliche Fehler, welche während der Rekombination der Daten entstanden sind, haftbar gemacht werden kann.

Im Wesentlichen ist zwischen der strafrechtlichen Haftung, welche auch mit Freiheitsstrafen geahnten werden kann sowie keinen Ausgleich für das Opfer vorsieht, und zivilrechtlicher Haftung, welche in der Regel mit einem finanziellen Ausgleich bestraft wird, zu unterscheiden. Im Falle des D4A-Kontextes steht die zivilrechtliche Haftung im Fokus. In dieser sind wiederum eine Vielzahl an verschiedenen Haftungsarten, wie die vertragliche, deliktische Haftung oder die Produzentenhaftung, enthalten.

Falls zwischen zwei Parteien ein Vertragsverhältnis besteht, greift die vertragliche Haftung. In diesem Fall werden natürlich Verstöße gegen die in dem Vertrag festgelegten Hauptpflichten geahndet. Der Schadensausgleich kann in diesem Fall bereits direkt vertraglich festgelegt oder durch eine allgemeine Regelung (Art. 280 BGB) festgelegt sein. Jedoch kann es auch zur Verletzung sogenannter Nebenpflichten kommen. Diese Pflichten sind nicht explizit im Vertrag genannt, sind aber nach gutem Glauben Teil der Vereinbarung. Darunter fallen das Aufklären über Risiken, Verhinderung von Missbrauch, aber auch Leistungen, welche für zur Erbringung der Hauptpflichten unabdingbar sind oder eine Üblichkeit der Branche darstellen. Besonders die Nebenpflichten sind im Rahmen des D4A-Projektes schwer zu identifizieren, da es sich hierbei um ein neuartiges Geschäftsmodell handelt und somit keine branchenspezifischen Üblichkeiten auszumachen sind. An dieser Stelle ist auch der Haftungsausschluss über eine AGB zu nennen. Durch diese können zwar viele Haftungsgründe ausgeschlossen werden, aber der Ausschluss ist nicht allumfassend möglich. Der Gesetzgeber sieht die AGB bei der Verletzung von Klauselverboten (Art. 308 & 309 BGB) als unwirksam an. Hierin ist beispielsweise der Haftungsausschluss bei Verletzung von Leben, Körper, Gesundheit und bei grobem Verschulden (Art. 309 Abs. 7 BGB) ausgeschlossen. Zudem ist anzufügen, dass bei Kollaborationen und Kooperationen der Vertrauensgrundsatz wirksam ist. Darunter wird verstanden, dass Vertragsparteien mögliche Fehler anderer Vertragsparteien im vornherein nicht mit einbeziehen müssen (Art. 705 ff. BGB). Dadurch ist das D4A-Projekt nicht dazu verpflichtet, die Richtigkeit der eingespeisten Daten zu überprüfen. Diese wird jedoch unwirksam, wenn eine der Parteien darüber in Kenntnis gesetzt wird, dass es mehrfach zu Fehlern gekommen ist.

Konträr zur vertraglichen Haftung steht die deliktische Haftung, da hier Haftungsansprüche bestehen, auch wenn kein vertragliches Verhältnis besteht. Allerdings ist ein Schadensersatz in diesem Falle nur notwendig, wenn rechtswidrig oder schuldhaft seitens einer Partei gehandelt worden ist. Auch eine Haftung ist ausgeschlossen, wenn die geschädigte Partei gegen gute Sitten (Art. 138 & 242 BGB) gehandelt hat, sodass nicht mit einem Schaden gerechnet werden konnte.

Werden die aufgekommenen Fragestellungen betrachtet,

lassen sich diese zumeist nur im Einzelfall klären. Oft spielen vielerlei externe Faktoren und verschiedene gesetzliche Vorgaben ein Rolle. Allerdings kann generell festgestellt werden, dass jede Partei zunächst für seinen eigenen Beitrag an einem Schaden haftbar gemacht werden kann. Im Kontext des D4A-Projektes bedeutet dies, dass bei Daten, welche bereits bei Einspeisung falsch sind, die einspeisende Partei haftbar gemacht wird. Im Falle der rekombinierten Daten ist das D4A-Projekt im Wesentlichen mit haftbar.

Allerdings ist durch den Gesetzgeber im Rahmen des Produkthaftungsgesetzes (ProdHaftG) eine Ausnahme für nicht wirtschaftliche Nutzung vorgesehen (Art. 1 Abs. 2 Nr. 3 ProdHaftG). Unter der Voraussetzung, dass kein Gewinn durch die Software erwirtschaftet wird, ist die Haftung von möglichen Schäden ausgeschlossen. Dies wäre eine Möglichkeit, wodurch sich das D4A-Projekt gegenüber möglichen Schadensansprüchen absichern kann, da es sich hierbei um ein Forschungsprojekt handelt. Zudem lassen sich durch vertragliche Übereinkünfte ein Großteil der Haftung ausschließen. Hierfür wären Nutzungsverträge und die AGB ein probates Mittel. Außerdem können im Rahmen zivilrechtlicher Haftung auch Versicherungen abgeschlossen werden, um finanzielle Schadensausgleiche abzudecken. Allgemein sollten Fahrlässigkeiten durch die Einhaltung von Sorgfaltspflichten verhindert werden. Ein Ausgangspunkt hierfür sind Verkehrsgepflogenheiten und ISO- wie auch DIN-Normen.

V. KONKLUSION

In dieser Arbeit konnten die wesentlichen Aspekte der Forschungsfrage, welche in Abschnitt II formuliert wurde, erarbeitet werden. Eine vollständige Beantwortung ist im Rahmen der Ausarbeitung jedoch nicht abschließend möglich gewesen, da oftmals eine Einzelfallbetrachtung vorgenommen werden muss. Allerdings wurde ein Überblick über die Themenbereiche gegeben, sodass auf Grundlage dieser Arbeit eine tiefere Einarbeitung möglich ist.

So wurden im Unterabschnitt IV-A die unterschiedlichen Bedingungen im Bezug auf die Urheberschaft gegeben. Dadurch wurde dargestellt, dass Daten im Allgemeinen nicht schutzwürdig sind, aber zusammengefasst in einem Datenbankwerk dem Urherberrecht unterliegen. Die Einschätzung, dass die Daten, welche von Dritten eingespeist werden, Nutzungsrechte durch die Drittpartei benötigen, konnte gegeben werden. Die Nutzung der Daten kann durch Verträge festgelegt werden. Dies gilt wiederum auch für die durch die D4A-Software bereitgestellten rekombinierten Daten gegenüber Dritten, da das D4A-Projekt Miturheber für diese ist.

Im Bereich des Datenschutzes, welches in Unterabschnitt IV-B behandelt wurde, konnte eine Vielzahl an Pflichten bei der Verarbeitung von personen-bezogenen Daten vorgestellt werden und wie sich diese von den Pflichten der nicht-personen-bezogenen Daten unterscheiden. Dabei wurde festgestellt, dass sich die ausschließliche Verwendung von nicht-personen-bezogenen Daten, wie auch anonymisierten Daten, im D4A-Kontext erstrebenswert ist, um die etwaigen genannten Pflichten, welche durch die DSGVO vorgegeben sind, nicht betrachten zu müssen. Darüber hinaus wurden die Pflichten, wie die Löschung von Daten, welche sich unabhängig von der Datenart ergeben, dargestellt.

Schlussendlich wurden in Unterabschnitt IV-C die hauptsächlichen Felder der Haftung betrachtet. Dabei wurde herausgearbeitet, dass sich vielerlei Haftungsgründe durch die Verwendung von AGBs oder durch Nutzungverträge ausschließen lassen oder Schadensansprüche durch andere Mittel, wie Versicherungen, abgedeckt werden können. Zudem wurde dargelegt, dass jegliche Haftung ausgeschlossen ist, sobald das D4A-Projekt keine Gewinne erzielt.

VI. HAFTUNGSAUSSCHLUSS

Diese Arbeit dient in keiner Weise als Rechtsberatung, sondern soll lediglich einen Überblick über die rechtlichen Gegebenheiten in den Thematiken der Urheberschaft, des Datenschutzes und der Haftung in Bezug auf das D4A-Projekt geben. Für rechtsgültige Aussagen ist die Auskunft eines Rechtsanwaltes einzuholen.

REFERENCES

[1] P. Halpin, “Meta hit by record 1.2 BLN Euro fine by EU over U.S. data transfers,” May 2023. [Online]. Available: https://www.reuters.com/technology/meta-hit-by-record-12-bln-euro-fine-by-eu-over-us-data-transfers-2023-05-22/

[2] F. Zandt and M. Brandt, “Infografik: DSGVO-Bußgelder erreichen 2023 neues Rekordhoch,” May 2023. [Online]. Available: https://de.statista.com/infografik/26629/strafen-auf-grund-von-verstoessen-gegen-die-datenschutz-grundverordnung/

[3] M. Holzhofer, “DSGVO Bußgeld Datenbank - immer aktuell und vollständig.” [Online]. Available: https://www.dsgvo-portal.de/dsgvo-bussgeld-datenbank/

[4] Leiding, “Bußgeld über 8,5 Millionen Euro gegen die BMW AG,” Feb 2019. [Online]. Available: https://www.justiz.bayern.de/gerichte-und-behoerden/staatsanwaltschaft/muenchen-1/presse/2019/03.php

[5] G. Spindler, “Verantwortlichkeiten von IT-Herstellern, Nutzern und Intermediären - BSI,” 2020. [Online]. Available: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/Studien/ITSicherheitUndRecht/Gutachten_pdf.pdf?__blob=publicationFile&v=4

[6] “Data for All.” [Online]. Available: https://www.interregnorthsea.eu/dataforall

[7] M. Ambros, “Verwertungsrechte des Schöpfers - Urheberrecht,” Jul 2023. [Online]. Available: https://www.urheberrecht.de/verwertungsrechte/

[8] D. Ewers, “FAQ Datenhoheit.” [Online]. Available: https://www.digitale-technologien.de/DT/Redaktion/DE/Standardartikel/FAQ-Recht/datenhoheit.html

[9] Bundesministerium für Bildung und Forschung, “Daten, Fakten und Ideen - BMBF Digitale Zukunft.” [Online]. Available: https://www.bildung-forschung.digital/digitalezukunft/de/wissen/urheberrecht/urheberrecht-in-der-wissenschaft/daten-fakten-und-ideen/daten-fakten-und-ideen_node.html

[10] T. Kreutzer and H. Lahmann, “Rechte an Forschungsdaten und Datenbanken,” Mar 2023. [Online]. Available: https://irights.info/artikel/rechte-an-forschungsdaten-und-datenbanken/29587

[11] Europäische Union, “Speicherung und Verarbeitung von Daten in Europa: Freier Verkehr nicht personenbezogener Daten,” Sep 2022. [Online]. Available: https://europa.eu/youreurope/business/running-business/developing-business/free-flow-non-personal-data/index_de.htm

[12] M. Lindner, S. Straub, and B. Kühne, *How to Share Data? Data-Sharing-Plattformen für Unternehmen*. Institut für Innovation und Technik, 2021.
[13] K. Bartels, S. Beck, M. Bürger, S. Straub, and B. Buchholz, "Kollaborative Wertschöpfungssysteme in der Industrie," Aug 2020. [Online]. Available: https://www.digitale-technologien.de/DT/Redaktion/DE/Downloads/Publikation/PAiCE_Leitfaden_Recht-und-Gesch%C3%A4ftsmodelle.html
[14] S. Bedürftig, "Der Weg zum Europäischen Datenschutzsiegel"," May 2021. [Online]. Available: https://www.datenschutz-notizen.de/der-weg-zum-europaeischen-datenschutzsiegel-3228776/