Abschlussbericht

der Projektgruppe RAPID

Version zur Veröffentlichung

Themensteller:	Prof. Dr.-Ing. Jorge Marx Gómez
Betreuer:	Dipl.-Inf. Manuel Osmers, M. Sc. Alexander Sandau, M.Sc. Daniel Stamer, Dr. Dipl.-Oec. Benjamin Wagner vom Berg
Vorgelegt von:	Kai Hänig, Christian Janßen, Philipp Schumacher, Olga Schwarz, Jannes Spekker, Kamiran Tizyani, Nils Worzyk,
Abgabetermin:	15. Oktober 2015

Danksagung

In diesem Abschnitt dankt die Projektgruppe allen denjenigen, die einen Beitrag zum Projekt geleistet haben. An erster Stelle sei allen Beteiligten der Abteilung VLBA gedankt, die dieses Thema initiiert haben. Dazu ist insbesondere Professor Dr. Marx Gómez zu zählen. Darüber sind auch die Betreuer Daniel Stamer, Alexander Sandau, Benjamin Wagner vom Berg und Manuel Osmers zu zählen. Sie haben jede Woche in den Sitzungen den aktuellen Fortschritt des Projektes aufgenommen, haben stets mit Vorschlägen und Hilfestellungen unterstützend eingegriffen und insbesondere im Zuge der Datenbeschaffung für das Zusammenkommen von Treffen zwischen der Gruppe mit anderen Parteien gesorgt. Auch der Verkehrsleitzentrale Oldenburg, dem Statistischen Bundesamt und dem DLR Braunschweig sei an dieser Stelle für ihre Kooperation gedankt. Weiterhin ist der Universität Magdeburg für ihre Kooperation und der Bereitstellung der SAP HANA Appliance zu danken.

Projektgruppe RAPID im Oktober 2015

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einleitung

1.1 Motivation

Beim Straßenverkehr handelt es sich um ein hochkomplexes und kompliziertes System. „Er beinhaltet die Bewegung von Personen, Waren und Informationen in einem zeitlich und räumlich begrenzten Verkehrsgebiet“ [Erl](S.5). Im Zuge des Verkehrsmanagements soll der Verkehr optimiert und gesteuert werden [Lem13] (S.3). Zudem soll es möglich sein auf Grundlage der Ergebnisse Maßnahmen einzuleiten, die die Qualität des Verkehrs verbessern. Um einen Eindruck über die Verkehrslage zu erhalten, ist die Betrachtung einer Vielzahl von Daten notwendig. Diese Daten stammen von unterschiedlichen Quellen (Induktionssensoren auf den Straßen, Infrarotkameras auf den Ampeln, etc.). Dadurch, dass von den Verkehrsteilnehmern immer mehr Endgeräte wie z.B. Headsets, Smartphones, Navigationssysteme, etc. verwendet werden, sind auch die von ihnen erzeugten Daten, in der Analyse zu berücksichtigen. Darüber hinaus werden diese ergänzt mit Daten allgemeinerer Bedeutung (Wetterdaten, Veranstaltungsdaten, etc.) [Lem13] (S.3)angereichert. Durch das Einbeziehen vieler dieser Dimensionen kann es in der Analyse zu Performanceproblemen kommen [AK11](S. 272 ff). Deshalb werden im OLAP-Kontext (Online Analytical Processing) die zu analysierenden Daten in ein Data Warehouse geladen und dort in Form spezieller Schemata aufbereitet (vgl. [AB09](161 ff.). Diese Schemata tragen dazu bei, dass die komplexe multidimensionale Betrachtung der Daten, auf Basis festgelegter Kennzahlen wesentlich einfacher und schneller verläuft (vgl. [AB09](161 ff.)). Da allerdings die Datenberge im Zeitalter von Big Data immer weiter wachsen und darüber hinaus auch die Anforderungen an die Analysen steigen, stößt auch diese Vorgehensweise schnell an ihre Grenzen [int] (S. 2 f). Eine weitere Schwierigkeit die sich ergibt ist, dass zunehmend semi- bzw. unstrukturierte Daten in die Analysen einfließen. Diese Daten sind an kein spezielles Schema gebunden und gelangen häufig über Sensoren und andere Messgeräte, in die Datenbanken. Diese Daten treten oft in Form von Data-Streams auf. Durch deren Einbezug in die Analyse verkompliziert sich diese weiterhin auf Kosten der Performance [int]. Eine Lösung hinsichtlich der Problematik der steigenden Performance-Anforderung ist die Verarbeitung aller zu analysierenden Daten im Arbeitsspeicher durch In-Memory Computing. Auf diese Weise ist der Bezug der Daten direkt aus den transaktionalen Datenbanken möglich, ohne dass es erforderlich ist, die zu analysierenden Teilausschnitte der Datenlandschaft in spezielle OLAP-Schemata aufzubereiten. Durch das Laden aller zur Analyse verwendeten Daten in den flüchtigen Arbeitsspeicher, beschleunigt sich deren Verarbeitung oft um mehr als das 1000-fache [reg](S. 3 ff.). Durch das In-Memory Computing ergibt sich dementsprechend ein großer Paradigmenwechsel hinsichtlich der Analyse von Daten. Durch den Einbezug weiterer Technologien, kann eine Infrastruktur für die Analyse von Verkehrsdaten realisiert werden.

1.2 Problemstellung und Zielsetzung

Die Projektgruppe RAPID ist eine Gruppe von Studenten der Carl von Ossietzky Universität Oldenburg, die im Zuge einer Projektgruppenveranstaltung in Zusammenarbeit mit

Betreuern aus der VLBA-Abteilung (Very Large Business Applications) an der Realisierung einer Analyseplattform für Verkehrsdaten arbeitet. RAPID steht für Regional Analysis and Prediction Platform by In-Memory Data. Dementsprechend sollen verschiedene verkehrsbezogene Daten auf diese Plattform geladen, integriert und mittels In Memory-Technologie verarbeitet werden. Eingesetztes Produkt ist die HANA Appliance von SAP. Dieses Produkt wird auf dem Server der Otto-von-Guericke Universität Magdeburg bereitgestellt. Durch ein Eclipse-PlugIn ist die Arbeit damit möglich.

Mit Hilfe weiterer Technologien soll eine Analyse-Plattform für Verkehrsdaten mit dem Fokus auf den Oldenburger Raum realisiert werden. Diese soll einer Vielzahl von Benutzern mit unterschiedlichen Kenntnissen und Interessen einen einheitlichen Zugangspunkt bieten und die Funktionen kapseln, die sie für ihre Tätigkeiten benötigen. Dazu werden die unterschiedlichen Anforderungen der Benutzer hinsichtlich der Analyse ermittelt. Vor diesem Hintergrund muss der Wissenstand der Benutzer hinsichtlich der Analyse ermittelt und berücksichtigt werden. Handelt es sich etwa um einen Analysten oder um eine Person, die weniger versiert in der Datenanalyse ist und keine Erfahrung in den Gebieten wie z.B. Data Mining, Statistik oder IT im Allgemeinen hat.

Um den unterschiedlichen Kenntnissen der Benutzer in angemessener Weise Rechnung zu tragen, muss zunächst ermittelt werden, in welchen Darstellungsformen die Ergebnisse im Frontend präsentiert werden. Später sollen den Benutzern auf Grundlage dessen, die Möglichkeit gegeben werden durch die Verwirklichung interaktiver Darstellungen diese intuitiv zu erforschen, so dass es möglich ist die Ergebnisse aus anderen „Blickwinkeln“ zu betrachten. Soweit es möglich ist sollen die Ergebnisse möglichst in Echtzeit angezeigt werden.

Durch die Plattform soll es dem Analysten möglich sein, ein schnelles Bild über die Verkehrssituation in Oldenburg zu erhalten. Darüber hinaus soll es möglich sein Maßnahmen abzuleiten, die Vorgänge im Verkehr in Oldenburg in die richtigen Bahnen lenken. Dazu ist die Erfassung der Daten des Status Quo erforderlich. Beispielsweise ist es wichtig Besonderheiten aufzugreifen, wie z.B. dass in Oldenburg bei 42,7 der Verkehrsteilnehmer um Fahrradfahrer handelt, was u.a. durch den relativ hohen Anteil an jungen Menschen in Oldenburg (insbesondere Studenten) zu begründen ist [fie](S. 5). Mit Hilfe bestimmter statistischer Kennzahlen, kann ein erster Eindruck über bestimmte Abhängigkeiten in ihnen erfolgen. Mit dem HANA Studio von SAP werden diese Daten so aufbereitet, dass diese einheitlich zusammengeführt und hinsichtlich Redundanzen, Inkonsistenzen, Ausreißer-Werten etc. bereinigt werden. Die entstehenden Tabellen werden dann den Analyseverfahren als Input bereitgestellt. Die Ergebnisse werden dann vom Frontend abgefragt und in den entsprechenden Darstellungsformen angezeigt.

1.3 Aufbau der Dokumentation

Einleitend wird im ersten Kapitel der Dokumentation das Projektmanagement beschrieben. Hier wird der Leser vorerst in die Ziele und das Konzept der Projektgruppe eingeleitet. Darüber hinaus werden die unterschiedlichen Projektphasen inklusive Phasenmodell und Gantt-Diagramm im Kapitel 2.2 aufgezeigt. Das Kapitel Projektmanagement wird mit der Beschreibung der CRISP-Data Mining und SCRUMP Modell abgeschlossen. Das zweite

Kapitel schließt fließend mit der Beschreibung der organisatorischen und technischen Rahmenbedingungen an. Zu den organisatorischen Rahmenbedingungen gehören zum einen die Vorstellung der Projektmitglieder und ihre Funktionen in der Projektgruppe und zum anderen die Beschreibung der Wöchentlichen Ablaufplanung sowie der Kommunikation untereinander, mit den Betreuern und strategischen Partnern. Die Technischen Rahmenbedingungen umfassen alle wesentlichen Programme und Serververwaltungen, die im Laufe des Projektes zum Erreichen der Ziele verwendet wurden. Hier zu zählt: die Nutzung von Redmine für die Projektgruppenorganisation, der Server, ODBC und weitere Tools, diese werden im Kapitel 3.2 beschrieben. Eine ausführliche Erläuterung des Grundwissens für SAP HANA wird im Kapitel 4 gegeben, woraufhin im fünften Kapitel die Datenstruktur in einem ER-Diagramm dargestellt wird. Kapitel 6,7 und 8 beziehen sich auf die Verarbeitung der Daten. Zunächst wird hierfür das Datenverständnis beschrieben demnach folgt die Datenvorbereitung und abschließend die Modellierung wobei die Modellierung in kurzfristige und langfristige Prognose unterteilt wird. Im Kapitel 9 Darstellung der Ergebnisse werden vorerst die Anforderungen an das Webportal aufgeführt. Im nächsten Schritt wurden die daraus resultierenden Anwendungsfälle kenntlich gemacht und Kapitel 9.3 bildet das Anwendungsszenario ab. Darüber hinaus befasst sich das Kapitel mit dem Backend und dem Frontend für die Darstellung der Ergebnisse, hier wird genauer auf das Rollenkonzept, dem Layout des Portals und die Visualisierung eingegangen. Eine zusammenfassende Evaluation und der Ausblick schließen die Projektdokumentation ab.

2 Projektmanagement

In diesem Abschnitt wird das Projektmanagement genauer erläutert. In erster Linie wird das Konzept und die Ziele der Projektgruppe beschrieben. Zusätzlich wird auf den Projektverlauf, mit den einhergehenden Projektphasen und dem Gantt-Diagramm genauer eingegangen. Im letzten Abschnitt des Kapitels werden die Modelle 'CRISP-Data Mining' und 'SCRUM' vorgestellt.

2.1 Konzept und Ziel der Projektgruppe RAPID

Bei der Projektgruppe RAPID (Regional Analysis and Prediction Platform by In-Memory Data) handelt es sich um Masterstudenten der Wirtschaftsinformatik und Informatik der Universität Oldenburg, die sich mit der Verwirklichung einer Mobilitätsdatenplattform beschäftigen. Dieses wird in der Lehrveranstaltung Projektgruppe mit der Laufzeit von einem Jahr durchgeführt. In der Projektgruppe RAPID wird eine intelligente Plattform für Mobilitätsdaten entwickelt mit dem Zweck, deren Nutzer dazu zu befähigen, eine Region ökonomisch und ökologisch effizient im Verkehrsbereich zu erschließen und zu bewirtschaften.

2.2 Projektverlauf

Das Projekt unterteilte sich in drei wichtige Phasen, die im Folgenden genauer beschrieben werden.

2.2.1 Vorbereitungs- und Seminarphase

In der Vorbereitungsphase wurden zunächst verschiedene Aufgaben und Funktionen an die Projektmitgliedern vergeben. Hierbei wurden den einzelnen Projektgruppenteilnehmern die folgenden Aufgaben zugeteilt, welche respektiv über den gesamten Projektzeitraum übernommen werden:

- Christian Janßen - Serverdamin
- Nils Worzyk – Stellvertretender Serveradmin
- Jannes Spekker – Frontendentwicklung und -pflege
- Kai Hänig – Projektleiter und Kommunikation
- Kamiran Tizyani – Stellvertretender Projektleiter
- Olga Schwarz - Dokumentationsbeauftragte
- Phillip Schumacher - Finanzbeauftrager

Neben den zugewiesenen Aufgaben wurden die Projektgruppenteilnehmern zusätzlich noch entsprechenden Arbeitsgruppen zugewiesen, welche im Folgenden noch erörtert werden.Als organisationstechnische Sitzungsgliederung wurden ein Logo, eine Protokoll- sowie

eine Sitzungsvorlage erstellt. Weiterhin wurde in dieser Phase ein E-Mail-Verteiler, Server und eine Webseite eingerichtet, auf der die Ergebnisse der Projektgruppe dargestellt werden. Des Weiteren wurden in dieser Phase drei Interviewgruppen gebildet, welche den Aufgabensteller, in diesem Fall den Projektgruppenbetreuern, detaillierte Fragen zu den Themen Use-Case, Plattform und Analyse gestellt werden sollten, um einen konkreten Projektrahmen zu bilden (siehe Anhang 11). Darüber hinaus wurde eine Einführung in das SAP-Hana System anhand eines Use-Cases durchgeführt, sodass der Einstieg den Projektmitgliedern erleichtert wurde. Anschließend wurde die Installation der benötigten Systemen auf den lokalen Rechnern der Projektgruppenmitglieder durchgeführt, um eine flexible Bearbeitung der anstehenden Aufgaben garantieren zu können, sodass keine Abhängigkeit von Computerräumen in den Gebäuden der Universität bestand. Zu Beginn des Projekts wurden verschiedene Seminarthemen seitens des Lehrstuhls vorgeschlagen, die einen direkten Zusammenhang zu den bevorstehenden Aufgaben innerhalb der Projektdurchführung bilden. Diese wurden anschließend an die einzelnen Projektgruppenteilnehmer vergeben und von diesen bis zum 09.03.2015 bearbeitet. Hierbei wurden die einzelnen Themen in einer 30-minütigen Präsentation sowohl den Lehrstuhlbetreuern, als auch den anderen Projektmitgliedern vorgestellt. Im Folgenden werden die einzelnen Seminarthemen mit dem jeweiligen Bearbeiter genannt (siehe Anhang 11):

- Potentiale von Big Data für das Customer Relationship Management (Christian Janßen)
- Datenanreicherung/-ergänzung (Jannes Spekker)
- Agiles Projektmanagement (Kai Hänig)
- Data Mining Methoden und Werkzeuge (Kamiran Tizyani)
- Entwicklungsumgebung und Frameworks (Nils Worzyk)
- Mobilitätsrelevante Sonsorik im Verkehr (Olga Schwarz)
- Visual Analytics (Phillip Schumacher)

Die gewonnenen Erkenntnisse der ersten Phase wurden gebündelt und in die nächste Phase, die Entwurfsphase übertragen um das Projekt zu konkretisieren.

2.2.2 Entwurfsphase

Die Aufgabe der Entwurfsphase war es, gewonnene Erkenntnisse aus voran gegangenen Phasen zu übernehmen und diese zu konkretisieren, um einerseits das zu erreichende Ziel zu definieren und andererseits einen Rahmen abzustecken, anhand dessen die Zielerreichung ermöglicht wird. Während der Entwurfsphase wurden vier unterschiedliche Gruppen gebildet, die sich im Folgenden mit den vier Hauptthemen auseinander gesetzt haben. Diese vier Hauptthemen wurden vorab von der Projektgruppe im Kollektiv erörtert und als Themen mit sehr hoher Relevanz eingestuft und bildeten somit die Basis der Entwurfsphase, auf der anschließend die Entwicklungsphase aufgebaut wurde:

Wetterdaten

Die Wetterdaten dienen der Bereitstellung von Meta-Informationen, die es den Customern ermöglichen, neben den konkreten Verkehrsflussdaten ebenfalls zusätzliche Informationen im zu erstellenden Web-Portal herranzuziehen, um eine Analyse des betrachteten Zeitsraums durchzuführen und ebenfalls eine genauere Prognose zu ermöglichen. Die Wetterdaten für den Großraum Oldenburg wurden aus der nächst möglichen Wetterstation XX Kilometer außerhalb des Stadtzentrums herangezogen.

OpenStreetMap-Daten

Zur genauen örtlichen Definition durch Koordinaten, wurden die Daten der OpenStreetMaps-Datenbank (OSM) Foundation herangezogen, sodass sämtlichen Zählschleifen digital auf einer Karte dargestellt werden können. Die räumliche Lage der vereinzelten Zählschleifen wurden der Projektgruppe lediglich anhand von physischen Karten im PDF-Format zur Verfügung gestellt, sodass eine Koordinatenfestlegung sämtlicher Zählschleifen zu Beginn der Projektgruppe essentiell war. Die Projektgruppe hat sich dazu entschieden, OSM-Daten zu nutzen, da diese als Open-Source vorliegen und die Nutzung somit keiner Regulation unterliegt. Darüber hinaus, werden die OSM-Daten genutzt, um sämtliche Karten innerhalb des Web-Portals darzustellen.

Allgemeine Datenrecherche

Die Allgemeine Datenrecherche-Gruppe hatte die Aufgabe, neben den gegebenen Daten, wie beispielsweise den Wetterdaten oder den Verkehrsflussdaten zusätzliche Datenquellen zu eruieren, welche Informationen und somit einen konkreten Beitrag zur umfangreicheren Funktionalität des Web-Portals liefern konnten. Um dieses Ziel zu erreichen, wurden diverse verkehrstechnische Leit- und Statistikeinrichtungen von der Gruppe kontaktiert. Im weiteren Verlauf wurde eine potentielle Zusammenarbeit erörtert und über die Sinnhaftigkeit diskutiert.

Konzeptionierung

Die Konzeptions-Gruppe zielte auf die Erstellung eines oder mehrere konkreter Use-Cases ab, anhand welcher das finale Web-Portal für die späteren Customer programmiert werden sollte. Hierbei wurde festgelegt, welche Daten genutzt und welche Algorithmen programmiert werden sollten. Ebenfalls wurden die Interessen sämtlicher Stakeholder berücksichtigt.

Am Ende der Entwurfsphase stand somit ein Konzept, welches definierte, welche Funktionalitäten das fertige Web-Portal aufweisen musste und wie diese umzusetzen waren. Es wurde genau definiert, mit welchen Programmiermethoden bzw. welchen Programmiersprachen und welchen Systeme spezifische Probleme gelöst werden sollten.

2.2.3 Entwicklungsphase

In der Entwicklungsphase wurden erneut vier Gruppen gebildet, die sich mit der konkreten Umsetzung der beschriebenen Aufgabe in Eigenregie auseinander setzten. Hierbei wurden umfangreiche Aufgabenpakete definiert, die einen konkreten projektabhängigen Output liefern mussten. Die Hauptaufgabe bestand darin die auferlegten Anforderungen in einzelne Teilaufgaben zu zerlegen und diese mit den gegebenen Systemen und Parametern umzusetzen. Die genannten Gruppen werden im Folgenden kurz beschrieben:

ADAC-Daten

In dieser Gruppe wurden die ADAC-Daten, die von den Projektbetreuern zur Verfügung gestellt wurden, standardisiert, bereinigt und anschließend optimiert. Diese Daten beinhalten detaillierte Informationen zu bezüglich des CO2 Ausstoßes und der Feinstaubbelastung einzelner Fahrzeuggruppen. Auf der Grundlage der ADAC-Daten, wurde ein CO2-Durchschnittswert ermittelt, mit Hilfe dessen, der CO2-Ausstoß der erfassten Fahrzeuge bei einer Zählspule berechnet wurde. Hierdurch konnte eine Aussage über den lokalen CO2-Belastungsfaktor bei einzelnen Zählschleifen in Oldenburg getroffen werden.

Kurzfristige Prognose

Diese Gruppe beschäftigt sich mit der Prognose von kurzfristigen Verkehrsflussdaten. Als Definition für die Kurzfristigekeit, wurde ein Zeitintervall zwischen 0 und 60 Minuten gewählt. Ebenfalls wurde für die kurzfristige Prognose die Prämisse eines wissenschaftlich fundierten Algorithmus gesetzt, sodass die Vorhersage ein realistisches Ergebnis generieren konnte.

Langfristige Prognose

Die Gruppe der langfristigen Prognose hatte die Aufgabe, den Verkehrsfluss anhand eines eigens recherchierten Algorithmus darzustellen. Die Hauptprämisse der langfristigen Prognose wurde so definiert, dass der Zeitraum als Übergang zur kurzfristigen Prognose auf größer als 60 Minuten festgelegt wurde. Andere Parameter konnten frei gewählt wurden. Essentiell hingegen war jedoch ein wissenschaftlich fundiertes Gerüst, welches den Algorithmus als valide Funktion zur Bestimmung zukünftiger Verkehrsflüsse identifiziert.

Frontend

Diese Gruppe befasste sich mit der Enddarstellung der Projektergebnisse. Hierbei soll ein Portal entwickelt werden, welches den Nutzern eine detaillierte Darstellung der generierten Informationen ermöglicht. Darunter fallen sowohl die Darstellung der Fahrzeugbelastungen an den Zählschleifen, als auch die Darstellung diverser Metainformationen, die ebenfalls eine gewisse Relevanz von Verkehrsflüssen aufweisen.

2.2.4 Gantt-Diagramm

Das Folgende Kapitel gibt zunächst einen Überblick über die einzelnen Phasen des Projektes Rapid. Im Anschluss daran, wird das Gantt Diagramm vorgestellt, welches die

zeitliche Abfolge der angesprochenen Phasen mit seinen Aktivitäten graphisch darstellt. Das Projekt ist in insgesamt 4 Phasen eingeteilt und wird in der Abbildung 1 dargestellt. In der Vorbereitungs- und Seminarphase sind zunächst grundlegende organisatorische Rahmenbedingungen zu klären. Zusätzlich dazu, wird die Einrichtung der lokalen projektabhängigen Systemstruktur vorgenommen. Während der Seminarphase werden die jeweiligen vergebenen Seminarthemen individuell von den Projektgruppenteilnehmern bearbeitet. Die Entwurfsphase, Entwicklungsphase und Dokumentationsphase stehen im direkten Projektbezug. Zunächst wird ein konkretes Konzept während der Entwurfsphase erstellt. Der Übergang zur Entwicklungsphase ist fließend und beinhaltet die Umsetzung der einzelnen Konzeptbereiche. Die Dokumentation des Gesamtprojekts beginnt bereits mit der Entwurfsphase und vollendet das Projekt- Phasenmodell.

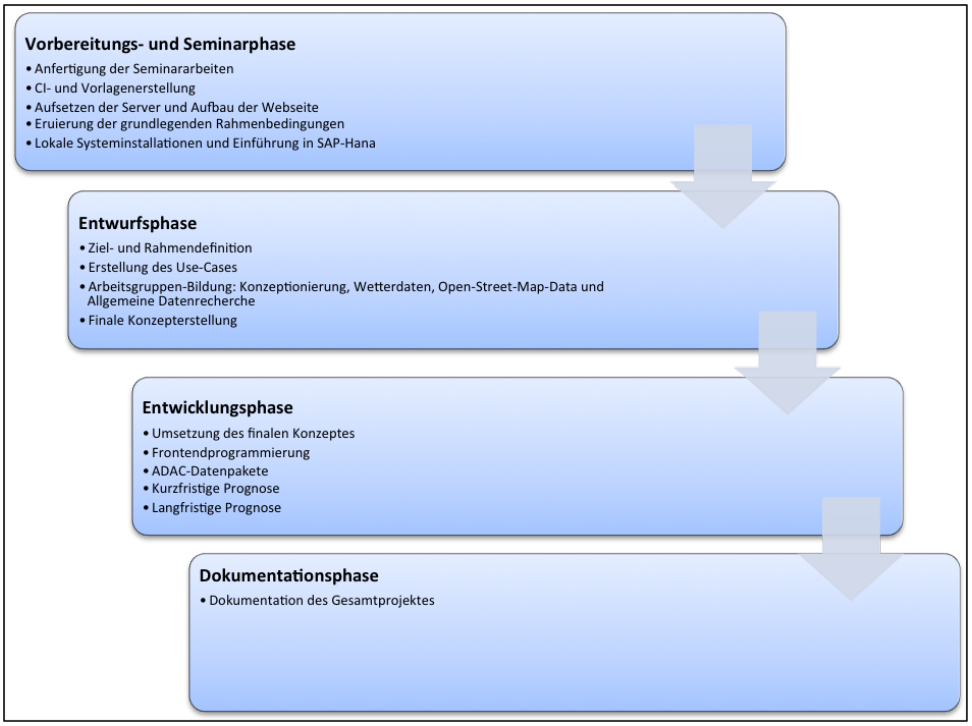


Abbildung 1: Projekt- Phasenmodell

Das zuständige Gantt Diagramm baut direkt auf dem Phasenmodell auf. Die Abbildung 2 zeigt eine tabellarische Zeitübersicht der einzelnen Phasen.

✓		Projektanfang	1 Tag	Mo 20.10.14	Mo 20.10.14
✓		**Projektverlauf**	**310 Tage**	**Mo 20.10.14**	**Do 15.10.15**
✓		**Vorbereitungs- und Seminarphase**	**121 Tage**	**Mo 20.10.14**	**Mo 09.03.15**
✓		**Entwurfsphase**	**71 Tage**	**Di 10.03.15**	**So 31.05.15**
✓		**Entwicklungsphase**	**109 Tage**	**Mo 01.06.15**	**Mo 05.10.15**
✓		Dokumentationsphase	118 Tage	Mo 01.06.15	Do 15.10.15
✓		Projektende	1 Tag	Do 15.10.15	Do 15.10.15

Abbildung 2: Tabellarische Zeitübersicht

Insgesamt wurde die Projektdauer auf 310 Tage beziffert wobei eine 6 Tage Arbeitswoche zugrunde gelegt wurde. Lediglich der Sonntag ist als arbeitsfreie Zeit deklariert. Die nachfolgende Abbildung 3 zeigt die Timeline der tabellarischen Zeitübersicht aus Abbildung 2.

Abbildung 3: Timeline Gesamtprojekt

Die Vorbereitungs- und Seminarphase umfasst 121 Tage. Die Erstellung der Seminararbeiten sowie die Eruierung grundlegender Rahmenbedingungen nehmen den Großteil der eingeplanten Zeit in Anspruch. Die Abbildung 4 zeigt einen kurzen Einblick des daraus resultierenden Gantt Diagramms.

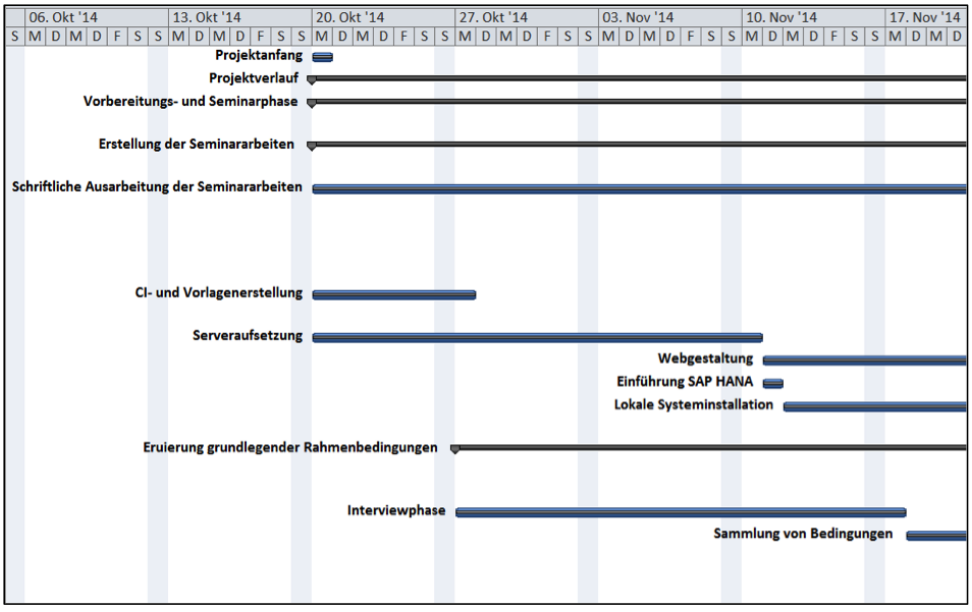


Abbildung 4: Gantt Diagramm Vorbereitungs- und Seminarphase

Für das Gantt Diagramm der Entwurfsphase, welche in Abbildung 5 dargestellt wird, wurden 71 Tage veranschlagt. Hierbei war vor allem die Einteilung der Arbeitsgruppen ein zeitintensiver Faktor der rund 2/5 der Zeit eingenommen hat. Die finale Konzeptgestaltung bildet den Übergang zur Entwicklungsphase und umfasste 23 Tage.

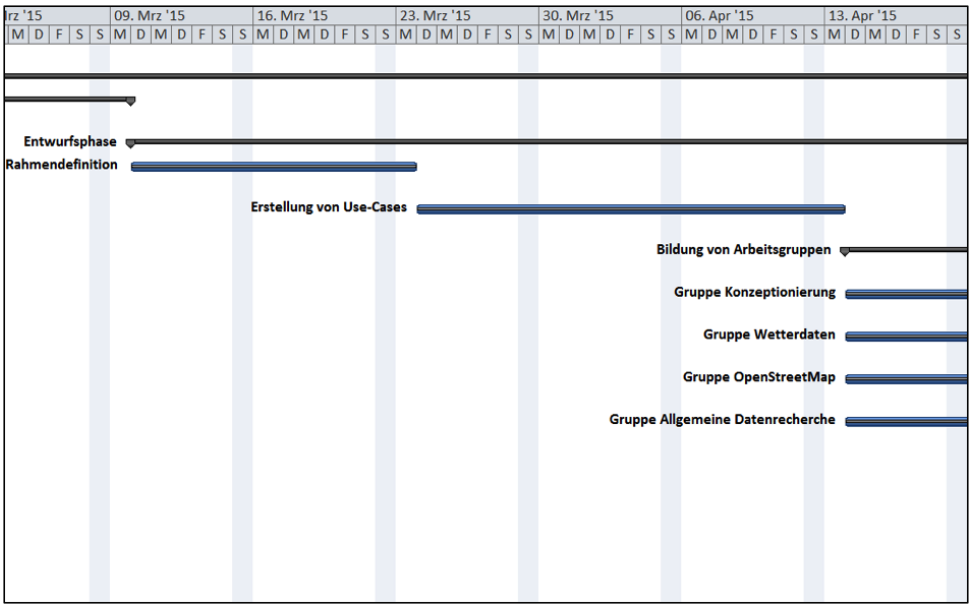


Abbildung 5: Gantt Diagramm Entwurfsphase

Aufbauend auf der Entwurfsphase wird die Entwicklungsphase betrieben. Dabei sind die Arbeitsgruppen abhängigen Aktivitäten kurzfristige Prognose, langfristige Prognose und ADAC-Datenpakete mit jeweils 97 Tagen veranschlagt worden. Die Programmierung des Portals nimmt den größten Bereich mit insgesamt 109 Tagen ein. Die nachfolgende Abbildung 6 gibt einen Überblick über das resultierende Gantt Diagramm der Entwicklungsphase.

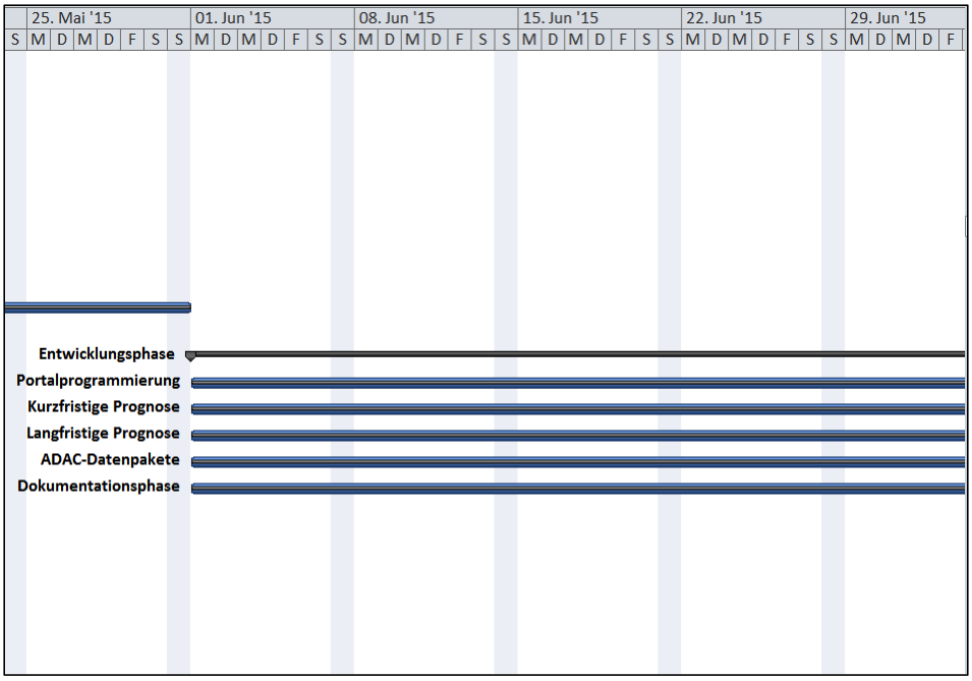


Abbildung 6: Gantt Diagramm Entwicklungsphase

Die Dokumentationsphase verläuft parallel zur Entwicklungsphase und ist mit 118 Tagen der Zweit längste Zeitfaktor des gesamten Projekts.

2.3 CRISP-Data Mining

Die Projektgruppe RAPID beschäftigt sich mit einer umfangreichen Data Mining Aufgabe und benötigt somit ein Projektmanagement Framework, welches eine bessere Strukturierung der Herangehensweise der eigentlichen Umsetzung des Gesamtprojektes ermöglicht. Um diese verbesserte Koordination bei der Umsetzung des Projektes nutzen zu können, wurde der Knowledge Discovery in Databases Prozesses (KDD) herrangezogen. Ein Prozess, der bei der eine effektive Datenanalyse und deren Verständnis angewendet wird. Eine der weit verbreitetsten Varianten des KDD ist der Cross Industry Standard Process for Data Mining (CRISP-DM) welcher an dieser Stelle genutzt werden soll. Dieser CRISP-DM besteht aus 6 Schritten und ist eine der weit verbreitesten KDD Prozesse. Die 6 Schritte, wie sie im Projekt RAPID genutzt werden sollen sind folgende:

1. Business Understanding
2. Data Understanding
3. Data Preparation
4. Modeling

5. Evaluation
6. Deployment

Neben der Reihenfolge ist ebenfalls die konkrete Anordnung der 6 Schritte essentiell für den Projekterfolg. Diese ist im folgenden Schaubild dargestellt.

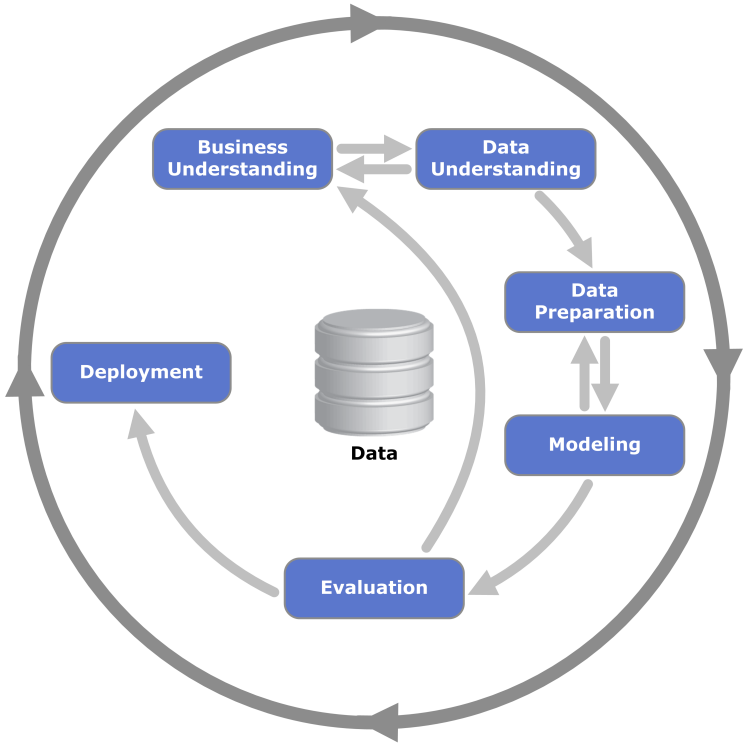


Abbildung 7: Prozessdiagramm für die Beziehungen der einzelnen Phasen des CRISP-DM

Source: „CRISP-DM Process Diagram“ by Kenneth Jensen - Own work. Licensed under CC BY-SA 3.0 via Wikimedia Commons - `https://commons.wikimedia.org/wiki/File:CRISP-DM_Process_Diagram.png#/media/File:CRISP-DM_Process_Diagram.png`

Die ersten Phase, Business Understanding, beschäftigt sich zunächst damit, die Ziele und Voraussetzungen für das Projekt aus unternehmerischer Sicht festzulegen. Diese unternehmerischen Ziele sollen dann weiterführend in konkrete Ziele überführt werden, die mit Hilfe von Data-Mining bearbeitet werden können. Die zweite Phase, Data Understanding, beschäftigt sich im Allgemeinen damit, zunächst benötigte Daten zu beschaffen und dann mit diesen Daten vertraut zu machen. Dabei sollen zum einen mögliche Probleme in der Datenqualität identifiziert und zum anderen die Daten hinsichtlich möglicher Zusammenhänge untersucht werden. Die dritte Phase, die Data Preparation, beschäftigt sich damit, die in der zweiten Phase beschafften Daten für die späteren Analysen vorzubereiten. Darunter fällt, besondere Werte wie Ausreißer oder NULL-Werte gesondert zu behandeln oder die Daten in das benötigte Format zu bringen. Es herrscht allgemeiner

Konsens darüber, dass dieser Prozessschritt die meiste Zeit [WFH11] beansprucht. Dabei gilt es einerseits aus den Gesamtdaten die relevanten Variablen herauszufiltern und bereit zu stellen und andererseits, Gegebene Daten zu bereinigen und lediglich die relevanten Daten zur Weiterverabeitung in Informationen freizugeben. Die vierte Phase, das Modeling, beschäftigt sich damit verschiedene Modelle zu definieren, mit Hilfe welcher die vorbereiteten Daten ausgewertet werden können. Beispiele hierfür sind Clustering-Verfahren, Regressionsanalysen oder künstliche neuronale Netze. Im Rahmen dieses Projektes werden insbesondere Regressionsanalysen zur modellhaften Darstellung relevanter Fakten herangezogen Die fünfte Phase, die Evaluation, beschäftigt sich mit der konkreten Ergebnisgenerierung des Projektes. Es werden sämtliche relevante Daten einbezogen und analysiert, sodass ein konkretes Fazit gezogen werden kann. Hierbei ist es essentiell, dass der eigentliche Output mit den gesteckten Zielen und Erwartungen, die in der Entwurfsphase des Projektes festgelegt wurden, abgeglichen wird. Darüber hinaus sollte eine Reflektion des Projektablaufs und die Planung der nächsten Schritte Teil der Evalutaion-Phase sein. Die sechste und letzte Phase, das Deployment, beschreibt den Entwurf und die Umsetzung einer Strategie zur Implementierung des generierten Modells und des beinhaltenden Programms. Darüber hinaus müssen Pläne zur Instandhaltung und Wartung des fertigen Systems erstellt und eingeführt werden. Der finale Projektreport wird durch die Projektteilnehmer geschrieben und eine finale Reflektion über den Gesamtprojektverlauf sollte durchgeführt werden. Wie genau die beschriebenen Schritte im Projekt RAPID eingearbeitet und umgesetzt werden, wird in den folgenden Kapiteln näher erörtert werden. Das CRISP-DM soll stets als Referenz dienen und wird kontinuierlich Teil der folgenden Dokumentation sein. In diesem Zusammenhang werden im Projekt RAPID lediglich die Schritte 2 bis einschließlich 5 berücksichtigt, da die erste Phase im Rahmen einer Seminararbeitsperiode erarbeitet wurde, sodass ein fundiertes Wissen über sämtliche Systeme und Vorraussetzungen vorhanden ist. Die sechste Phase wird ebenfalls weggelassen, da es sich um ein universitäres Projekt handelt, sodass eine konkrete Implementierungsstrategie des Systems nicht erarbeitet werden muss.

2.4 SCRUM

Neben CRISP-DM wird SCRUM als konkretes Umsetzungframework der einzelnen Schritte herangezogen und dient der wöchentlichen bzw. monatlichen Planung der einzelnen zu bewältigenden Aufgaben und der entsprechenden Projektteilnehmer. Bei SCRUM handelt es sich um ein agiles Projektmanagement-Framework, welches stets in der Softwareentwicklung eingesetzt wird (vgl. [DKS01] S. 13). Im Zuge der Projektgruppe RAPID wird der Inbegriff des SCRUM abgewandelt und auf die individuellen Bedürfnisse der Projektgruppe angepasst. So wird das Projektmanagement-Framework nicht lediglich zur reinen wöchentlichen Planung genutzt, den sogenannten Sprints, sondern unterstützt das Team bei einer allgemeinen Aufteilung der SCRUM-Teams mit umfangreichen Aufgaben. Hierbei wurden, wie bereits in Kapitel 2.2.3, nach umfangreicher Planungen und Analysen der Projektaufgabe, vier Gruppen gebildet, die über die gesamte Entwicklungszeit beibehalten werden sollen.

1. Gruppe: Frontenddarstellung der Ergebnisse und deren Visualisierung
2. Gruppe: ADAC
3. Gruppe: kurzfristiger Prognosealgorithmus der Zählschleifen
4. Gruppe: langfristiger Prognosealgorithmus der Zählschleifen

Jede Gruppe beinhaltet ein Entwicklungsteam von 2-4 Entwicklern und einen SCRUM-Master, der ebenfalls Teil des Entwicklungsteams ist, jedoch ebenfalls administrative Aufgaben übernimmt und mit dem Product Owner kommuniziert. Der Product Owner ist somit der Projektleiter, dem die Komplettkoordination der Projektumsetzung obliegt und der stets in Kontakt mit den Scrum-Mastern steht, um den konkreten Fortschritt zu überprüfen. Ebenfalls ist der Product-Owner für die Kommunikation mit den Stakeholdern verantwortlich. Die obigen vier Gruppen arbeiten autark voneinander und werden lediglich über den Product Owner gesteuert, der den kontinuierlichen Progress der SCRUM-Teams stets überwacht und neue Aufgaben innerhalb der Gruppen mit dem SCRUM Master koordiniert. Da die Aufgaben, die die einzelnen SCRUM-Teams zu bewältigen hatten, jedoch äußerst umfangreich waren und stets unterschiedliche Entwicklungszeiträume in Anspruch genommen haben, wurde diese Abwandelung der individuellen Koordination innerhalb der Gruppen gewählt und lediglich 2 wöchentliche Meetings vereinbart, bei denen der aktuelle Stand der Gruppen vorgetragen und eine umfangreiche Abstimmung, falls nötig, durchgeführt werden konnte. Man kann somit festhalten, dass das sehr agile Projektmanagement Framework SCRUM innerhalb der Projektgruppe RAPID nochmals agiler gestaltet wurde, um den entsprechenden Gegebenheiten und Problemstellungen gerecht zu werden. Diese wurden aufgrund von unerwartet auftretenden Sachverhalten nötig, die in der Projektplanung, welche im Vorhinein durchgeführt wurden, nicht abzusehen waren.

3 Rahmenbedingungen

In diesem Kapitel werden zunächst die organisatorischen- und anschließend die technischen Rahmenbedingungen beschrieben.

3.1 Organisatorische Rahmenbedingungen

Die organisatorischen Rahmenbedingungen für die Durchführung des Projekts wurden bei dem ersten Treffen am 20.10.2014 festgelegt. Es wurde festgelegt, dass die Moderation der Sitzung und das zu erstellende Protokoll durch die Projektmitglieder in einer wöchentlichen Abwechslung durchgeführt und geschrieben werden. Der Projektstart begann am 20.10.2014 und endete am 15.10.2015. Das Projekt wurde auch in den Semesterferien weitergeführt. Die Projektmitglieder hatten einen Anspruch auf drei Urlaubswochen.

3.1.1 Projektgruppenmitglieder und ihre Rollen

Die Projektgruppe RAPID besteht aus sieben Masterstudenten der Wirtschaftsinformatik der Universität Oldenburg. Im Folgenden werden diese kurz mit Ihren Aufgaben innerhalb der Projektgruppe vorgestellt.

Abbildung 8: Gruppenfoto der Projektgruppe RAPID: sieben Gruppenmitglieder und drei Betreuer

- **Kai Hänig**
  Kai Haenig ist Projektgruppenleiter von RAPID. Er befasst sich hauptsächlich mit organisatorischen Aufgaben wie der Pflege des verwendeten Ticketsystems Redmine. Darüber hinaus kommuniziert er mit den verschiedenen externen Akteuren, die im Zuge des Projektes eine Rolle spielen (Verkehrsleitzentrale Oldenburg, DLR Braunschweig, etc). Mit Nils Worzyk ist er für die kurzfristige Prognose der Verkehrsdaten

zuständig. Passend zu seiner Tätigkeit befasst er sich in seiner Seminararbeit mit Konzepten, Werkzeugen und Methoden des agilen Projektmanagements.

- **Kamiran Tizyani**
  Kamiran ist stellvertretender Projektleiter der Gruppe RAPID. Er unterstützt Kai bei seiner Tätigkeit. Zudem ist er für die Berechnung des Schadstoffverbrauches zuständig. In seiner Seminararbeit befasst er sich mit den Methoden und Werkzeugen des Data Mining. Des Weiteren nahm und kümmerte er sich um die Kommunikation mit diversen Unternehmen deutschlandweit auf, um Informationen über Datenstrukturen und weitere Beschaffenheit und Möglichkeiten zu erhalten.

- **Philipp Schumacher**
  Als Finanzbeauftragter der Projektgruppe hatte Philipp die Aufgabe die Kasse zu führen und somit das Projektbudget zu verwalten. Darüber hinaus beschäftigt er sich mit statistischen Verfahren und ist im Zuge der langfristigen Prognose für die Implementierung von Algorithmen in der Sprache R Language verantwortlich. In seiner Arbeit setzte sich Philipp mit Visual Analytics auseinander.

- **Olga Schwarz**
  Olga ist hauptsächlich für die Pflege und Aufbereitung der Dokumentationsinhalte zuständig. Des Weiteren kümmerte sie sich um die Beschaffung und den Import der Eventdaten in SAP Hana. Mit Kamiran zusammen befasst sie sich ebenfalls mit der Berechnung des Schadstoffverbrauches. In ihrer Arbeit beschäftigte sie sich mit Sensorik im Verkehrswesen.

- **Jannes Spekker**
  Jannes ist Webseiten-Beauftragter der Gruppe RAPID. Im Zuge der Implementierung ist er verantwortlich für die Darstellung im Frontend. Dazu schreibt er PHP-Skripte und bindet entsprechende Frameworks wie D3.js ein. Seine Seminararbeit lautet „Datenanreicherung/-ergänzung durch Open Data Sources“.

- **Christian Janßen**
  Christian ist neben Nils Worzyk für die Serveradministration verantwortlich. Im Zuge der langfristigen Analyse arbeitet er mit Philipp zusammen. Darüber hinaus ist er mit Jannes am Frontend und insbesondere mit der Administration dort und sicherheitsrelevanten Aspekten beschäftigt. In seiner Arbeit behandelt er CRM und Big Data.

- **Nils Worzyk**
  Nils ist der zweite Zuständige für die Administration der Server. Zudem ist er mit Kai an der kurzfristigen Prognose beteiligt. Er setzt sich darüber hinaus mit der Umsetzung von Ablaufmodellen aus dem Data Mining auseinander und erforscht inwieweit sich diese im Zuge der Projektgruppe anwenden lassen. In seiner Arbeit setzt er sich mit Entwicklungsumgebungen und Frameworks um SAP HANA auseinaander.

- **Galina Janusauskiene**
  Am 05.01.2015 aus der PG ausgetreten.

- **Milan Tomovic**
  Am 20.02.2015 aus der PG ausgetreten.
- **Janine Haase**
  Am 30.04.2015 aus der PG ausgetreten.

3.1.2 Projektgruppenbetreuer

Die Projektgruppe RAPID besteht neben den einzelnen studentischen Mitgliedern auch aus den Betreuern, die das Team insbesondere bei der Kommunikation mit den Partnerunternehmen, bei dem Betrieb von SAP HANA und bei der allgemeinen Organisation unterstützen:

- M.Sc. Alexander Sandau
- Dr.-Ing Benjamin Wagner vom Berg
- M.Sc. Daniel Stamer
- Dipl.-Inf. Manuel Osmers (Bis einschließlich 31.08.2015)

3.1.3 Wöchentliche Ablaufplanung

Das offizielle Projektmeeting wurde jeden Montag, 12:15 Uhr am betreuenden Lehrstuhl abgehalten. Hierbei waren alle Betreuer der Projektgruppe anwesend, um den aktuellen Status der Projektgruppe zu erfahren und bei der Klärung möglicher Fragen oder Anliegen zur Verfügung zu stehen. Dieses Meeting wurde stets mit einer Tagesordnung vorab geplant und mithilfe eines Protokolls niedergeschrieben, welches alle Beschlüsse und Informationen enthielt. Die Projektgruppe mit ihren sieben Mitgliedern hat stets jeden Montag, Dienstag und Mittwoch den Projektgruppenraum zur Verfügung gestellt bekommen und konnte so im Team an der Aufgabenstellung weiter arbeiten. Hierbei wurden Kernarbeitszeiten von 11 Uhr – 16 Uhr jeden Montag und Mittwoch festgelegt und die Erfüllung des minimalen wöchentlichen Stundenziels von 14 Stunden in Gleitzeit den Teilnehmern selbst überlassen.

3.1.4 Gruppenkommunikation

Die Kommunikation der Gruppe wurde überwiegend durch direkte Kommunikation vor Ort durchgeführt, sodass die einzelnen SCRUM-Teams sich direkt miteinander abstimmen konnten. Darüber hinaus, wurde neben einem E-Mail Verteiler ebenfalls Mobiltelefonnummern ausgetauscht um einerseits stets erreichbar zu sein und sich auch an Tagen, an denen Home-office betrieben wurde mit den anderen Projektteilnehmern austauschen zu können. Des Weiteren wurde eine Whats-App Gruppe eingerichtet, die sowohl der kurzfristigen Kommunikation von wichtigen Terminen oder Anliegen als auch zur Abstimmung inhaltlicher Fragen diente, die alle Projektteilnehmer betreffen.

3.1.5 Strategische Partner

Der Strategische Partner der Projektgruppe RAPID und gleichzeitig wichtigster Datenlieferrant ist die Stadt Oldenburg, bzw. die Verkehrsleitzentrale Oldenburg. Der Projektgruppe wurde von dem Partner einerseits eine detaillierte Einführung über die Funktionsweise der Verkehrsflusszählung im innerstädtischen Bereich Oldenburgs gegeben und andererseits stets auf sämtliche auftretenden Fragen geantwortet.

3.2 Technische Rahmenbedingungen

In den ersten Wochen der Projektgruppe wurden zahlreich technische Rahmenbedingungen aufgestellt. Dazu zählte die Einrichtung der SAP HANA an den privaten PC's der Projektgruppenmitglieder sowie die Einrichtung eines Servers und die Kommunikation zwischen den Mitgliedern und der Betreuer. Diese werden im folgenden Kapitel genauer beschrieben.

3.2.1 Nutzung von Redmine für die Projektgruppenorganisation

Redmine ist ein web-basiertes Projektmanagement Tool, welches viele verschiedene Aufgaben mehrere User innerhalb eines oder mehrerer Projekte verwaltet. Im Falle des Projektes RAPID wurde lediglich ein Projekt verwaltet, welches jedoch in mehrere Kategorien unterteilt wurde. Um den stetigen Fortschritt der Projektgruppe zu dokumentieren und sämtliche Aufgaben web-basiert zu speichern und allen Projektmitgliedern zugänglich zu machen, wurde das Projektmanagement-Tool Redmine eingeführt. Dieses erlaubt es sämtlichen Usern, Tickets bzw. Aufgaben zu erstellen und diese sich selbst oder anderen Teilnehmern zuzuweisen. Das System ermöglicht es umfangreiche Rollenkonzepte zu verteilen, sodass lediglich Rechte vergeben werden können, die von den einzelnen Nutzern auch verwendet werden sollen. Durch Redmine ist der Projektleiter in der Lage den stetigen Fortschritt der einzelnen Projektteilnehmer einzusehen und diesen mit den jeweiligen Personen direkt zu besprechen. Hierbei kann neben dem konkreten inhaltlichen Fortschritt ebenfalls der benötigte Zeitaufwand gebucht werden, sodass der Projektleiter Aufgabenumfänge basierend auf der Geschwindigkeit des Programmierers vergeben kann. Darüber hinaus kann gewährleistet werden, dass kein User keine konkrete Aufgabe hat und somit Leerlauf von vorne herein vermieden wird. Um den Gesamtprojektverlauf stets beobachten zu können, werden sogenannte due-dates vergeben. Diese beschreiben das geplante Abgabedatum der übertragenen Aufgabe. Diese werden durh das System automatisiert in ein Ganntt-Chart überführt sodass der stetige Projektverlauf dargestellt werden kann und eventuelle Verzögerungen direkt auffallen. Das Projekt RAPID wurde, wie bereits beschrieben, in mehrere Kategorien unterteilt, dies sich wie folgt zusammensetzen:

- Research
- Projektmanagement
- Daten

  - Datenerhebung
  - Datenanalyse
  - Datenprognose
- Server
- Front-End
- Dokumentation

Hierbei wurden sämtliche Aufgaben stets einer Kategorie zugewiesen. Research beinhaltet sämtliche Aufgaben die eine umfangreichere Recherche in Vorbereitung auf andere Aufgaben beinhalten. In dieser Kategorien sind ebenfalls sämtliche Aufgaben zur Einführung in die einzelnen Systeme beinhaltet. Im Projektmanagement wurden alle Aufgaben des Projektleiters zur Planung und Organisation des Projektes verbucht. Ebenfalls sind sämtliche administrativen Aufgaben des Redmine-Systems als Projektmanagement-Tool Teil dieser Kategorie. Die Daten-Kategorie unterteilt sich in 3 Unterkategorien. Diese sind die Datenerhebung, die Datenanalyse und die Datenprognose. Diese Schritte wurden sukzessiv durchlaufen. Hierbei wurden die erhaltenen externen Daten zuerst erhoben bzw. die physische Präsens der Daten sichergestellt, anschließend wurde die Datenanalyse durchgeführt, sodass die Daten innerhalb der SAP HANA- Anwendung finden konnten. Zuletzt wurden basierend auf der Datenanalyse Prognosealgorithmen erstellt, die möglichst nah an den tatsächlich auftretenden Werten liegen sollte. Die Kategorie Server, beinhaltet alle Aufgaben, die bezüglich der Serverorganisation und deren Verwaltung anfielen. Hierbei sind sowohl Cloud-Systeme zur Dokumentenverwaltung als auch die Web-Server zur konkreten Darstellung des Web-Portals inkludiert. Das Frontend beinhaltet alle Aufgaben, die sich mit der direkten Visualisierung der Daten bzw. der umgebenden Verwaltung beschäftigen. Neben der Front-End Darstellung für den Kunden ist in dieser Kategorie ebenfalls die Programmierung des Backends und der Nutzerverwaltung des Web-Portals enthalten. Die finale Kategorie, die Dokumentation, beinhaltet sämtliche dokumentationsbzogene Aufgaben. Texte und Diagramme, die sich in der finalen Dokumentation wiederfinden können in dieser Kategorie anhand eines Tickets erstellt und beschrieben werden. Redmine ist somit ein essentielles Tool für die Projektgruppe um ein präzises und vollständiges Projektmanagement durchführen zu können.

3.2.2 Dokumentation mittels LaTeX

Für die Anfertigung der Seminararbeiten, Protokolle und die finale Dokumentation wird das Textverarbeitungsprogramm Latex verwendet. Dieses Tool wird insbesondere im naturwissenschaftlichen Bereich benutzt. Es handelt sich hierbei um ein Commpellierprogramm, welches mittels verschiedener Befehle die Formatierung des Dokumentes vordefiniert. Im Vergleich zu einem "What you see is what you get" - Textverarbeitungsprogramm wie Microsoft Office ist Latex geeigneter für Anfertigungen von großen Textdokumenten. Der Entschluss Latex für die Anfertigung der Dokumente während der Projektgruppe zu

benutzen, wurde in den ersten Wochen von den Projektmitgliedern gemeinsam entschieden.

3.2.3 Website

Die Projektgruppe erstellte eine Webseite, auf der sämtliche Informationen über das Projekt für außenstehende zur Verfügung gestellt werden. Die Webseite ist unter der Adresse http://www.rapid-ol.de/ aufrufbar. Auf der Webseite können sich außenstehende u.a. Informationen über die Ziele, das Thema, die einzelnen Teammitglieder, die Partner sowie den Auftraggeber verschaffen.

3.2.4 Server

Als Server Umgebung wurde eine Linux basierte Variante gewählt. Mit dem Ubuntu Server steht eine leistungsstarke Plattform mit einer hohen Programmvielfalt sowie aktueller Software zur Verfügung. Dadurch kann ein stabiler, sicherer und einfach zu bedienender Ablauf gewährleistet werden. Im Vergleich zu anderen Betriebssystemen bietet die Linux Distribution im Server- Einsatz erheblich Vorteile:

- Ubuntu ist für Server-Anwendungen kostenlos und frei verfügbar
- Kurze und klare Release Zyklen, im Schnitt alle Zwei Jahre, erleichtern die Planbarkeit und Transparenz der Server-Installation
- In der LTS Version wird ein fünfjähriger Sicherheitssupport durch Sicherheits-Updates für Paket-Installationen übernommen [Kof11]

Für die Projektgruppe steht die aktuelle Ubuntu Version 14.04.2 LTS trusty in der 64 Bit Version zur Verfügung. Mit einer Gesamtspeicherkapazität von 30 GB und einem standardmäßigen Ram-Speicher von 2 GB können die anfallenden Aufgaben bewältigt werden. Im Folgenden wird ein Überblick über eingerichtete, nutzerbasierte Zugriffe und Rollen auf dem Server gegeben:

- Administration Projektgruppe
  - Christian (Rolle Admin)
  - Nils (Rolle Co-Admin)
- Mitglieder Projektgruppe
  - Olga (Rolle Mitglied)
  - Kamiran (Rolle Mitglied)
  - Kai (Rolle Mitglied)
  - Philipp (Rolle Mitglied)
  - Jannes (Rolle Mitglied)
  - Janine (Rolle Ex-Mitglied)

  - Galina (Rolle Ex-Mitglied)
  - Milan (Rolle Ex-Mitglied)
- Externe Zugriffe
  - Betreuer (Rolle Extern)
  - R User (Rolle Extern RLang-Zugriff)

Neben der standardmäßigen Einrichtung sind folgende Features installiert worden:

- *PHP5:* Skriptsprache zur Erstellung von dynamischen Webseiten
- *phpMyAdmin:* Anwendung zur Administration von MySQL-Datenbanken
- *Redmine:* Unabhängige, frei verfügbare, webbasierte Projektmanagementsoftware mit integrierter Ticketverwaltung
- *Wordpress:* Web-Software zur Umsetzung von modernen Internetpräsenzen
- *SAP Hana Client:* Zugriffspunkt auf die SAP HANA In-memory Plattform
- *R und Rserve:* R als Open-Source-Sprache zur statischen Berechnung von Grafiken sowie Rserve als Server für den Zugriffspunkt auf R-Funktionalitäten Um den Datenaustausch per SFTP vorzunehmen, wurde ein gruppeninterner Ordner erstellt, auf dem alle Projektgruppenmitglieder Zugriff haben.

PuTTY als Administrationsverwaltung

PuTTY ist ein SSH fähiger Terminal-Client der im Rahmen der Projektgruppe vor allem als Administrationsunterstützung für den eingerichteten Server dient. Dabei muss zunächst eine neue Verbindung eingerichtet werden.

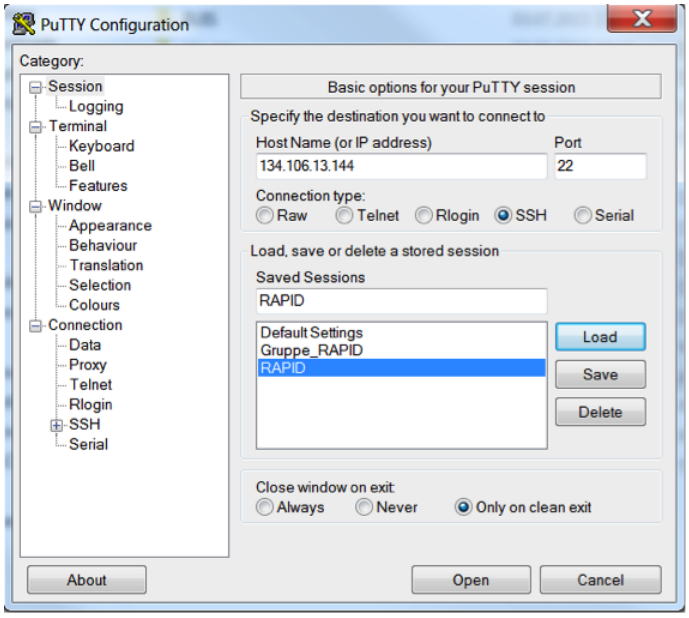


Abbildung 9: Kofigurationsdarstellung von PuTTY zum besseren Verständnis für die Einrichtung von PuTTY

Durch die Eingabe der IP Adresse, des dazugehörigen Ports sowie der Auswahl des Connection Types, im Fall der Projektgruppe RAPID SSH, kann eine PuTTY Session auf dem angegebenen Server gestartet werden.

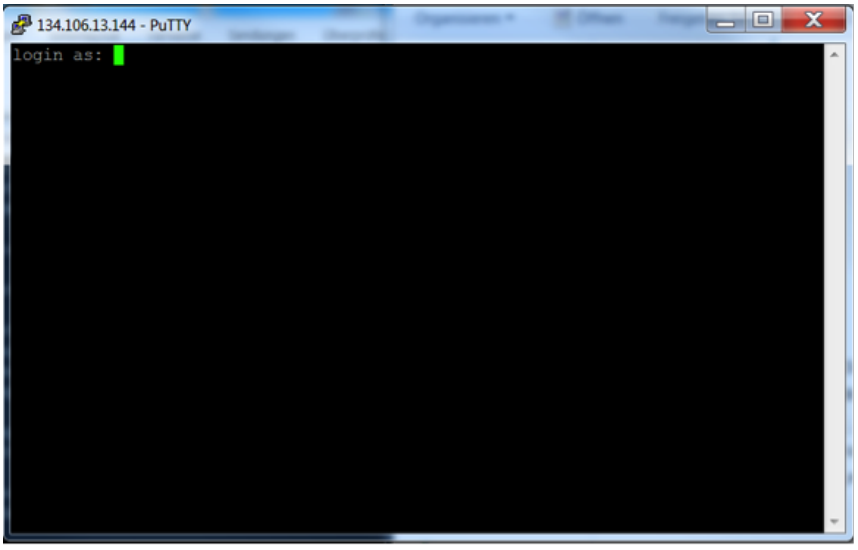


Abbildung 10: Terminal für PuTTY

In der gestarteten Terminalsitzung (Abbildung10), wird nach Eingabe des User-Namens sowie das korrespondierende Passworts die Administration des im vorherigen Kapitel vorgestellten Servers durchgeführt.

FileZilla

Mit der frei-verfügbaren Client- Server Software Filezilla wird der SFTP Zugriff auf den Ubuntu Server sichergestellt. Die Projektgruppenmitglieder können nach Einrichtung Dateien einsehen sowie hoch- bzw. herunterladen. Die Vorteile von Filezilla liegen vor allem in seiner Multiplattform Verfügbarkeit und in seiner übersichtlichen grafischen Oberfläche die eine komplizierte Einarbeitung vermeidet. Weitere Vorteile sind unter anderem:

- IPv6 Unterstützung
- Drag und Drop Unterstützung
- Ordner Synchronisierung
- Variable Übertragungsgeschwindigkeit sowie Unterbrechungen
- Protokollierung von Arbeitsaktivitäten

Um lange Suchzyklen zu vermeiden, wurde eine vorgegebene Ordnerstruktur (Abbildung 11) erstellt, in der die Projektgruppenmitglieder Daten einsehen und hochladen können.

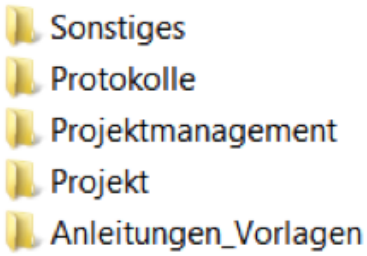


Abbildung 11: Ordnerstruktur von FileZilla

Die Ordner ”Sonstiges” und ”Anleitung_Vorlagen” beherbergen vor allem Projektgruppen interne Dokumente die eine Organisation von Prozessen vereinfachen sowie Recherchemöglichkeiten verschiedenster Projektbereiche ermöglichen. Im Ordner ”Protokolle” werden Sitzungsprotokolle der wöchentlich angelegten Meetings abgelegt und die Ordner ”Projekt”, Projektmanagement beinhalten Projektbezogene Daten sowie die Dokumentation des gesamten Projektes.

3.2.5 ODBC

Wie bereits in Kapitel 3.2.4 erwähnt, wurde der SAP HANA Client auf dem Ubuntu Server installiert. Um eine geeignete Steuerung zwischen der SAP HANA Datenbank und dem Ubuntu Server zu ermöglichen, wird eine unabhängige Datenbankschnittstelle benötigt. Zu diesem Zweck existieren verschiedenste Schnittstellen, wobei die Kommunikation mit dem Datenbank-Management-System (DBMS) hauptsächlich durch ODBC (Open Database Connectivity) und JDBC (Java Databse Connectivity) Schnittstellen ermöglicht wird [SAP14]. ODBC und JDBC unterstützen in ihrer Schnittstellenweisung dynamisches SQL und somit die Möglichkeit, SQL-Anweisungen während der Laufzeit zusammenzustellen. Zudem ist die Plattformunabhängigkeit und Flexibilität ein Charakterisierungsmerkmal, welches einen vielseitigen Einsatz vor allem bei ODBC ermöglicht [Cla15]. Aus Kompatibilitätsgründen wird daher die ODBC Schnittstelle im Rahmen der Projektgruppe für die Kommunikation mit der SAP HANA Datenbank verwendet. ODBC stellt defacto einen Standard dar, der durch die enge Zusammenarbeit der Herausgeber mit verschiedensten Datenbankherstellern schnell und unkompliziert neue Treiber einfließen lassen kann [Gil15]. Zudem stellt die Ubuntu Server Umgebung ein breites Repertoire an ODBC kompatiblen Treibern und Dokumentationen für die weitere Verwendung zur Verfügung. Des Weiteren fungiert das folgende Schema als Zugriffsreferenz und stellt einen beispielhaften Zugriff auf eine Datenbank mittels ODBC und PHP dar, wie er durch die Projektgruppe implementiert wurde:

1. Verbindungsaufbau zu einer beliebigen Datenbank

```
$conn = odbc_connect('hana', 'Verbindungsdaten der Datenbank',
   SQL_CUR_USE_ODBC);
```

2. Senden eines SQL- Statements

```
$queryExec = odbc_exec($conn, $queryString);
```

3. Ergebnisauswertung

```
return $queryExec;
```

4. Beenden des gesendeten SQL-Statements

```
\$close_handling;
```

5. Verbindungsabbau zur Datenbank

```
odbc_close($conn);
```

3.2.6 R-Language und R-Integration

R ist sowohl eine freie Programmiersprache für statistisches Rechnungen und Grafiken als auch eine Umgebung. Sie wurde von John Chambers und einigen seiner Kollegen in Bell Laboratories im Zuge eines GNU-Projektes entwickelt. R ist Teil dieses Projekts und auf vielen Plattformen verfügbar. R kann durch eigene Funktionen und einer großen Anzahl online abrufbarer Pakete erweitert werden. Die Sprache bietet Schnittstellen zu anderen Programmiersprachen und Software und lässt sich in zahlreiche Anwendungen, wie auch SAP HANA, integrieren [Fou15]. Da R ebenfalls Pakete für das Arbeiten hinsichtlich von Data Mining Aufgaben bereitstellt und sich darüber hinaus auch in den SAP-Script-Code einbetten lässt (R-Integration), hat sich die Projektgruppe für dessen Verwendung entschieden.
Da R ebenfalls Pakete für das Arbeiten hinsichtlich von Data Mining Aufgaben bereitstellt und sich darüber hinaus auch in den SAP-Script-Code einbetten lässt (R-Integration), hat sich die Projektgruppe für dessen Verwendung entschieden. Um die Eigenschaften

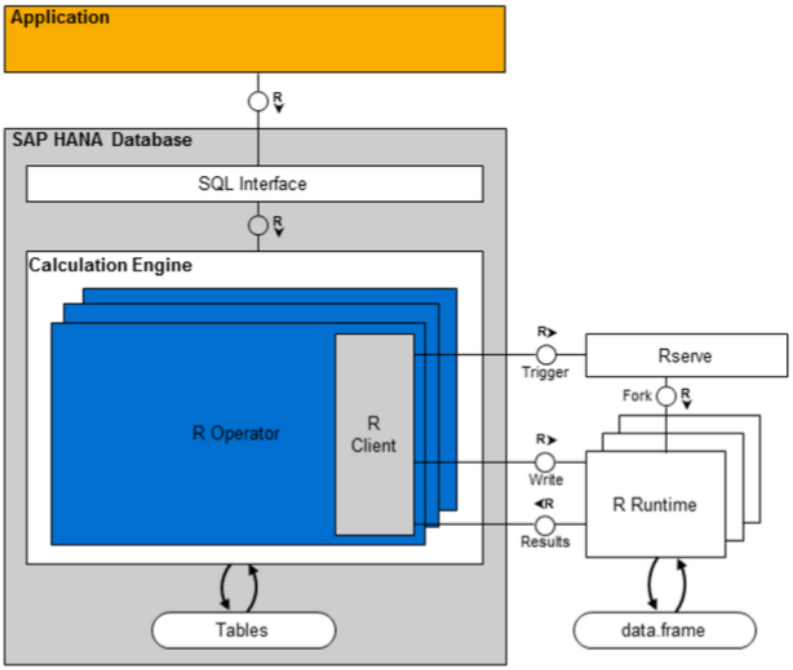


Abbildung 12: R Integration Architecture

von R-Language hinsichtlich der Daten aus den Tabellen in SAP HANA zu nutzen, ist die Einbindung von R-Language in die SAP HANA-Umgebung erforderlich. Dazu muss eine Verbindung über einen Putty-Server hergestellt werden. Auf diese Weise lässt sich R-Integration nutzen. Durch diese Integration ist es möglich in den SQL-Scripten der HANA-Datenbank R-Code in der Sprache RLANG einzubinden. Die benötigten Tabellen, die als Parameter innerhalb der R- Methoden verwendet werden, werden mit dem RLANG-Code, per Request an den R-Host (RSERV) weitergereicht. Dort werden die Tabellen in

einen „dataframe“ umgewandelt. Auf diese Weise kann der Inhalt der Tabelle in eine R-Variable abgebildet werden, die wiederum als Input-Parameter an die zuständige R-Funktion weitergereicht werden kann. Anschließend wird die Prozedur abgewickelt. Der oder die zurückgegebenen „dataframes“ werden wieder in Tabellen umgewandelt und an die HANA- Datenbank weitergereicht [SAP15] S. 3f.

4 SAP HANA

Eine der Rahmenbedingungen für die Projektgruppe war die Verwendung eines SAP HANA Systems. Das System wurde durch die Abteilung VLBA der Carl-von-Ossietzky Universität Oldenburg zur Verfügung gestellt, die sich wiederum SAP HANA Serverkapazitäten bei der Otto-von-Guericke Universität Magdeburg angemietet haben. Um die Funktionsweise und Vorteile des SAP HANA Systems besser verstehen zu können, wird zunächst in Abschnitt 4.1 ein allgemeiner Überblick über das SAP HANA System gegeben. In Abschnitt 4.2 wird dann näher auf die Architektur der SAP HANA eingegangen und welche Schnittstellen es innerhalb und außerhalb des Systems gibt.

4.1 Einleitung in SAP HANA

Die SAP HANA-Technologie ist eine von SAP entwickelte und vertriebene Lösung, die durch eine Kombination aus Hard- und Software Echtzeitanalysen oder -transaktionen unterstützen soll. Es handelt sich dabei weiterhin um eine relativ junge Technologie, erst Ende November 2010 wurde die erste SAP HANA-Anwendung ausgeliefert. Dennoch ist die aktuelle Version SPS 10 (Support Package Stack 10), was weiterhin zeigt, dass im Laufe der letzten fünf Jahre einige Veränderungen/Verbesserungen durchgeführt wurden.

Auf der Softwareseite stellt das SAP HANA System, wie auch schon frühere Systeme, zeilen- und spalten-, aber auch objektorientierte Möglichkeiten zur Speicherung von Daten zur Verfügung(vgl. [Kel13]). Der bedeutende Unterschied des SAP HANA Systems zu klassischen Systemen wie Oracle Datenbanken oder ähnlichen ist, dass das SAP HANA System In-Memory Technologie verwendet.

Die Idee für In-Memory Technologie hat sich aus dem Problem entwickelt, dass bei klassischen Datenbanksystemen für eine Berechnung die benötigten Daten von Festplatten in den Arbeitsspeicher geladen werden müssen. Nachdem die Berechnung durchgeführt wurde, wird das Ergebnis wieder zurück auf die Festplatte geschrieben. Jeder dieser Lese- und Schreibzugriffe vom Arbeitsspeicher auf die Festplatte und umgekehrt kostet sehr viel Zeit und da nicht nur am Anfang und Ende einer Berechnung solche Zugriffe erfolgen, stellt die Verwendung von Festplattenspeicher und Arbeitsspeicher ein sog. Bottleneck dar. Um dieses Problem zu umgehen ist es die Idee den kompletten Festplattenspeicher durch Arbeitsspeicher zu ersetzen. Dieses Vorgehen steigert die Performance der Datenbanksysteme enorm und ermöglichst es Berechnungen nahe Echtzeit durchzuführen.

Aber nicht nur der Wechsel von Festplattenspeicher zu Arbeitsspeicher ermöglicht die schnellen Berechnungen, sondern auch die Anzahl der Rechenkerne, die dem System zur Verfügung stehen. Das System beispielsweise, welches der Projektgruppe zur Verfügung gestellt wurde, hatte für die 240 GB Arbeitsspeicher maximal 100 Rechenkerne, die genutzt werden konnten.

4.2 SAP HANA Architektur und Schnittstellen

In diesem Abschnitt wird die Architektur, welche in Abbildung 13 graphisch dargestellt wird, des SAP HANA Systems näher betrachtet.

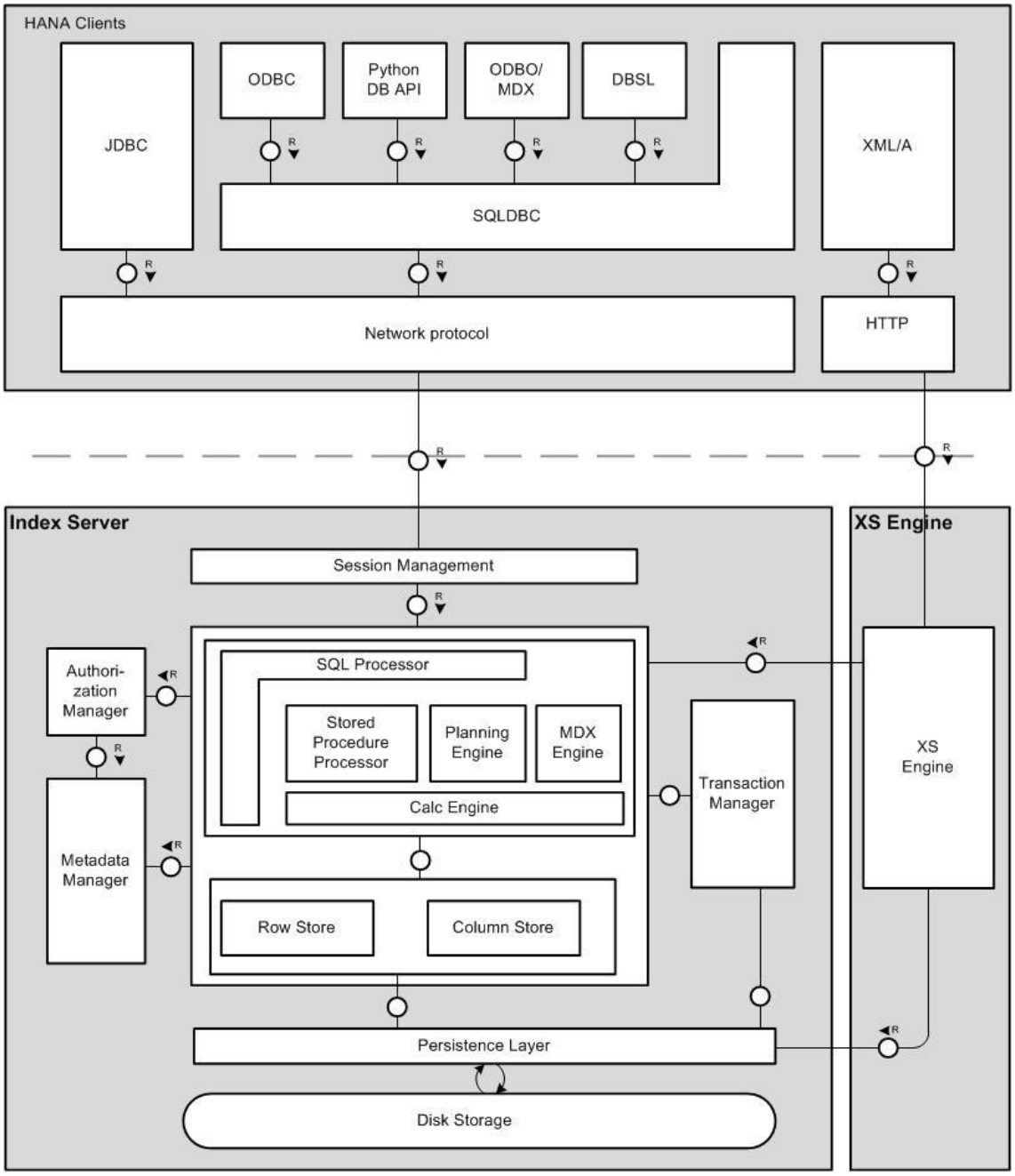


Abbildung 13: Architektur der HANA und der Zugriffsmöglichkeiten auf die HANA, Quelle: nach [Ill12]

Der Abbildung ist zu entnehmen, dass die Architektur in drei Bereiche aufgeteilt werden kann. Auf der einen Seite gibt es den Bereich *HANA Clients*. Von diesem aus kann ein Benutzer durch verschiedene Schnittstellen auf das SAP HANA System zugreifen. Für

den von SAP empfohlenen Zugriff wird darüber hinaus die *XS Engine* verwendet. In Abschnitt 4.2.1 wird deswegen näher darauf eingegangen, welche Möglichkeiten es bei der Anwendungsentwicklung gibt auf ein SAP HANA System zuzugreifen. Der dritte große Bereich ist der *Index Server*, welcher sich beispielsweise um die Verwaltung der Daten oder die Authentifizierung kümmert. Eine nähere Betrachtung des *Index Servers* findet in Abschnitt 4.2.2 statt.

4.2.1 Anwendungsentwicklung mit einem SAP HANA System

Bei der Anwendungsentwicklung versucht SAP durch die Einführung der *XS Engine* (SAP HANA Extended Application Services) einen Paradigmenwechsel, dargestellt in Abbildung 14, hervorzurufen. Auf der linken Seite der Abbildung wird die Anwendungsentwicklung durch die klassische Modell-View-Controller-Aufteilung dargestellt.

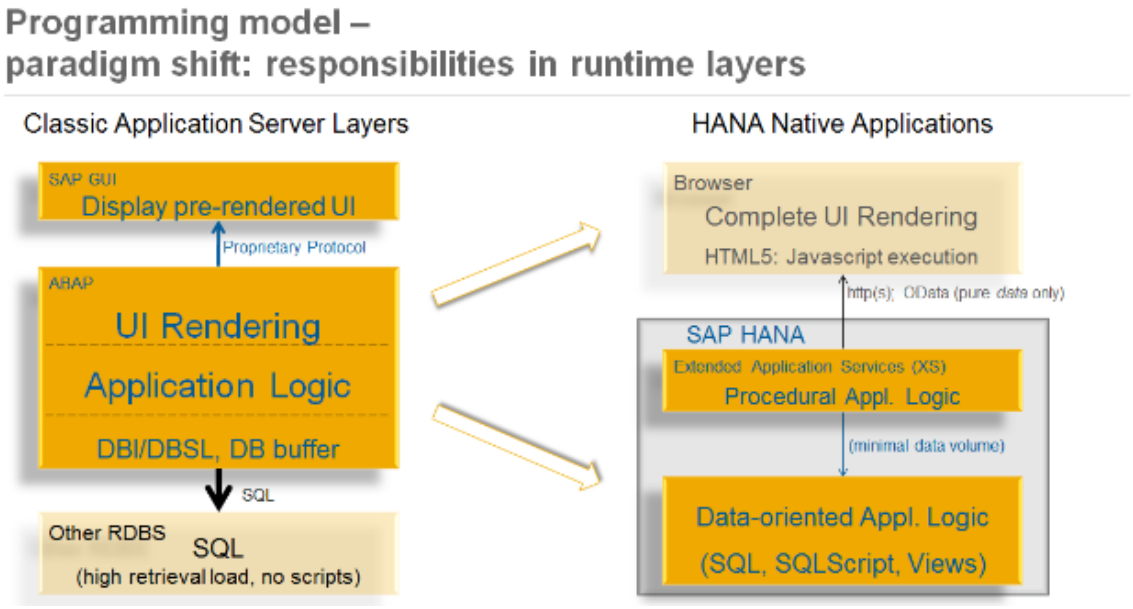


Abbildung 14: Paradigmenwechsel durch die Einführung der XS Engine, Quelle: nach [Jun12]

Auf der rechten Seite der Abbildung ist der Paradigmenwechsel dargestellt. Durch die Einführung der XS Engine, welche einen vollwertigen Anwendungsserver, Webserver und eine Entwicklungsumgebung in das SAP HANA System integriert, wird die Anwendungsentwicklung in zwei Bereiche aufgeteilt. Zum einen gibt es die Darstellung der Anwendung in einem Browser, realisiert durch HTML5 und Javascript. Auf der anderen Seite wird alles, was mit dem Laden und der Verarbeitung der Daten zu tun hat auf dem SAP HANA Server ausgeführt. Dies bringt vor allem Vorteile bei der Performance mit sich, da die Daten zum einen nicht erst auf einen anderen Server geladen werden müssen, sondern direkt vorhanden sind. Hier spielt auch wieder die In-Memory Technologie eine große Rolle. Und zum anderen können bei der Datenverarbeitung über die klassischen SQL-Statements

hinaus auch andere, auf das SAP HANA System angepasste und optimierte Verfahren wie SQLScripts oder Views verwendet werden.

Die Verbindung zwischen der Darstellung und der XS Engine, respektive dem gesamten SAP HANA System kann über OData, serverseitiges JavaScript oder XMLA durchgeführt werden. In Abbildung 15 sind die einzelnen Komponenten und die dort verwendeten Programmiersprachen der Anwendungsentwicklung die SAP vorschlägt, aufgeführt.

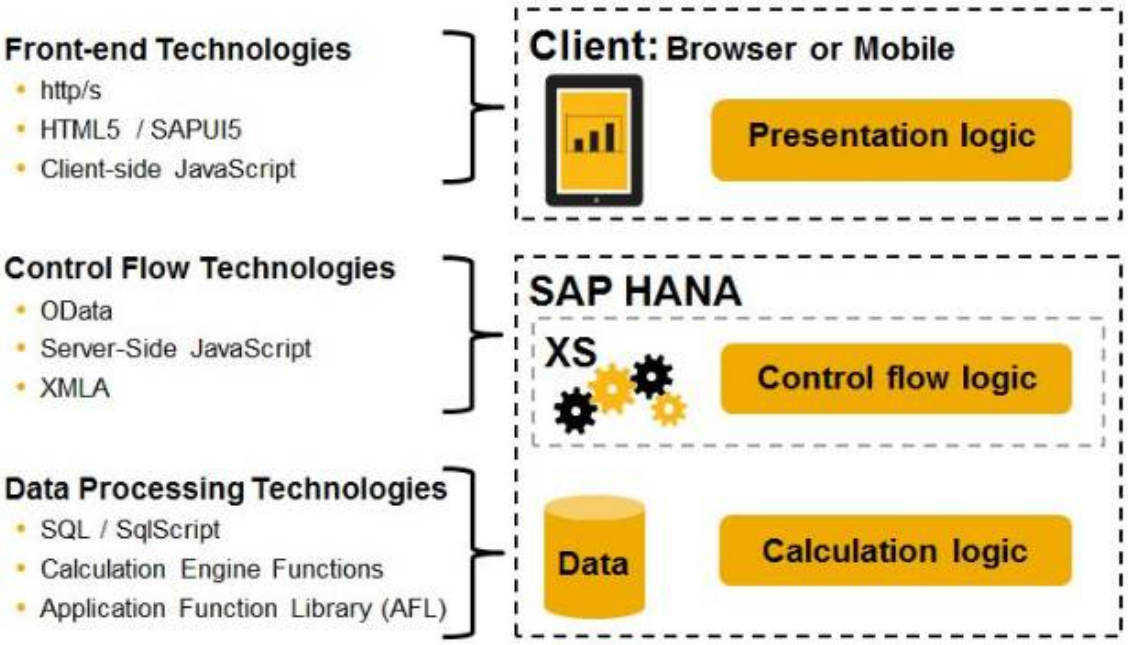


Abbildung 15: Programmiermodell durch die Einführung der XS Engine, Quelle: nach [Jun12]

Neben der Anwendungsentwicklung über die XS Engine ist es auch möglich auf das SAP HANA System von klassischen Programmiersprachen wie Java oder C/C++ aus zuzugreifen. Der Zugriff wird dann über die entsprechenden Schnittstellen wie JDBC für Java oder ODBC für C/C++ basierte Systeme durchgeführt. Der Projektgruppe stand der Zugriff auf das SAP HANA System über die XS Engine nicht zur Verfügung, weswegen es notwendig war einen Zugriff von PHP aus, in welcher das Portal programmiert wurde, über eine ODBC Schnittstelle einzurichten. Welche Schritte dabei unternommen wurden wird in Abschnitt 3.2.5 erläutert.

4.2.2 Index Server

Der Index Server umfasst sieben große Komponenten, die verschiedene Funktionalitäten des SAP HANA Systems bereitstellen.

1. Das *Session Management* ist dafür da, von Clients gesendete Anfragen zu verwalten und die Verbindung zur Datenbank herzustellen. Anfragen können dabei entweder

von dem SAP HANA System authentifiziert werden, per Benutzername und Passwort, oder an externe Dienste delegiert werden.

2. Wenn die Anfrage autorisiert wurde, wird sie an den *SQL Processor* weitergeleitet. Innerhalb des SQL Processors werden die Anfragen, je nach Typ an vier weitere Engines weitergeleitet

   - Der *Stored Procedure Processor* wertet Anfragen aus, die auf bereits vorher definierte und optimierte Prozeduren zugreifen.
   - Die *Planning Engine* ist dafür da Planungsanwendungen zu erstellen. Ein Beispiel dafür wäre eine Finanzplanung für das nächste Jahr. Dafür werden in einem einfachen Fall die Daten des alten Jahres kopiert und mit Filtern manipuliert.
   - Multidimensional Expressions (MDX) ist eine Sprache für die Anfrage und Manipulation von multidimensionalen Daten, die in OLAP-Würfeln gespeichert werden. Um Anfragen in dieser Sprache kümmert sich die *MDX Engine*.
   - Die *Calculation Engine* ist dafür da, Anfragen in Calculation Models umzuwandeln und bemüht sich dabei einen hohen Grad an Parallelisierung zu erreichen.

3. Nachdem die Anfragen durch den SQL Processor optimiert wurden, wird auf den Datenspeicher zugegriffen um die benötigten Daten zu laden. Die Daten, die hier angefragt werden, können auf zwei Arten gespeichert sein und sind vollständig im Hauptspeicher vorhanden. Um die Vorteile der beiden Arten von Speicherung zu zeigen soll Tabelle 1 als Beispieldatenbank dienen.

Jahr	**ProfitGes**	**ProfitDeu**	**ProfitCH**	**ProfitA**
2010	14	6	5	3
2011	14	6	6	2
2012	16	7	5	4
2013	16	7	5	4
2014	22	10	7	5

Tabelle 1: Beispiel Tabelle mit dem Gewinn eines fiktiven Unternehmens, aufgeteilt nach dem Jahr, dem Gesamtprofit (ProfitGes) und dem Profit der Länder Deutschland (Deu), Schweiz (CH) und Österreich (A)

   - Wenn die Daten im *Row Store* abgespeichert werden, werden die einzelnen Zellen reihenweise in den Speicher eingetragen. Für die Beispieltabelle bedeutet das, dass zunächst das Jahr in den Speicher geschrieben wird, gefolgt von dem Gesamtprofit für das entsprechende Jahr und so weiter. Der Vorteil dieser Art die Daten zu speichern ist, wenn beispielsweise für ein bestimmtes Jahr die einzelnen Profite verwendet werden sollen. Da diese im Speicher direkt nebeneinander liegen, kann der Zugriff sehr schnell ausgeführt werden.

- Wenn die Daten hingegen im *Column Store* gespeichert werden, werden sie spaltenweise in den Speicher geschrieben. Für die Beispieltabelle würde das bedeuten, dass zunächst alle Jahre hintereinander in den Speicher geschrieben werden, dann alle Werte für den Gesamtprofit und so weiter. Der Vorteil dieser Art die Daten zu speichern ist, wenn beispielsweise alle Profitzahlen für Deutschland ausgegeben werden sollen. Da diese Werte im Speicher direkt nebeneinander stehen, kann der Zugriff mit geringem Zeitaufwand durchgeführt werden.

4. In Abbildung 13 befinden sich links vom SQL Processor zwei weitere Komponenten. Die eine davon ist der *Authorization Manager*. Diese Komponente wird von anderen Komponenten angefragt um zu prüfen, ob der Nutzer, der die initiale Anfrage gestartet hat, die benötigten Rechte hat, um die angeforderte Operation durchzuführen. Diese Rechte können entweder an einzelne Benutzern oder an Rollenprofile vergeben werden. Sie erlauben dem jeweiligen Benutzer bestimmte Operationen (beispielsweise erstellen, updaten, etc.) auf bestimmte Objekte (beispielsweise Tables, Views, etc.) durchzuführen.

5. Die zweite Komponente auf der linken Seite des SQL Processor ist der *Metadata Manager*. Diese Komponente enthält Metadaten über eine Vielzahl an Objekten, die in dem SAP HANA System gespeichert sind. Beispielsweise werden dort die Definitionen von Tabellen oder Views, aber auch von SQLScript-Prozeduren gespeichert.

6. Auf der rechten Seite vom SQL Processor befindet sich in Abbildung 13 der *Transaction Manager*. Innerhalb des SAP HANA Systems werden einzelne SQL Anfragen als Transaktionen bezeichnet. Diese Komponente kontrolliert und koordiniert diese Transaktionen. Dazu gehört es unter anderem relevante Daten an die entsprechenden Engines zu senden und diese darüber zu informieren, dass eine Aktion ausgeführt werden soll.

7. Am unteren Ende des Index Servers befinden sich in Abbildung 13 noch zwei weitere Komponenten. Die erste davon ist der *Persistance Layer*. Diese Komponente ist dann wichtig, wenn das SAP HANA System entweder geplant oder ungeplant neu gestartet wird. Deswegen werden hier alle 5 bis 10 Minuten so genannte *Save Points* erstellt, um bei dem Neustart einen relativ aktuellen Stand des SAP HANA Systems wieder herstellen zu können. Außerdem ist eine Kommunikation zwischen dem *Persistance Layer* und dem *Transaction Manager* wichtig, um bei einem Neustart die Atomarität der Datenbank zu gewährleisten.

8. Die letzte Komponente, die in Abbildung 13 dem Index Server zuzuordnen ist, ist der *Disk Storage*. Diese Komponente ist traditioneller Festplattenspeicher, der als Speicher für alte Daten genutzt werden kann, die nicht mehr benötigt werden oder als Back-up Speicher für den Fall eines Desasters oder anderer Zwischenfälle.

5 Datenstruktur

Im Folgenden ist nochmals eine Zusammenfassung aller Stored Procedure in Verbindung mit den entsprechend manipulierten Tabellen dargestellt. Somit kann ein detaillierter Überblick über die Datenstruktur gewährleistet werden, sodass sämtliche Zusammenhänge dem Leser auf dem folgenden Schaubild ersichtlich werden. Hierbei stehen die Rhomboiden für einzelne Stored Procedures, die Quelltabellen nutzen, um entsprechende neue Daten bzw. Informationen in Zieltabellen schreiben, die anschließend entsprechend weiterverarbeitet werden können.
Das Diagramm ist ausgehend vom Zentrum, der Tabelle *ZAEHLSCHLEIFEN* zu lesen, da es zwei Hauptäste gibt die sich entsprechend entwickelt haben. Der nach oben gerichtete Ast beschreibt die langfristige Prognose, wohingegen der untere Ast die kurzfristige Prognose darstellt. Bei der kurzfristigen Prognose, werden die 3 Zeitpunkte von 15-, 30- und 60 Minuten in der Zukunft vorhergesagt. Dies wird mit Hilfe der Stored Procedure *calculation_XX* erreicht, wobei XX für 15, 30 oder aber 60 ersetzt werden kann. Diese Stored Procedures nutzen eine Tabelle *PREDICTON_XX*, welche als Zwischenspeicher für vorgenerierte Werte dient und essentiell für die finale Berechnung der Vorhergesagten Werte ist. Aus Systemseitigen Gründen wurde hierfür eine fixe Tabelle anstelle einer temporären Tabelle verwendet. Die untere Tabelle *PREDICTION*, welche die finale Tabelle ist, enthält Werte, die durch die entwickelten Algorithmen generiert und anschließend dem Front-End zur visuellen Ausgabe übergeben werden. Dies ist ebenfalls nochmals in Kapitel 8.5.1 nachzulesen.
Im Gegensatz zur kurzfristigen Prognose ist der Ast der langfristigen Prognose nach oben gerichtet, geht jedoch ebenfalls von der Tabelle *ZAEHLSCHLEIFEN* aus. Die erste Stored Procedure, *averager_group_X* generiert eine Tabelle, welche die Durchschnittswerte über die gesamte zur Verfügung stehende Datenbasis generiert und die Tageszeit in 30 Sekunden Intervallen als Maßstab anlegt. Hierbei steht das X für einen Integer zwischen 1 und 4, da dies, wie in Kapitel 8.5.2 beschrieben für die einzelnen definierten Tagesgruppen steht. Der Output dieser Stored Procedure sind entsprechend die 4 Tabellen *AVERAGES_GROUP_X*. Diese Tabellen wurden jedoch durch die Stored Procedure *Table_day_time_fill* vorbereitet, welche die entsprechenden Tagesintervalle in die Spalte *TIMESTAMP* schreibt, sodass der anschließende Algorithmus mit einer *UPDATE* Funktion sämtliche Zeilen befüllen kann.
Die Procedures *createpvXtable* (wobei *X* wieder für eine Zahl zwischen 1 und 4 steht) befüllt die Tabellen *AVERAGEVALX*, die zuvor einmalig durch die Procedures *FILLTIMEX* mit allen Tageszeiten in 30-Sekunden-Abständen (00:00:00-23:59:30) befüllt wurden. In diesen Tabellen stehen die Durchschnittswerte der einzelnen Zählschleifen zu jedem Tageswert, welcher in der korrespondierenden Gruppe liegt. D.h. für jede Zählschleife werden über allen Zählwerten zu einer jeden Tageszeit, der Durchschnitt gebildet. Dieser wird in die Tabellen *AVERAGEVALX* eingetragen. Über die Procedure *GETDATA* werden dann die Tabellen *PREDVALX* mit Werten befüllt. Diese Werte werden auf Grundlage der Procedure *ZAEHL* (nicht im Diagramm ersichtlich), die in *GETDATA* eingebettet ist, auf Grundlage eines Regressionsmodells befüllt. Das Modell bezieht die Werte aus den Tabellen *AVERAGEVALX* und berechnet die Vorhersagewerte auf Grundlage der Zeitwerte

(siehe Kapitel 8.5.2). Diese werden dann aus den Tabellen *PREDVALX* Frontend-seitig bezogen, um sie in den Darstellungen der Plattform sichtbar zu machen.

Business Rules

Entität	Beschreibung	Attribute	Identifikator
Zaehlschleifen			
AVERAGES_GROUP_1			
AVERAGES_GROUP_2			
AVERAGES_GROUP_3			
AVERAGES_GROUP_4			
PREDICTION			
PREDICTION_15			
PREDICTION_30			
PREDICTION_60			
AVERAGEVAL1			
AVERAGEVAL2			
AVERAGEVAL3			
AVERAGEVAL4			
PREDVAL1			
PREDVAL2			
PREDVAL3			
PREDVAL4			

Beziehung	Beschreibung	Beteiligte Entitäten	Attribute
averager_group_1		Zaehlschleifen (0,1), AVERAGES_GROUP_1 (0,1)	
averager_group_2		Zaehlschleifen (0,1), AVERAGES_GROUP_2 (0,1)	
averager_group_3		Zaehlschleifen (0,1), AVERAGES_GROUP_3 (0,1)	
averager_group_4		Zaehlschleifen (0,1), AVERAGES_GROUP_4 (0,1)	
calculation_15_quick		PREDICTION (0,1), PREDICTION_15 (0,1), Zaehlschleifen (0,1)	
calculation_30		Zaehlschleifen (0,1), PREDICTION (0,1), PREDICTION_30 (0,1)	
calculation_60		Zaehlschleifen (0,1), PREDICTION_60 (0,1), PREDICTION (0,1)	
Table_Day_Time_fill		AVERAGES_GROUP_1 (0,1), Zaehlschleifen (0,1), AVERAGES_GROUP_2 (0,1), AVERAGES_GROUP_3 (0,1), AVERAGES_GROUP_4 (0,1)	
insert_columnnames		Zaehlschleifen (0,1), PREDICTION (0,1)	
createpv1table		AVERAGES_GROUP_1 (0,1), AVERAGEVAL1 (0,1)	
createpv2table		AVERAGES_GROUP_2 (0,1), AVERAGEVAL2 (0,1)	
createpv3table		AVERAGES_GROUP_3 (0,1), AVERAGEVAL3 (0,1)	
createpv4table		AVERAGES_GROUP_4 (0,1), AVERAGEVAL4 (0,1)	
fill_time		PREDVAL1 (0,1), PREDVAL2 (0,1), PREDVAL3 (0,1), PREDVAL4 (0,1)	
getdata		AVERAGEVAL1 (0,1), AVERAGEVAL2 (0,1), AVERAGEVAL3 (0,1), AVERAGEVAL4 (0,1), PREDVAL1 (0,1), PREDVAL2 (0,1), PREDVAL3 (0,1), PREDVAL4 (0,1)	
calculation_15		PREDICTION_15 (0,1), PREDICTION (0,1)	
cleaning_columns		Zaehlschleifen (0,1)	

Abbildung 16: Business Rules

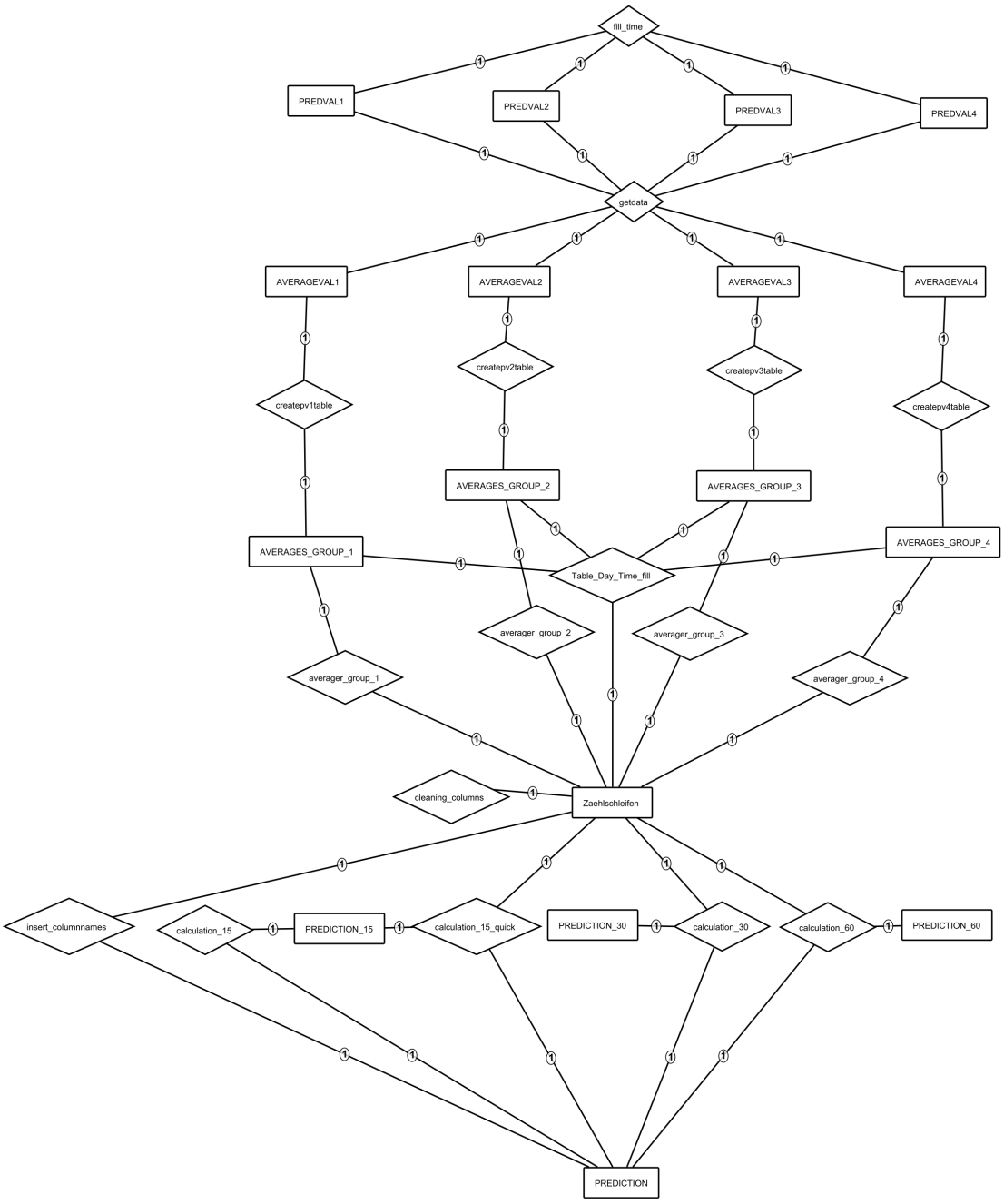


Abbildung 17: ER-Diagramm SAP HANA

6 Datenverständnis

Die zweite Phase im CRISP Data Mining, dem Data Understanding, befasst sich mit den zur Verfügung stehenden Daten. In dieser Phase wird festgelegt, welche Daten für die Fortführung des Projekts relevant sind und welche Daten zur Verfügung stehen (vgl. [Cle14] S. 7). Weiterhin können in dieser Phase zusätzliche Daten aus internen oder externen Quellen beschaffen werden, soweit diese für die Fortführung der Analyse notwendig sind (vgl. [Gab09] S. 127). Es ist sehr wichtig, die Daten und ihre Bedeutung für die weitere Untersuchung genauer zu verstehen. Die Untersuchung der Daten dient auch dazu deren Quantität und Qualität einzuschätzen (vgl. [Cle14] S. 7). In dieser Phase verschafft sich der Data-Miner somit einen genauen Überblick, über die Datentypen, Datenquellen, die Datenmenge, sowie die Datenqualität. Die zweite Phase unterteilt sich in vier Schritte, die im Folgenden genannt und anschließend beschrieben werden:

- Daten sammeln
- Daten beschreiben
- Daten untersuchen
- Datenqualität (vgl. [IBM15] S. 14-20)

6.1 Sammeln ursprünglicher Daten

Im ersten Schritt der zweiten Phase werden sämtliche Daten zunächst gesammelt. Diese Daten können aus bestehenden-, erworbenen- oder zusätzlichen Daten stammen. Zu den bestehenden Daten gehören Transaktionsdaten, Webprotokolle, Umfragedaten etc., die dem Projekt bereits zur Verfügung stehen. An dieser Stelle wird geprüft, ob die vorhandenen Daten für die Untersuchung ausreichen. Bei den erworbenen Daten handelt es sich um unternehmensinternen Daten, die im Unternehmen vorhanden sind, aber in das Projekt noch nicht eingebunden wurden. Diese Daten können soweit sie erforderlich sind, angefordert und in das Projekt für die Untersuchung eingebunden werden. Die zusätzlichen Daten werden von externen Quellen oder durch Umfragen ergänzt, wenn die bestehenden- und erworbenen Daten für die Erfüllung der Anforderungen nicht ausreichen (vgl. [IBM15] S. 14-20). Während dem Sammeln der Daten können folgenden Kriterien berücksichtigt werden:

- Sind genügend Daten für die Erstellung von Prognosen vorhanden?
- Sind Attribute in der Datenbank vorhanden, die einen großen Erfolg versprechen?
- Sind Attribute vorhanden, die irrelevant sind und daher ausgeschlossen werden können?
- Sollen Methoden für die Behandlung von fehlenden Werten entwickelt werden?

6.2 Beschreiben von Daten

Im zweiten Schritt wird ein Bericht über die Eigenschaften, Quantität und Qualität der gesammelten Daten erstellt und festgehalten. Zu den Eigenschaften gehören Wertetypen und Kodierungsschemata.

- Wertetypen:
  - Die vorhandenen Daten können in verschiedenen Formaten boolesch (wahr/falsch), nummerisch oder kategorial (Zeichenkette), vorliegen. Damit es bei einer späteren Verwendung der Daten keine Probleme entstehen, ist es von Bedeutung, dies in dem Bericht aufzunehmen.
- Kodierungsschemata:
  - In den Datenbanken werden die gesammelten Werte unterschiedlich dargestellt. So wird z.B. in einem Daten-Set für das Merkmal Geschlecht, die Werte M für Männlich und W für Weiblich verwendet. Dagegen werden diese Werte in ein anderes Daten-Set, durch die nummerischen Werte 1 und 2 genutzt (vgl. [IBM15] S. 14-20).

Während der Erstellung eines Berichtes zur Datenbeschreibung, können weiterhin folgende Fragen berücksichtigt werden:
„Datenquantität

- In welchen Formaten liegen die Daten vor?
- Ermitteln Sie die Methode, mit der die Daten erfasst wurden, z.B. ODBC.
- Wie groß ist die Datenbank (Anzahl der Zeilen und Spalten)?“ ([IBM15] S. 17).

6.3 Untersuchen von Daten

Im dritten Schritt können die gesammelten Daten mit Hilfe von Visualisierungsmethoden untersucht werden. Durch die Visualisierung von Daten, erhält der Data-Miner die ersten Einblicke in die Struktur der Daten, wodurch der Data-Miner Schlussfolgerungen aus den Daten ziehen kann (vgl. [Kei15] S. 1-6). Die Ziele der Datenvisualisierung sind:

- Beziehungen unter den Attributen und Korrelationen entdecken (vgl. [Kei15] S. 1-6)
- Die Erkennung von Zusammenhänge, Mustern und Strukturen
- Entdeckung von neuen Erkenntnissen (vgl. [IBM15]] S. 14).

Die Analyse der Daten in diesem Schritt unterstützt das Data Mining-Ziel, das während der ersten Phase (Geschäftsverständnis) erarbeitet wurde, in Betracht zu ziehen. Weiterhin hilft die Analyse bei der Gestaltung der Datentransformationsaufgaben die während der Datenvorbereitungsphase durchgeführt werden (vgl. [IBM15] S. 17-18).

6.4 Datenqualität

Der Begriff „Datenqualität“ wird wie folgt definiert: „Datenqualität ist die Gesamtheit der Ausprägungen von Qualitätsmerkmalen eines Datenbestandes bezüglich dessen Eignung, festgelegte und vorausgesetzte Erfordernisse zu erfüllen“([Hil15] S. 88). Die Datenqualität kann beispielsweise anhand der folgenden fünf Kategorien bewertet werden:

1. Gültigkeit
2. Genauigkeit
3. Vollständigkeit
4. Konsistenz
5. Konstanz

Die Gültigkeit als erste Kategorie wird anhand folgender acht Attribute bewertet:

1. *Datentyp Bedingungen* (engl. Data-Type Constraints) geben an, ob bestimmte Attribute nur von einem bestimmten Typ (NVARCHAR, INT, etc.) sein dürfen. Beispielsweise kann die Anzahl von verkaufen Produkten nicht vom Typ DECIMAL sein, da keine halben Produkte verkauft werden können.
2. *Wertebereich Bedingungen* (engl. Range Constraints) geben an, ob die Werte für bestimmte Attribute innerhalb eines bestimmten Bereichs liegen müssen. Beispielsweise kann eine numerische Angabe des Monats nur zwischen eins und zwölf liegen.
3. Notwendigkeitsbedingungen (engl. Mandatory Constraints) geben an, ob ein Attribut einen Wert enthalten muss. Beispielsweise kann in einer Kundendatenbank ein Name mehrmals vorkommen. Deswegen ist es in den meisten Fällen erforderlich dem Kunden eine ID zuzuweisen um eine eindeutige Identifizierung durchführen zu können. Dieses Feld darf dann für keinen Kunden leer sein.
4. *Einzigartigkeitsbedingungen* (engl. Unique Constraints) geben an, ob ein bestimmter Wert pro Feld, Bereich oder im Allgemeinen nur ein einziges Mal vorkommen darf. Auch hier kann die Kunden-ID als Beispiel dienen, da sie für jeden Kunden einzigartig sein sollte.
5. *Mitgliedschaftsbeziehungen* (engl. Set-Membership Constraints) geben an, ob der Wert eines Attributes diskret ist. Beispielsweise kann der Wochentag nur durch einen der sieben Tage Montag bis Sonntag angegeben werden.
6. *Fremdschlüssel Beziehungen* (engl. Foreign-key Constraints) geben an, ob bestimmte Werte eines Attributes wiederum in einer anderen Tabelle hinterlegt sind um die Übersichtlichkeit zu verbessern. Beispielsweise kann bei der Adresse die Aufschlüsselung der Postleitzahl in einer separaten Tabelle hinterlegt sein.

7. *Regular expression Patern* (engl. Regular expression patterns) geben an, ob die Werte eines Textfeldes in einem bestimmten Format hinterlegt sein müssen um über Regular Expressions ausgewertet werden zu können. Beispielsweise kann im amerikanischen die Telefonnummer im Format (999) 999-9999 hinterlegt werden.

8. *Validierung* (engl. Cross-field validation) gibt an, ob bestimmte Felder bestimmte Bedingungen erfüllen müssen. Beispielsweise muss die Summe alle prozentualen Anteile von Automarken in Deutschland 100 ergeben. Die Genauigkeit gibt an, wie sehr die erfassten Werte an einen Standard oder „wahren“ Wert herankommen. Das Problem dabei ist allerdings, dass der „wahre“ Wert nicht unbedingt bekannt ist, weswegen dieses Problem teilweise nicht behoben werden kann.

Das nächste Kriterium ist die Vollständigkeit, also wie vollständig die Daten sind. Auch dieses Kriterium ist fast unmöglich (exakt) zu beheben, da Daten, die nicht erfasst wurden höchstens angenommen werden können. Das vierte Kriterium ist die Konsistenz. Dieses Kriterium gibt an, wie einheitlich die Messwerte innerhalb unterschiedlicher Systeme sind. Beispielsweise kann derselbe Kunde in zwei Kundendatenbanken mit unterschiedlicher Adresse auftauchen, allerdings kann nur eine davon stimmen. Probleme dieser Art sind durch die Datenbereinigung nicht immer zu beheben, da eine Entscheidung getroffen werden muss, welches Datum als rrichtigängenommen wird. Für eine solche Entscheidung können dann verschiedene Faktoren herangezogen werden. Welche Datenbank ist aktueller, welche Quelle ist vertrauenswürdiger und vieles mehr. Das letzte Kriterium ist die Konstanz und wird dadurch bestimmt, wie einheitlich die Messwerte bezüglich ihrer Einheit in unterschiedlichen Systemen sind. Ein Beispiel dafür stellt die Messung von Geschwindigkeiten in km/h oder mph dar. Wenn beispielsweise Autos in einer Datenbank bezüglich ihrer Geschwindigkeiten verglichen werden sollen, müssen zunächst alle Geschwindigkeiten in die gleiche Einheit umgerechnet werden

6.5 OpenStreetMap

Eine essentielle Darstellungsgrundlage für Verkehrsdaten bildet die OpenStreetMap Basis, ein freies Projekt, mit dem Ziel weltweite Geo-Daten über Straßen, Eisenbahnen, Flüssen, Wäldern, Häuser oder ähnlichen Objekten auf Kartenbasis zu sammeln. Die dafür eingesetzten Ressourcen obliegen einer „Open Database License“ und sind im Gegensatz zu unfreien proprietären Kartenmaterialien wie das Angebot von Google-Maps ohne Einschränkungen zugänglich und nutzbar. Geoinformationen sind in den meisten Fällen selten frei erhältlich. Aufgrund der Vielzahl von weltweit agierenden ehrenamtlichen Kartografen (Mappern) ist OpenStreetMap eine leistungsstarke Community die einem stetigen Wandel unterliegt. Rohdaten bestehend aus Vektordaten und Tags werden mittels eines Editors aufbereitet und durch das OSM-XML-Protokoll dargestellt. Diese Daten können bearbeitet und anschließend auf dem Server (OSM API) hochgeladen und in der Datenbank gespeichert werden. Für Kartennutzer besteht die Möglichkeit gerenderte Karten einzusehen und für beliebige Zwecke einzusetzen. Die dabei unerlässliche Qualitätskontrolle und Richtigkeit der Informationen stellen ebenfalls wieder die agierenden Kartografen sicher.

Die folgende Abbildung 18 zeigt die Grundarchitektur die dem eben beschriebenen Ablauf zu Grunde liegt.

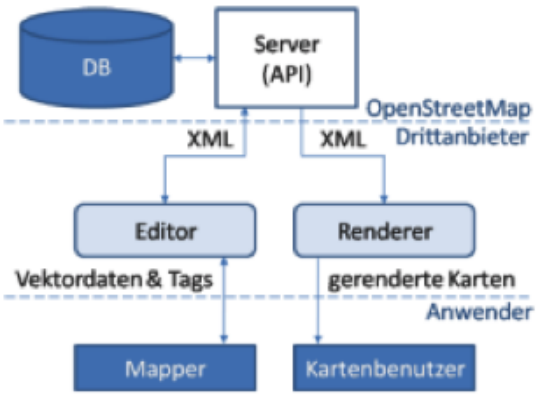


Abbildung 18: OpenStreetMap Grundarchitektur [Wie15]

Zudem ist das OpenStreetMap Projekt gut dokumentiert und bietet neben verschiedensten Veröffentlichungen auch ein eigenes Wiki an. Durch die hohe Fluktuation und Aktualität der Daten ist die Verwendung von OpenStreetMap die favorisierte Darstellungsgrundlage für Verkehrsdaten im Rahmen der Projektgruppe RAPID. Um einen geeigneten Kartenausschnitt über dem Großraum Oldenburg zu erhalten, wird das angebotene OpenStreetMap Exporttool (abrufbar über die Kartendarstellung) verwendet. In der Abbildung 19 werden die genauen Positionsangaben durch die minimalen sowie maximalen geographischen Breiten (Latitude) und Längen (Longitude) angegeben.

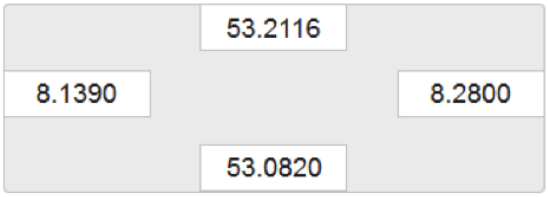


Abbildung 19: OpenStreetMap Positionsangaben

Der daraus resultierende Kartenausschnitt wird in Abbildung 20 dargestellt. Bei der Datenerhebung wurde besonders Wert darauf gelegt, alle Stadtteile Oldenburgs (Zentrum, Dobben/Haarenesch, Gerichtsviertel, Ziegelhof, Eversten, Haarentor, Bloherfelde, Bürgerfelde/Wechloy, Donnerschwee, Osternburg, Krusenbusch, Kreyenbrück, Bümmerstede, Na-

dorst, Ohmstede, Bornhorst, Etzhorn und Ofenerdiek) sowie kleinere Ortschaften im Großraum Oldenburg zu erfassen.

Abbildung 20: OpenStreetMap Kartenausschnitt

Mit Hilfe des angebotenen Exporttools kann durch die „Overpass API“ der angegebene Kartenausschnitt extrahiert und heruntergeladen werden. Die entstandene Darstellungsbasis für Verkehrsdaten liegt im bereits angesprochenen OSM-XML Protokoll vor und spezifiziert gemäß dem XML Standard alle zulässigen Elemente genau.
Das Datenmodell des OpenStreetMap Projekts obliegt einer einheitlichen und flexiblen Gliederung von Elementen wie in Abbildung 21 zu erkennen. Zusätzlich steht eine Versionshistorie zur Verfügung, die über interne ’Changesets’ Änderungen an bestehenden Elementen aufzeigt und die Möglichkeit bietet einen früheren Zustand wiederherzustellen.

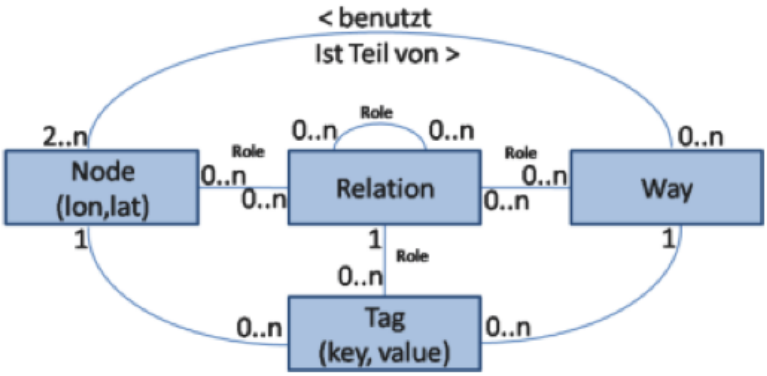


Abbildung 21: Datenmodell von OpenStreetMap [Wie15]

Innerhalb des Datenmodells gibt es drei Hauptelemente „Nodes“ (Koordinaten), „Ways“ (Flächen und Folgen von mindestens 2 aufeinanderfolgenden Nodes) sowie „Relations“ (Gruppierung oder auch übergeordnete Beziehung von Ways untereinander) welche die extrahierte Datengrundlage in übersichtliche Bereiche aufteilen. Jedes Element hat zudem weitere Eigenschaften, die „Tags“ genannt werden. Tags können beispielsweise genauere Informationen, wie die Art eines Weges liefern. Sie besitzen jedoch immer mindestens einen Schlüssel und einen Wert. Im nachfolgenden XML Einschub wird die beispielhafte Darstellung auf der Grundlage des beschriebenen Datenmodells der Oldenburger OSM Daten dargestellt.

```
<meta osm_base="2015-02-12T14:25:02Z"/>
  <bounds minlat="53.0821" minlon="8.1388" maxlat="53.2116" maxlon
     ="8.2858"/>
  <node id="122485" lat="53.1539166" lon="8.1861832" version="8"
     timestamp="2015-02-10T15:22:00Z" changeset="28752588" uid="
     133419" user="dachefte"/>
    <tag k="TMC:cid_58:tabcd_1:Class" v="Point"/>
    <tag k="TMC:cid_58:tabcd_1:Direction" v="negative"/>
    <tag k="TMC:cid_58:tabcd_1:LCLversion" v="9.00"/>
    <tag k="TMC:cid_58:tabcd_1:LocationCode" v="10744"/>
    <tag k="TMC:cid_58:tabcd_1:NextLocationCode" v="10745"/>
    <tag k="TMC:cid_58:tabcd_1:PrevLocationCode" v="10743"/>
    <tag k="highway" v="motorway_junction"/>
    <tag k="name" v="Oldenburg-Eversten"/>
    <tag k="ref" v="13"/>
  </node>
  <node id="122494" lat="53.1284117" lon="8.2031553" version="21"
     timestamp="2013-09-22T11:01:43Z" changeset="17971505" uid="
     323886" user="hermann51"/>
```

```
<node id="122495" lat="53.1279157" lon="8.2106758" version="13"
   timestamp="2013-09-22T11:07:02Z" changeset="17971609" uid="
   323886" user="hermann51"/>
  [...]
```

Insgesamt umfasst die erhobene OSM Datenmenge 1.359.707 Millionen loc (Lines of Code) die im nächsten Verarbeitungsschritt der Datenvorbereitung (Kapitel 7) an die Bedürfnisse des Projektes RAPID angepasst werden müssen.

6.6 Daten der Stadt Oldenburg

Die Idee dieser Projektgruppe ist es eine intelligente Plattform für Mobilitätsdaten zu schaffen. Eine wichtige Grundlage um diese Idee zu gewährleisten stellen die aktuellen Verkehrsdaten dar. Im Rahmen des Prozessschrittes 'Datenverständnis' galt es also zunächst die Verkehrsdaten der Stadt Oldenburg zu beschaffen.

Erste Kontakte um an Verkehrsdaten der Stadt Oldenburg zu kommen wurden vor dem Start der Projektgruppe durch die Betreuer initialisiert, allerdings gab es zunächst Unklarheiten über die endgültige Quelle für die Verkehrsdaten. Als Partner hat sich letztendlich die Verkehrsleitzentrale Oldenburg ergeben mit welcher es am 02.04.2015 zu einem Treffen zwischen der Projektgruppe und dem Ansprechpartner Herr Brandt kam. Bei dem Treffen wurde von Herrn Brandt zunächst eine Einführung in die generelle Arbeitsweise der Verkehrsleitzentrale gegeben und welche Daten alle erfasst werden. Danach wurde über den Umfang der Daten, die der Projektgruppe zur Verfügung gestellt werden könnten und den Datentransfer diskutiert. Für den Umfang ergab die Diskussion, dass der Projektgruppe zunächst die Werte für den Monat März zur Verfügung gestellt werden sollten. Diese wurden seitens der Projektgruppe zunächst dazu genutzt die genaue Struktur der Datentabellen zu erfassen und gegebenenfalls Attribute zu bestimmen, die relevant für die Aufgabenstellung sein könnten. Darüber hinaus wurden ebenfalls das Einladen der Tabellen in das SAP HANA System vorbereitet. Seitens der Verkehrsleitzentrale war der zusätzliche Arbeitsaufwand um die Informationen für die Projektgruppe bereitzustellen ebenfalls akzeptabel. Für den Datentransfer wurde diskutiert, ob es möglich sei einen Datenstream bereitzustellen, was seitens der Verkehrsleitzentrale technisch nicht zu realisieren war. Das Ergebnis für den Datentransfer war, dass Herr Brandt Zugriff auf den Uni-Server der Projektgruppe bekam und die Daten dort in einen vorgegeben Ordner kopiert hat. Am 22.04. standen der Projektgruppe die Daten zur Verfügung, welche Folgendes umfassten:

- **Lagepläne der Zählspulen:** Der Projektgruppe wurden insgesamt 158 Lagepläne als PDF-Datei zur Verfügung gestellt, auf welchen die Positionen der Schaltanlagen grafisch dargestellt wurden. Ein Beispiel für einen Lageplan ist in Abbildung 22 dargestellt. Die Abkürzungen sind wie folgt definiert:
  - **D:** In der Regel einfacher, normaler Detektor zur Anforderung am Haltebalken oder Verlängerung der Grünzeit bei etwa 30m vor dem Haltebalken
  - **T, MT oder nur Ziffern:** Detektor zur Verkehrszählung entweder in der Ausfahrt der Kreuzung oder 80m vor der Kreuzung. Zählt in der Regel besser als Schleifen in der Nähe von Haltebalken

  - **AT:** Anforderungstaster für Fußgänger oder Radfahrer
  - **ATFS:** Anforderungstaster für Freigabesignal für Sehbehinderte. Deswegen zwei Detektoren an einem Mast für normale Fußgänger (AT) und Sehbehinderte (ATFS)

- **Messwerte für die Schaltanlagen:** Der Projektgruppe wurden insgesamt 12 CSV-Dateien mit den erfassten Werten für die Schaltanlagen zur Verfügung gestellt. Die einzelnen CSV-Dateien umfassten jeweils den Zeitraum vom 01.03.2015 bis 31.03.2015 wobei für jede Schaltanlage maximal alle 90 Sekunde neue Werte vorhanden sind. Dieses Zeitintervall ist auf technische Gründe in der Erfassung der Werte zurückzuführen. Rein theoretisch ergeben sich daraus knapp 30.000 Werte pro Schaltanlage, die der Projektgruppe für die Analyse zur Verfügung stehen. Allerdings ist es ebenfalls aus technischen Gründen möglich, dass für einzelne Schaltanlagen über einen längeren Zeitraum keine Werte erfasst wurden.

  Ein Beispielausschnitt aus einer Datei ist in Tabelle 2 dargestellt. In der ersten Spalte wurde jeweils eine ID gespeichert, welche innerhalb der jeweiligen Datei einzigartig ist. Zwischen den einzelnen Dateien waren die IDs allerdings nicht einzigartig. In der zweiten Spalte wird das Datum und die Uhrzeit gespeichert, für welche der jeweilige Wert erfasst wurde. In den ersten drei Zeilen des Beispiels zeigt sich das normale Verhalten, unter welchem für alle 90 Sekunden neue Werte vorliegen. Zwischen der dritten und vierten Zeile allerdings sind 9:30 Minuten vergangen, in welchen keine neuen Daten erfasst wurden. Die dritte und vierte Spalte stellen Beispiele für erfasste Werte dar, wobei der erste Substring *VSA191* bedeutet, dass die Schaltanlage auf Lageplan 191 zu finden ist. *D1*, bzw. *T1* gibt die genaue Schaltanlage an, wobei neben *D*- und *T*-Anlagen wie oben beschrieben auch Werte für *AT*- und *ATFS*-Anlagen aufgenommen wurden. Der letzte Substring *Zaehl*, bzw. *Beleg* gibt die Art der Datenerfassung an. Der *Zaehl*-Wert wird berechnet aus der Anzahl der erfassten Autos in den letzten 90 Sekunden, hochgerechnet auf eine Stunde, bzw. wie oft ein Ampeltaster in den letzten 90 Sekunden betätigt wurde auf eine Stunde hochgerechnet. *Beleg* gibt die relative Belegungsdauer der Zählspule während der letzten 90 sekündigen Periode an.

- **Fahrrad Zählspulen:** Neben den Zählspulen für Autos hat die Stadt Oldenburg an einigen Stellen in der Stadt auch Zählspulen für Fahrräder installiert. Auch die erfassten Werte für diese Zählspulen wurden der Projektgruppe zur Verfügung gestellt, allerdings mit Hinblick auf den Use-Case und aus zeitlichen Gründen zunächst zurückgestellt.

- **Busprotokolle:** Neben Autos und Fährrädern gibt es auch für die Busse Protokolle, die aufzeichnen wann ein Bus bestimmte Wegpunkte passiert. Auch diese wurden der Projektgruppe zur Verfügung gestellt, allerdings auch, wie die Werte der Fahrradzählspulen, mit Hinblick auf den Use-Case und aus zeitlichen Gründen zurückgestellt.

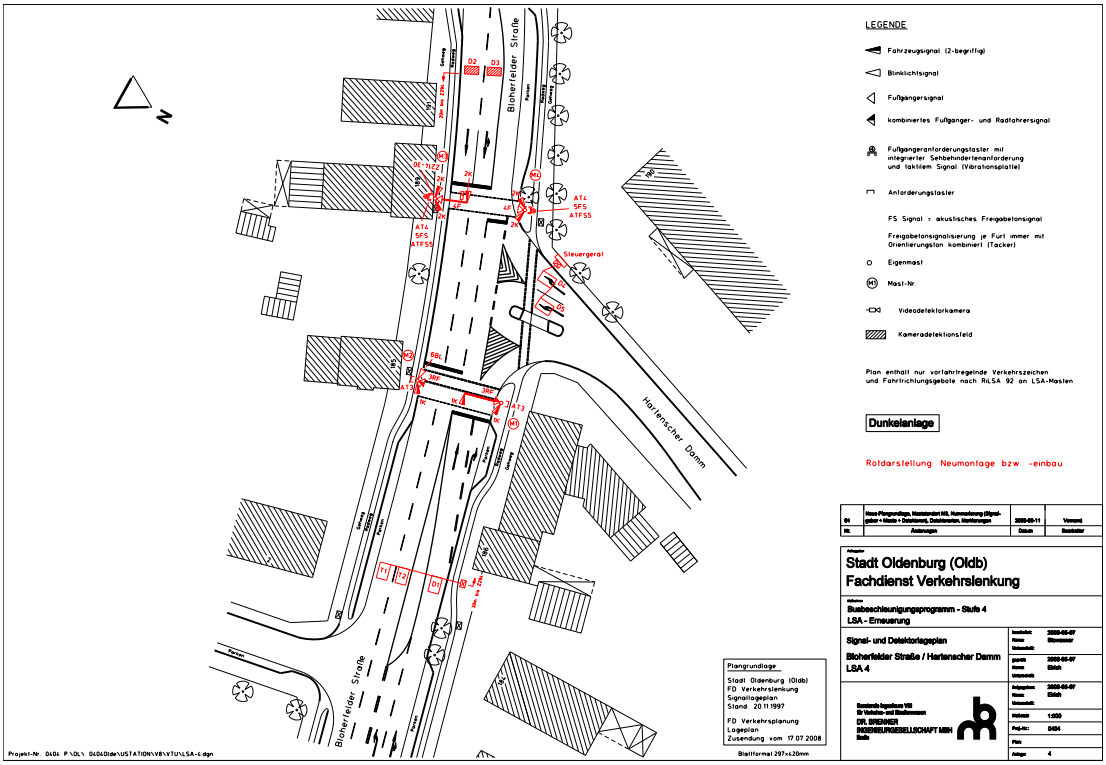


Abbildung 22: Beispiel für einen Lageplan mit den Zählspulen

id	Datum	VSA191_D1_T1_Zaehl	VSA191_D1_T1_Beleg
90768	2015-03-01 00:00:30	136	3
90769	2015-03-01 00:02:00	120	2
90770	2015-03-01 00:03:30	120	2
90771	2015-03-01 00:13:00	150	3

Tabelle 2: Ausschnitt aus der CSV-Datei mit den Zählspulwerten

Wie bereits in den Erklärungen der Daten beschrieben, wurde mit Hinblick auf den Use-Case schon während der Datenerhebung entschieden, dass zunächst nur die Werte für die Auto-Zählspulen betrachtet werden. Die übrigen Daten wurden zunächst zurück gestellt, allerdings ist es durchaus möglich sie in einer späteren Projektgruppe mit einzubinden. Weiterhin wurde entschieden nur die *Zaehl*-Werte zu betrachten, da sie, im Gegensatz zu den *Beleg*-Werten die konkrete Anzahl der Autos pro Zeiteinheit angeben und keinen relativen Wert.

Um mit den Daten im weiteren Verlauf arbeiten zu können, war der nächste Schritt sie in die HANA zu integrieren. Die Recherche zu diesem Thema hat ergeben, dass der einfachste und geläufigste Weg ist, die Daten auf einen mit der HANA verbunden FTP-Server zu laden und anschließend über ein *Control File* in eine bereits vordefinierte Tabelle zu laden[1]. Die vordefinierte Tabelle muss dabei nur die benötigte Struktur aufweisen und keine Daten enthalten. Das *Control File* würde dann Informationen darüber enthalten, welche CSV-Datei in welchen Table geladen werden soll und wie die einzelnen Delimiter für Spalten und Zeilen aussehen. Ein Beispiel für ein Control File sieht dabei wie folgt aus:

```
import data
into table XXX."YYY"
from 'ZZZ.csv'
record delimited by '\n'
fields delimited by ','
optionally enclosed by '"'
error log 'Text_Tables.err
```

Eine Anfrage in Magdeburg, ob der Projektgruppe ein mit der HANA verbundener FTP- oder Cloud-Server zur Verfügung gestellt werden könnten, hat ergeben, dass eine Bereitstellung nicht möglich sei. Deswegen musste nach einer Alternativen gesucht werden um die Daten in der Hana bereitstellen zu können.

Eine praktikable Alternative wurde in dem Import per PHP-Skript über eine ODBC Schnittstelle gefunden. Das eine solche Schnittstelle möglich ist und wie sie eingerichtet werden kann wird in Abschnitt 3.2.5 erläutert. Nachdem die Daten wie in Abschnitt 7.10 beschrieben bereinigt wurden, war das Ergebnis des Importes letztendlich eine einzige Tabelle mit allen erfassten Werten für alle Zählspulen.

Im Zuge der *Datenvorbereitung* und *Datenanalyse* wurde festgestellt, dass für eine bessere Vorhersage weitere Daten benötigt werden. Deswegen wurde die Verkehrsleitzentrale am 22.07.2015 erneut angeschrieben und gebeten uns die Daten für drei weitere Monate zur Verfügung zu stellen. Dieses Anfrage wurde am 03.08.2015 seitens der Verkehrsleitzentrale abgelehnt, weil die nötigen zeitlichen Kapazitäten nicht gegeben waren.

6.7 ADAC Daten

Für die Berechnung der CO2-Belastung der Stadt Oldenburg benötigte die Projektgruppe bestimmte Daten und Werte, um die Schadstoffbelastung zu berechnen und diese später auf dem Portal der Projektgruppe darstellen zu können. Durch die Kommunikation zwischen

[1] vgl. http://wiki.scn.sap.com/wiki/display/inmemory/Importing+CSV+files+into+SAP+HANA

der Projektgruppe und den Projektbetreuern wurde festgestellt, dass die nötigen Daten für die Berechnung der CO2-Belastung in der Projektinternen Abteilung VLBA (Very Large Business Applications), vorliegen. Nach Anfrage bei den Projektbetreuern wurde zunächst ein Testdatensatz zur Verfügung gestellt, um sich damit vorab vertraut zu machen. Der Testdatensatz beinhaltete diverse neue und alte Fahrzeugmodelle der Marke-Opel und stammt ursprünglich aus einer ADAC-Datenbank. Der Umfang der Opel-Datensätze umfassen ca. 7000 Daten und wurde im CSV-Format zur Verfügung gestellt. Die Einträge der ADAC-Datensätze beinhalteten verschiedene Informationen und Werte, die im Folgenden genannt werden:

- Eindeutige ADAC Fahrzeug-ID
- Kennzeichnung auf Modellaktualität
- Marke (Hersteller)
- Fahrzeugmodell
- Fahrzeugtyp
- Produktionsstart des einzelnen Modells
- Produktionsende des einzelnen Modells
- Baureihe
- Produktionsstart der Baureihe
- Produktionsende der Baureihe
- Karosserieart
- Fahrzeugklasse
- Fahrzeug-Grundpreis in Euro
- Gesamtaufpreis für übliche Extras
- Fahrzeug-Listenpreis inkl. Extras
- Motorenart
- Hubraum in ccm
- Leistung in kW
- Leistung in PS
- Getriebeart
- CO2-Emission in Gramm pro KM

- CO2-Effizienzklasse
- Kraftstoffart bzw. Qualität
- Kraftstoffart bzw. Qualität
- Einstufung in Schadstoffklasse
- Leergewicht in kg
- Zulässiges Gesamtgewicht in kg
- Anzahl der Türen
- Anzahl der Sitze
- Anzahl der Sitze maximal
- EU-Verbrauch gesamt
- EU-Verbrauch innerorts
- EU-Verbrauch außerorts
- 2. EU-Verbrauch gesamt
- 2. EU-Verbrauch innerorts
- 2. EU-Verbrauch außerorts
- Gesamtkosten in Euro pro Monat
- Gesamtkosten in Cent pro Kilometer
- Wertverlust in Euro pro Monat
- Gesamte Betriebskosten in Euro pro Monat
- Gesamte Fixkosten in Euro pro Monat
- Gesamte Werkstatt- und Reifenkosten pro Monat
- Jahresbeitrag für die Haftpflichtversicherung
- Jahresbeitrag für die Teilkaskoversicherung
- Jahresbeitrag für die Vollkaskoversicherung

Nachdem der Testdatensatz gesichtet wurde, konnte mit dem Gesamtdatensatz fortgefahren werden. Im Gegensatz zu dem Testdatensatz umfasste der ADAC-Gesamtdatensatz ca. 61.000 Daten und lag im .txt Format vor. Die Daten wurden zunächst in einem CSV-Format umgewandelt, gespeichert und anschließend untersucht. Während der Untersuchung der Daten wurde festgestellt, dass die Daten nicht vollständig sind und mehrere Werte fehlen. Die erworbenen ADAC-Daten reichten für die Ermittlung der Schadstoffbelastung der Stadt Oldenburg nicht aus. Für die Ergänzung der ADAC-Daten wurden zusätzliche Daten von der Webseite des Kraftfahrt-Bundesamtes herangezogen. Die Daten des Kraftfahrt-Bundesamtes lagen in PDF-Format vor und wurden zunächst in CSV-Format umgewandelt, gespeichert und anschließend untersucht. Ebenfalls wie bei den ADAC-Daten wurde auch hier festgestellt, dass einige Daten fehlen und in der folgenden Datenvorbereitungsphase ergänzt werden müssen.

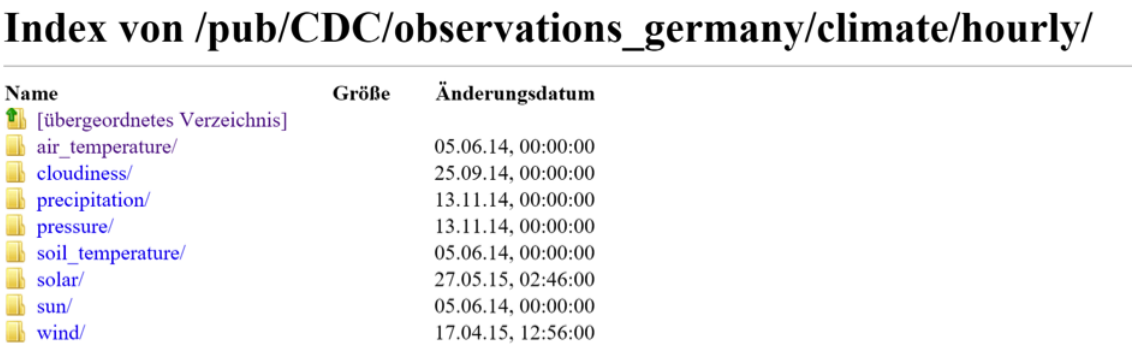
6.8 Feinstaubbelastung

In Oldenburg ist allgemein eine hohe Feinstaubbelastung, verursacht durch den Verkehr zu vermerken. Insbesondere Busse und Dieselmotoren verursachen primär Feinstaub in der Luft. Gerade an dem Heilligengeistwall, mit dem höchsten Busaufkommen, ist vermehrt eine hohe Feinstaubbelastung festzustellen. An dieser Stelle befindet sich in Oldenburg ein Feinstaubmessgerät. Die zahlreichen Messung über Jahre hinweg, zeigen eine Überschreitung des jährlichen Mittelwertes, der laut Statistischem Bundesland bei $40 Mikrogramm/Kubikmeter(\frac{\mu g}{m3}))$ liegt. Der Tagesmittelwert liegt bei $50\frac{\mu g}{m3}$. In Oldenburg liegt der momentane Durchschnittswert der Feinstaubbelastung bei $51\frac{\mu g}{m3}$ (vgl. [sta09]). Aufgrund der hohen Belastung durch den Feinstaub und weiteren Schadstoffen wie CO2, sieht die Projektgruppe hohes Verbesserungspotential für die Stadt Oldenburg und verwendet die vorgegeben Mittelwerte, um in Verbindung mit den Zählspuldaten festzustellen in welchen Regionen der Mitteilwert überschritten wird und wo keine hohe Feinstaubbelastung durch den Verkehr zu vermerken ist. Dadurch können gebietsweise Verbesserungen durch die Stadt Oldenburg eingeleitet werden, um den Feinstaub zu minimieren und nachhaltig die Gesundheit der Bevölkerung in Oldenburg zu stärken.

6.9 Wetter

Einen Einflussfaktor auf den Straßenverkehr stellt das Wetter dar. Damit dieser Faktor in die Betrachtung einbezogen werden kann, muss eine Zugriffsmöglichkeit auf die Daten gefunden werden. Der Deutsche Wetterdienst stellt einen öffentlichen FTP-Zugang zur Verfügung über den aktuelle Informationen zu verschiedenen Umweltfaktoren abgerufen werden können. Die Zugangsdaten sind auf der Internetseite des DWD zu finden. Um die Nutzbarkeit der Daten zu testen werden diese zunächst genauer betrachtet. Über den FTP-Client erfolgt der Zugriff auf den Server des DWD. Die für unsere Region relevanten Daten befinden sich im Ordner /observations_germany/climate/, da die hier vorliegenden Datensätze das Klima in Deutschland darstellen. In diesem Ordner ist der Betrachtungszeitraum zu wählen. Daten liegen in jährlicher, monatlicher, täglicher und stündlicher

Form vor, wobei im Rahmen des Projekts der stündliche Zeitraum relevant ist, da sich die weiteren Zeithorizonte hieraus ableiten lassen und eine feingranulare Sicht spezifischere Auswertungen zulässt.

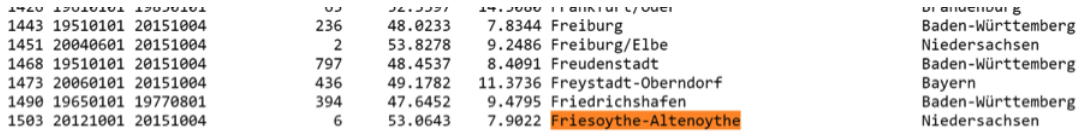


Index von /pub/CDC/observations_germany/climate/hourly/

Name	Größe	Änderungsdatum
[übergeordnetes Verzeichnis]		
air_temperature/		05.06.14, 00:00:00
cloudiness/		25.09.14, 00:00:00
precipitation/		13.11.14, 00:00:00
pressure/		13.11.14, 00:00:00
soil_temperature/		05.06.14, 00:00:00
solar/		27.05.15, 02:46:00
sun/		05.06.14, 00:00:00
wind/		17.04.15, 12:56:00

Abbildung 23: Ordner zu angebotenen Wetteraspekten des DWD

Der Ordner für stündliche Auswertungen beinhaltet die verschiedenen erhobenen Klimaarten. Zu finden sind die Daten zu Lufttemperatur, Bodentemperatur, Bewölkung, Niederschlag, Luftdruck, Solar, Sonne und Wind. Die Unterschiedlichen Klimaaspekte sind durch Buchstabenkürzel abgegrenzt, die als Identifier dienen und in der Datenvorbereitung aufgegriffen werden. Da für jede in Deutschland vorhandene Wetterstation eine separate Datei mit den entsprechenden Datensätzen vorliegt, die anhand einer Stations-ID differenziert wird, muss diese zunächst ausfindig gemacht werden. Im Ordner zum betreffenden Klimaaspekt ist eine CSV Datei enthalten, die sämtliche Wetterstationen mit Informationen zu ID-Nummer, Erhebungszeitraum, Koordinaten, Ortsnamen und Bundesland auflistet. Für Oldenburg ist ein Eintrag mit der ID-Nummer 3791 vorhanden, allerdings zeigt der Erhebungszeitraum an, dass die Wetterdaten zwar ab dem 01.01.1991 erhoben wurden, jedoch ab dem 01.10.2012 keine Erhebung mehr stattfindet. Um die nächstgelegene Wetterstation ausfindig zu machen wird diese telefonisch beim DWD erfragt. Der Mitarbeiter bestätigt, dass die Wetterstation Oldenburg nicht aktiv ist und stattdessen seit dem 01.10.2012 die Wetterstation Friesoythe-Altenoythe für die Erhebung genutzt wird. Über die Auflistung der Stationen wird die Stations-ID erhoben, diese lautet 1503.



1443	19510101	20151004	236	48.0233	7.8344	Freiburg	Baden-Württemberg
1451	20040601	20151004	2	53.8278	9.2486	Freiburg/Elbe	Niedersachsen
1468	19510101	20151004	797	48.4537	8.4091	Freudenstadt	Baden-Württemberg
1473	20060101	20151004	436	49.1782	11.3736	Freystadt-Oberndorf	Bayern
1490	19650101	19770801	394	47.6452	9.4795	Friedrichshafen	Baden-Württemberg
1503	20121001	20151004	6	53.0643	7.9022	Friesoythe-Altenoythe	Niedersachsen

Abbildung 24: Auflistung der Wetterstationen

Anhand dieser Stations-ID können die Datensätze zu den jeweiligen Wetterdaten abge-

rufen werden. Diese Datensätze liegen auf dem FTP-Server in Form einer ZIP-Datei vor. Nachdem diese Datei heruntergeladen und entpackt worden ist kann auf die CSV Datei mit den Wetterdaten zugegriffen werden. Die Datei enthält sämtliche vorhandene Werte in aufsteigender chronologischer Reihenfolge. Die Werte des letzten vorhandenen Tages liegen in den unteren 24 Zeilen vor. Als Trennzeichen für Spalten wird ein Semikolon verwendet, der Datensatz für die Lufttemperatur enthält beispielsweise die Daten zur Lufttemperatur und der relativen Luftfeuchtigkeit. Darüber hinaus ist der angegebene Zeitstempel relevant, welcher das Jahr, den Monat, den Tag und die Stunde der Erhebung beinhaltet und daher als eindeutiger Bezeichner genutzt werden kann.

6.10 Nahverkehr

Der Nahverkehr als wichtiger Verkehrsbestandteil der Stadt Oldenburg soll als weiterer Parameter Berücksichtigung finden. Der Aspekt der Echtzeit kann an dieser Stelle nur simuliert werden, da von der Stadt Oldenburg keine Echtzeit Daten zu eingesetzten Bussen zur Verfügung gestellt werden. Die Simulation nutzt daher den vorhandenen Busfahrplan legt die Annahme zugrunde, dass im Nahverkehr keine Verspätungen auftreten. Für die Datenerhebung liegen verschiedene Ansätze vor. Ein Ansatz nutzt die OSM-Daten zur Erhebung der Linienpläne, der zweite Ansatz verwendet die Linienpläne der VWG als Grundlage. Zwar enthalten die OSM-Daten Attribute, die die Knoten als Bushaltestelle identifizieren und eine Zuordnung zu bestehenden Relationen ermöglichen, jedoch ist eine Sortierung der Haltestellen aufgrund der fehlenden Positionsangabe nicht möglich. Auch die Zuordnung der Haltestellen zu Abfahrtszeiten ist nicht durchführbar, da die korrekten Bezeichnungen der Haltestellen fehlen. Für die Erhebung Linienpläne wird daher auf die angebotenen Datensätze der VWG zurückgegriffen. Die VWG betreibt die Busse im Stadtverkehr und stellt auf ihrer Internetseite den Liniennetzplan, sowie Linienpläne für die einzelnen Buslinien zur Verfügung. Sämtliche Pläne liegen im PDF-Dateiformat vor und sind daher nicht unmittelbar elektronisch zu erfassen. Da eine Änderung der Buslinien nicht ohne weiteres vorgenommen wird und die Erfassung sämtlicher Daten einen einmaligen Aufwand darstellt werden diese manuell erhoben. Zunächst erfolgt der Download sämtlicher Linienpläne von der Internetseite der VWG. Der Linienplan enthält in geordneter Reihenfolge die Auflistung der Haltestellen sowie die Abfahrtszeiten. Beachtet werden muss, dass es bei bestimmten Buslinien Abwandlungen im Fahrplan in Abhängigkeit von der Uhrzeit gibt, außerdem ist die Position der Haltestellen abhängig von der Fahrtrichtung. In einer Excel-Liste werden zunächst die Linienpläne erfasst, hierzu werden der Haltestellenname, die Liniennummer und die Position der Haltestelle im Linienplan erhoben. Zur Berücksichtigung der differierenden Haltestellenabfolge zu bestimmten Uhrzeiten und Richtungen wird die Liniennummer um einen Buchstaben als eindeutiger Bezeichner ergänzt. Zur Verortung der Haltestellendaten werden diese mit den OSM Daten verknüpft. Die OSM Knoten-Objekte enthalten den Tag public_transport, welcher bei Haltestellen den Wert stop_position besitzt. Zu den Buslinien liegen zudem Relationen in OSM vor, die für den Tag ref die Angabe der Buslinien-Nummer beinhalten. Den Buslinien-Relationen sind sämtliche Knoten der Bushaltestellen zugeordnet. Über die Such-Funktion auf der OSM Plattform kann die Relation der Buslinie abgerufen werden, ein Mausklick

auf die einzelnen Knoten ermöglicht die Anzeige der Knoten-ID. Die nach der Erhebung vorliegende Excel-Tabelle enthält demnach Angaben zu Buslinie, Haltestellenname, Haltestellenposition und Knoten-ID. Die fehlenden Angaben zu Längen- und Breitengrad kann über die Knoten-ID aus den in der Datenbank vorliegenden OSM-Daten abgerufen werden.

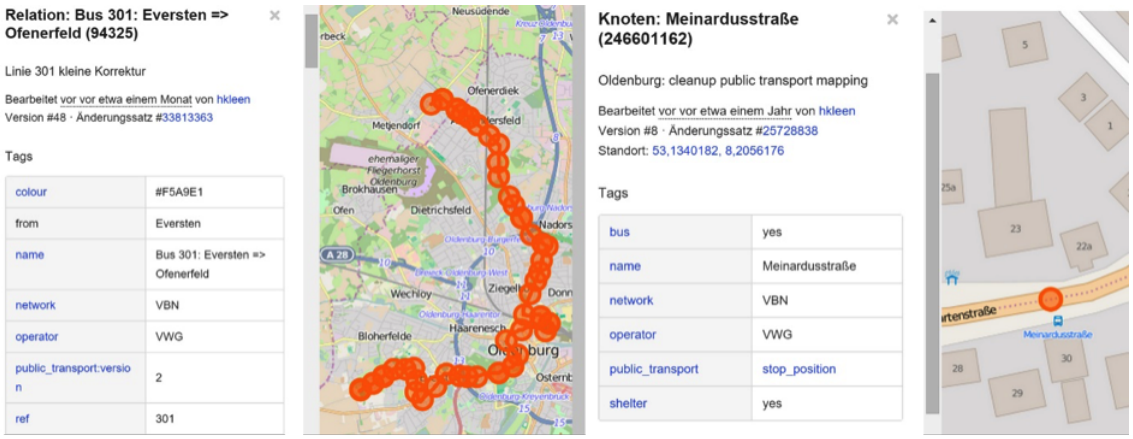


Abbildung 25: OSM Relation zu Buslinie 301 und Knoten zu Haltestelle Meinardusstraße

Die Daten zu Abfahrtzeiten sind ebenfalls über die Linienpläne der VWG einsehbar. Es werden dabei drei Zeiträume unterschieden, Montag-Freitag, Samstag sowie Sonn- und Feiertag. Die hier enthaltenen Angaben werden ebenfalls über eine manuelle Erhebung in einer Excel-Tabelle vorgenommen. Neben der Abfahrtszeit wird der Zeitraum als Gruppe (1,2 oder 3) sowie die Haltestelle und die Linie inklusive Buchstaben zur eindeutigen Zuordnung zum Fahrplan erhoben.

6.11 Events

Die Eventdaten für Oldenburg erlangten die Studierenden über diverse Webseiten im Internet. Dazu zählen insbesondere die Wochenmärkte in verschiedenen Regionen Oldenburgs sowie Verkaufsoffene Sonntage, lange Shoppingnächte und weitere Events, die den Verkehr in Oldenburg beeinflussen. Des Weiteren wurden spezielle Veranstaltungen wie Großveranstaltungen, Konzerte, Ferien etc. in dem Zeitraum vom 01.03.2015 - 31.03.2015 gefiltert und in einer Tabelle zusammen getragen. Dies bedürfte gründlicher Recherchearbeit. nach Beendigung der Recherche wurden die Veranstaltung mit Datum/ Uhrzeit, Adresse, Knotenpunkt auf der OpenStreetMap und der voraussichtlichen Besucherzahl in einer CSV Datei zusammengetragen, die anschließend in die SAP HANA importiert wurde. Schließlich können Zusammenhänge zwischen den vorliegenden Events und dem Verkehrsverhalten in Oldenburg erkannt und Rückschlüsse für die Verbesserung des Verkehrsflusses gezogen werden.

6.12 Kommunikation mit weiteren Unternehmen

Es wurden aufgrund der Unsicherheit bezüglich der zu erhaltenden Daten diverse Unternehmen deutschlandweit angeschrieben, um Informationen über Datenstrukturen und weitere Beschaffenheit sowie Möglichkeiten zu erhalten, um für Oldenburg gegebenenfalls realitätsgetreue Simulationsdaten erzeugen zu können. Auf der MDM-Plattform sind diverse Anbieter dargestellt, die unterschiedliche Daten zur Verfügung stellen. Anschließend wurden Kontaktdaten der in Frage kommenden Unternehmen extrahiert (siehe nachfolgende Tabelle). Hiernach wurde ein Schreiben durch die Projektgruppe erstellt und an die Kontaktliste gesendet. Durch die angeschriebenen Unternehmen ergaben sich unterschiedliche Antworten (siehe Kontaktkurzprotokoll). Die Projektgruppe trat schließlich mit der Verkehrsleitzentrale der Stadt Oldenburg in Kontakt. Hiernach hat die Projektegruppe entschieden, den Kontakt zu den genannten Unternehmen einzustellen und die zur Verfügung gestellt Daten durch die Unternehmen nicht zu nutzen. Dies liegt daran, dass sich die Projektgruppe zu dem Schritt entschieden hat, sich lediglich auf Daten der Stadt Oldenburg zu stützen.

Name	Datengeber	Datenart	Bezeichnung	Antw.	Ergebnis
Elektromobilität Parkdaten für Ladestationen statisch	regio iT aachen gmbh	Parkdaten	Germany	Nein	siehe Kon.Pro Nr. 1
Verkehrsmeldungen	Hessen Mobil - Straßen- und Verkehrsmanagement	Verkehrsinformationen, Verkehrsdaten	BAB in Hessen	Ja	siehe Kon.Pro Nr. 2
Baustellen Schleswig-Holstein	Landesbetrieb Straßenbau und Verkehr Schleswig-Holstein	Verkehrsinformationen	Schleswig-Holstein	Ja	siehe Kon.Pro Nr. 3
Verkehrsmeldungen	Stadt Kassel	Verkehrsinformationen, Parkdaten	Stadt Kassel	Ja	siehe Kon.Pro Nr. 4
Tagesbaustellen Toll-Collect	B.A.S. Verkehrstechnik AG	Verkehrsinformationen	Kontrollbrücken der Toll-Collect	Ja	siehe Kon.Pro Nr. 5
Qualitätssicherung der Schaltzeitprognose	BMW	Verkehrsdaten	Deutschland	Ja	siehe Kon.Pro Nr. 6
ADAC Ganglinien zu Verkehrsflusszuständen	ADAC e.V.	Verkehrsdaten	Deutschland	Ja	siehe Kon.Pro Nr. 7
Statische Parkdaten Stadt Magdeburg	ifak - Institut für Automation und Kommunikation e.V. Magdeburg	Parkdaten	Stadt Magdeburg	Ja	siehe Kon.Pro Nr. 8
Meldungsmanagement - Informationen über Arbeitsstellen im Saarland	Landesbetrieb für Straßenbau Saarland (LfS)	Verkehrsinformationen	Saarland	Ja	siehe Kon.Pro Nr. 9
Verkehrsmeldungen Stadt Düsseldorf	Landeshauptstadt Düsseldorf	Verkehrsinformationen, Umfeld-Daten, Verkehrslage	Stadt Düsseldorf	Nein	siehe Kon.Pro Nr. 10
Bürgerbüro-Nord Stadt Oldenburg	Bürgerbüro-Nord	Kraftfahrzeugbestand	Oldenburg	Nein	siehe Kon.Pro Nr. 11
VInfoTest	Heusch/Boesefeldt GmbH	Verkehrsinformationen	Hessen	Ja	siehe Kon.Pro Nr. 12

Im folgenden Kontaktkurzprotokoll werden die Antworten der angeschriebenen Unternehmen aufgelistet.

- **Kontaktkurzprotokoll Nr. 1**: Es gab keine Rückmeldung
- **Kontaktkurzprotokoll Nr. 2**: Die zuständige Mitarbeiterin teilte uns mit, die Anfrage direkt an einen weiteren Mitarbeiter, der für unsere Anfrage zuständig ist, zu richten. Die Anfrage wurde an den genannten Mitarbeiter weitergleitet. Auf die Anfrage wurde jedoch nicht geantwortet.
- **Kontaktkurzprotokoll Nr. 3**: Es wurde geschrieben, dass dem Unternehmen Verkehrsdaten der Verkehrszählungen, die alle fünf Jahre stattfinden, vorliegen. Darüber hinaus standen Daten der stetig durchgeführten Dauerzählungen zur Verfügung. Am 10.04.2015 wurde telefonischer Kontakt mit der zuständigen Mitarbeiterin aufgenommen. Sie definierte die Daten etwas konkreter und nannte Details, die in den Daten enthalten seien. Hierzu zählt die Art und MEnger der Fahrzeuge sowie der Füßgänger, die spezifische Ampeln überqueren. Diese Daten können nicht zur Verfügung gestellt werden, da diese mehrere Terra Bytes groß sind. Der Projektgruppe wurden anschließend Zählstellenkarten sowie Ërgebnisse der automatischen Dauerzählung 2014ïm PDF-Format zur Verfügung gestellt. Jedoch konnte kein sinnhafter Zusammenhang mit den bereits importierten Daten der Verkehrsleitzentrale hergestellt werden, sodass die Daten verworfen werden mussten.
- **Kontaktkurzprotokoll Nr. 4**: Der angeschriebene Mitarbeiter teilte uns mit, dass sie derzeit unterschiedliche Verkehrsdaten über die MDM-Plattform anbieten. Er bat uns um ein Telefongespräch, um offene Fragen zu klären. Am 27.03.2015 kam der Kontakt mit dem Unternehmen zustande und es wurden Park- und Baustellendaten über die MDM-Plattform angeboten. Um jedoch Zugang zu besagten Daten zu erhalten wurde eine sehr komplexe Schnittstelle benötigt, die den Rahmen der Projektgruppe gesprengt hätte. Aus diesem Grund, musste auf die Einbindung der Daten seitens der Projektgruppe verzichtet werden.
- **Kontaktkurzprotokoll Nr. 5**: Die B.A.S. Verkehrstechnik AG kann uns nicht weiter helfen, da es sich hierbei um private Kundendaten handelt.
- **Kontaktkurzprotokoll Nr. 6**: Am 31.03.2015 wurde telefonischer Kontakt mit dem zuständigen Mitarbeiter aufgenommen. Er teilte mit, dass uns die Daten nicht weitergegeben werden können, da diese privatbezogen sind.
- **Kontaktkurzprotokoll Nr. 7**: Am 01.04.2015 wurde telefonischer Kontakt zu dem zuständigen Mitarbeiter aufgenommen. Nach Absprache mit dem Mitarbeiter wird uns eine Emailkundenversorgung mit Verkehrsmeldungen und Baustellenmeldungen für den Großraum Oldenburg bereitgestellt.
- **Kontaktkurzprotokoll Nr. 8**: Der zuständige Mitarbeiter teilte uns mit, dass wir uns zunächst an die Stadt Magdeburg wenden müssen um eine entsprechende Freigabe zu erhalten. Anschließend können Daten freigegeben werden.

- **Kontaktkurzprotokoll Nr. 9**: In einer E-Mail wurde seitens eines Mitarbeiter die Beireitschaft zur Zusammenarbeit signalisiert und die Herausgabe von Baustellen- und Verkehrsdaten (Daten zur Verkehrsbelastung, insbesondere auf den Bundesautobahnen im Saarland) signalisiert. Nach einem Telefonat, sollte die Anfrage geprüft und anschließend eine Herzausgabe durchgeführt werden. Nach mehrfachem Nachfragen gab es jedoch keine Rückmeldung mehr.
- **Kontaktkurzprotokoll Nr. 10**: Es gab keine Rückmeldung.
- **Kontaktkurzprotokoll Nr. 11**: Es wurde zunächst telefonischer Kontakt mit der Stadt Oldenburg aufgenommen und der Sachverhalt sowie das Vorhaben erläutert. Auf Hinweis des Mitarbeiters wurde die Anfrage per E-Mail direkt an das Bürgerbüro-Nord der Stadt Oldenburg gesendet. Auf die Anfrage per Mail gab es keine Rückmeldung.
- **Kontaktkurzprotokoll Nr. 12**: In der Rückmeldung wurde mittgeteilt, dass die Daten ohne die Genehmigung der entsprechenden Landesämter und Städte nicht zur Verfügung gestellt werden können.

7 Datenvorbereitung

In Kapitel **6** wurde beschrieben, was allgemein für das Datenverständnis zu beachten ist und am Ende des Prozessschrittes stehen alle Daten und deren Beschreibungen zur Verfügung. Außerdem wurden bereits aufgrund der konkreten Fragestellung oder erster Einblicke Attribute entfernt, die keine vielversprechenden Ergebnisse liefern. Aus diesen vorhandenen Daten ist nun im allgemein das Ziel von Data Mining-Prozessen neues Wissen in Form von beispielsweise neuen Zusammenhängen zu generieren. Egal welches konkrete Verfahren dabei angewandt wird, das neu erzeugte Wissen basiert auf den vorhandenen Daten. Doch damit das neue Wissen aussagekräftig ist, müssen die erhobenen Daten nicht nur quantitativ in entsprechendem Maße vorhanden sein, sondern auch qualitativ. Beispielsweise könnten bei der Aufnahme der Daten Attribute mit aufgenommen worden sein, wie das Alter einer Wohnung in einer Immobilienkartei, die zu dem Zeitpunkt als wichtig angesehen wurden. Dadurch würde der Umfang der Daten zwar erweitert werden, allerdings wäre ein solches Attribut bei der konkreten Fragestellung nach einem Zusammenhang zwischen der Wohnfläche und der Adresse von Wohnungen unter Umständen nicht sehr aussagekräftig. Da aber das Laufzeitverhalten von den meisten Data Mining Verfahren direkt von der Anzahl der Attribute und der Anzahl der Beispiele abhängt, sollten die Daten keine unnötigen Werte enthalten. Neben der Auswahl der wichtigen Attribute ist auch deren Formatierung wichtig. Wenn das geplante Analyseverfahren nicht mit symbolischen Werten wie *Herr* oder *Frau* bei der Anrede umgehen kann, müssen die Werte beispielsweise in die numerischen Werte 1 oder 2 überführt werden. Ebenso verhält es sich, wenn die Standardisierung von Werten auf einen bestimmten numerischen Bereich erwartet wird. In diesem Prozessschritt, der *Datenvorbereitung* geht es nun darum die beschafften Daten weiter vorzubereiten, um sie in ein Analyseverfahren einspeisen zu können. Die *Datenvorbereitung* teilt sich dabei in fünf Aufgabenbereiche auf, in welchen unterschiedliche Aspekte behandelt werden. Der erste Aufgabenbereich ist die Datenauswahl, welche in Abschnitt 7.1 beschrieben wird. Dabei geht es darum, die bereits getroffene Auswahl der Daten zu überprüfen und unter Gesichtspunkten wie möglicher Analyseverfahren neu zu evaluieren. In Abschnitt 7.2 wird darauf eingegangen, wie Fehler innerhalb der ausgewählten Daten behandelt werden können. Fehler können dabei zum einen fehlende Werte und zum anderen verrauschte Daten sein. Der dritte Aufgabenbereich, die Datenkonstruktion, wird in Abschnitt 7.3 behandelt. Das Ziel dieses Arbeitsschrittes ist es vorhandene Daten zu kombinieren um neue Daten zu erschaffen, die aussagekräftiger oder für die Analyse eher verwendbar sind. Außerdem sollen einzelne Attribute in ein Format gebracht werden, welches die Analyse verlangt. In Abschnitt 7.4 wird die Datenintegration behandelt, bei welcher es darum geht mehrere Datensätze, die zu dem gleichen Objekt vorhanden sind, zusammenzuführen. Der letzte Schritt, die Datentransformation wird in Abschnitt 7.5 behandelt und liefert am Ende Tabellen, die in das gewählte Analyseverfahren eingespeist werden können. Die darauf folgenden Abschnitte beschäftigen sich mit der konkreten Datenvorbereitung der ADAC-Daten (vgl. Abschnitt 7.8), der OSM-Daten (vgl. Abschnitt 7.9) und der Zählspuldaten von der Stadt Oldenburg (vgl. Abschnitt 7.10).

7.1 Datenauswahl

Am Ende des Prozessschrittes *Datenverständnis*, auf welches in Abschnitt 6 eingegangen wurde, stehen alle Daten und deren Beschreibungen zur Verfügung. Außerdem wurden bereits aufgrund der konkreten Fragestellung oder erster Einblicke Attribute entfernt, die keine vielversprechenden Ergebnisse liefern. In diesem Arbeitsschritt gilt es die Auswahl der Daten weiter zu präzisieren, wodurch die Datenbasis eventuell weiter verkleinert werden kann. Und da die Laufzeit der meisten Analyseverfahren direkt von der Anzahl der Attribute und Beispiele abhängt, führt eine Verkleinerung der Datenbasis zu einer verbesserten Laufzeit der Analyseverfahren. Allerdings ist stets darauf zu achten, dass keine oder mindestens so wenig wie möglich Informationen verloren gehen. Eine Möglichkeit um die Datenbasis zu reduzieren ist es eine Dimensionsreduktion durchzuführen. Dabei werden einzelne Attribute entfernt, die entweder irrelevant oder redundant sind. Beispielsweise könnten in einer Datenbank sowohl das Alter, als auch das Geburtsdatum angegeben sein. Da aus dem Geburtsdatum das Alter errechnet werden kann, sind die beiden Attribute redundant, beziehungsweise die Angabe über das Geburtsdatum ist sogar noch präziser, als lediglich das Alter. In einem solchen Fall könnte das komplette Attribut „Alter“ entfernt werden ohne Informationen zu verlieren. Eine andere Möglichkeit um die Datenbasis für die Analyseverfahren zu reduzieren ist das Daten Sampling. Die Idee dabei ist es in dem späteren Analyseverfahren nicht den vollständigen Datensatz zu verwenden, sondern lediglich eine ausgewählte Anzahl an Beispielen. Für einige Analyseverfahren ist es sogar notwendig, die Datenbasis in mindestens zwei Sets aufzuteilen. Beispielsweise benötigen neuronale Netze ein Trainings-Set und ein Test-Set, welche disjunkt sein müssen. Das Trainings-Set wird dazu verwendet, das Netz hinsichtlich einer bestimmten Problemstellung zu trainieren. Das Test-Set wird nach dem Training dazu verwendet das Gelernte zu überprüfen und die Güte des Netzes zu bestimmen. Die Auswahl der Beispiele kann dabei komplett zufällig vorgenommen werden oder aber auch nach bestimmten Kriterien. Beispielsweise können die Daten anhand des Wertes eines bestimmten Attributes in Gruppen unterteilt werden. Einzelne Gruppen werden dann zufällig ausgewählt und entsprechend alle Beispiele dieser Gruppe verwendet. Bei einer Unterteilung nach Kriterien, ist allerdings darauf zu achten, dass die Auswahl am Ende immer noch repräsentativ ist, wenn das Ergebnis der Untersuchung auf alle Daten zutreffen soll. Da bei den beiden vorgestellten Verfahren möglicherweise Informationen durch das entfernen von Attributen oder das Betrachten von Sample-Sets statt der ganzen Datenbasis verloren gehen, ist stets Protokoll darüber zu führen, welche Daten aus welchem Grund nicht betrachtet wurden. Außerdem sollten die Originaldaten irgendwo unverändert gespeichert werden um sie zu einem späteren Zeitpunkt eventuell unter anderen Gesichtspunkten neu auswerten zu können. Fällt in diesem, oder einem der anderen Arbeitsschritte auf, dass Daten fehlen, ist es immer möglich in den vorherigen Prozessschritt zu wechseln und neue Daten, unter Beachtung der Erfahrungen aus diesem Prozessschritt hinsichtlich der Daten Qualität, dem Ergebnis der Untersuchungen der Daten oder dem späteren Analysemodel, zu erheben.

7.2 Datenbereinigung

Die Datenbereinigung stellt einen weiteren Arbeitsschritt innerhalb der Datenvorbereitung dar. Dabei geht es darum Fehler in den Daten zu erkennen und entsprechend zu behandeln. Die Fehler die dabei auftreten, können in zwei Kategorien aufgeteilt werden: fehlende und verrauschte Werte. Diese entstehen, weil die Daten nicht sorgfältig genug erfasst wurden, aufgrund von Missverständnissen innerhalb der Kommunikation oder aus anderen Gründen. Allerdings können durch solche Fehler Probleme bei der Analyse entstehen, weil das ausgewählte Analyseverfahren nicht mit fehlerhaften Werten umgehen kann oder weil Attribute bei denen überwiegend Werte fehlen nutzlos sind.

7.2.1 Fehlende Werte

Die erste Art von Fehlern die hier behandelt werden soll, sind fehlende Werte. Eine Möglichkeit um fehlende Werte zu behandeln ist es, diese durch andere Daten, z.B. aus anderen Systemen, zu ersetzen. Ein Beispiel dafür ist eine Kundendatei in welcher für einen Kunden die Adresse fehlt. Diese kann unter Umständen einem öffentlichen Telefonbuch entnommen und dadurch vervollständigt werden. Wenn solche zusätzlichen Daten vorliegen ist es das Beste diese Möglichkeit anzuwenden, da die zu ergänzenden Daten dann nicht abgeleitet werden müssen und als korrekt angenommen werden können. Liegen zusätzliche Daten allerdings nicht vor, gibt es dennoch Möglichkeiten die fehlenden Werte zu ergänzen. Allerdings muss dabei immer berücksichtigt werden, dass die neuen Werte dann nur angenommen werden und nicht korrekt sein müssen. Eine der einfachsten Möglichkeiten ist es, die Tupel bei welchen ein Wert fehlt komplett zu löschen und nicht weiter zu betrachten. Dies ist allerdings nur dann sinnvoll, wenn nur wenige Tupel davon betroffen sind. Ansonsten kann dieses Vorgehen schnell dazu führen, dass der komplette Datensatz unbrauchbar ist, weil kaum noch Daten vorhanden sind. Andersrum verhält es sich, wenn bei vielen Tupeln für gleiche Attribute die Werte fehlen. In diesem Fall wäre es eine Möglichkeit das komplette Attribut zu löschen. In beiden Fällen sollte, wie auch schon bei der Datenauswahl, darauf geachtet werden, dass die Originaldaten irgendwo gespeichert werden, falls zu einem späteren Zeitpunkt neue Daten zur Verfügung stehen. Wenn das Löschen von Tupeln oder Attributen allerdings keine Option ist, weil der Datensatz dadurch zu stark beeinflusst würde ist eine weitere relativ einfache Möglichkeit die fehlenden Werte durch Default-Werte zu ersetzen. Die einfachste, allerdings auch die ungenauste Möglichkeit dabei ist es für jedes Attribut einen einzigen Default-Wert zu definieren und diesen einzufügen. Ein Nachteil davon ist, dass dieser Default-Wert unabhängig von den bekannten Daten ist und beispielsweise den Durchschnittswert eines Attributes verfälschen kann. Um diesem Problem etwas entgegenzusteuern kann als Default-Wert auch der Durchschnitt der vorhandenen Werte verwendet werden. Dadurch ist eine gewisse Abhängigkeit gegeben und der Durchschnitt aller Werte wird nicht mehr verändert. Eine weitere Möglichkeit ist es, vor allem wenn die einzelnen Tupel bereits vorher in Klassen unterteilt wurden, als Default-Wert den Durchschnitt der zugeordneten Klasse zu verwenden. Diese drei Möglichkeiten stellen nur Beispiele dar, wie ein Default-Wert verwendet werden kann. Im Allgemeinen kommt es bei der Auswahl des Default-Wertes

immer darauf an, was später mit den Daten gemacht werden soll und welchen Einfluss die Auswahl eines Default-Wertes auf das mögliche Ergebnis hat. Eine Alternative zu den Default-Werten stellt die Vorhersage von wahrscheinlichen Werten dar, wobei es wiederum viele verschiedene Möglichkeiten für eine Vorhersage gibt. Zur Verdeutlichung sollen die Verkaufszahlen für einzelne Monate dienen, bei welchen die Werte für einige Monate fehlen. Für eine Vorhersage könnten dann nur der Monat vor und nach dem Fehlenden genutzt und ein linearer Zusammenhang angenommen werden. Der Monat würde dann mit dem Mittelwert der beiden einschließenden Monate beschrieben werden. Diese Möglichkeit ist relativ einfach, berücksichtigt allerdings nicht die Historie vor dem fehlenden Monat. Wenn die letzten x Monate vor dem fehlenden Wert beispielsweise einen exponentiellen Anstieg vermuten lassen, dann kann der fehlende Wert auch durch eine andere, als eine lineare Funktion beschrieben werden. Es zeigt sich also, dass eine Vorhersage nicht immer einfach durchzuführen ist und unter Umständen selber ein gewisses Maß an Rechenaufwand und einem Verständnis der vorhandenen Daten beansprucht. Auch hier kommt es wieder darauf an, was mit den Daten später gemacht werden soll, um zu entscheiden ob der zusätzliche Rechenaufwand gerechtfertigt ist oder nicht. Es gibt aber auch Fälle, bei denen die bis hierher aufgeführten Verfahren nicht anwendbar sind. Beispielsweise ist es wenig sinnvoll bei der Anrede den Default-Wert *Herr* einzuführen, da die Wahrscheinlichkeit, dass diese dann falsch ist bei 50% liegt. Eine Vorhersage könnte sich auch als schwierig gestalten, da die Anrede unabhängig von seiner Umgebung ist. Wenn die eingrenzenden Einträge jeweils *Herr* sind, muss das nicht zwingend bedeuten, dass der fehlende Eintrag auch *Herr* ist. Selbige Problematik tritt auf, wenn die eingrenzenden Einträge beide *Frau* oder gemischt sind. Im Fall der Anrede lässt sich allerdings die Anrede möglicherweise aus dem Vornamen ableiten. Generell gesprochen ist es eine weitere Möglichkeit fehlende Werte aus anderen, vorhandenen Werten abzuleiten, wobei auch dieses Vorgehen ein gewisses Datenverständnis voraussetzt. Wenn alle bis hierher beschriebenen Verfahren nicht anwendbar sind, bleibt noch die Möglichkeit den möglichen Wertebereich um einen Wert *unbekannt* zu erweitern. Dadurch bekommt das Nicht-Wissen eine Semantik und kann ausgewertet werden. In einigen Fällen kann es sogar besser sein einen unbekannten Wert als solchen zu deklarieren, als ihn durch einen angenommenen Wert zu ersetzen.

7.2.2 Verrauschte Daten

Die zweite Art von Fehlern sind verrauschte Daten, welche entstehen können, wenn beispielsweise die Datenerfassung falsch durchgeführt wird oder es Fehler bei der Datenübertragung gibt. Die erste Schwierigkeit die behoben werden muss um Fehler dieser Art zu behandeln ist, sie zunächst zu Identifizieren. Da die Daten im Gegensatz zu fehlenden Daten vorhanden sind, kann die Datenbasis nicht nach NULL-Werten durchsucht werden. Eine erste Möglichkeit um verrauschte Daten, bzw. Ausreißer zu identifizieren ist, die Überprüfung expliziter semantischer Bedingungen. Das bedeutet die Daten werden darauf untersucht, ob ihr Wert zunächst überhaupt möglich sein kann. Wenn der Wertebereich für ein Attribut beispielsweise im Bereich [0..1] definiert ist, der eingetragene Wert allerdings 2 ist, kann dieser Wert nicht stimmen. Eine andere Möglichkeit ist es die vorhandenen Werte in Gruppen einzuteilen (engl. clustering). Beim *Dichtebasierten Clustern* beispiels-

weise wird überprüft, ob es für einen gegebenen Wert einen anderen Wert innerhalb eines bestimmten Abstandes gibt, wobei der Abstand für ein Attribut die Differenz der beiden Werte sein kann. Wenn dies der Fall ist werden die beiden Werte *Nachbarn* genannt und zu einer Gruppe (engl. Cluster) zusammengefasst. Gibt es jetzt noch weitere Punkte, die zu einem beliebigen Punkt aus dem Cluster ebenfalls eine Nachbarschaftsbeziehung aufweisen, werden auch diese mit in das Cluster aufgenommen. Dieser Vorgang führt dazu, dass am Ende alle Werte, die ähnlich zueinander sind, einem Cluster zugeordnet wurden. Die Größe der Cluster gibt dann unter anderem Aufschluss über die Verteilung der Werte. Große Cluster stellen dabei den „Normalfall“ dar, wogegen Werte, die keinem oder nur sehr kleinen Clustern zugeordnet werden können, Ausreißer darstellen. Die Schwierigkeit bei diesem Verfahren liegt darin, das Abstandsmaß richtig zu bestimmen. Wenn Werte nur dann als Nachbarn gelten, wenn die Differenz sehr klein ist, kann es dazu kommen, dass es insgesamt nur sehr kleine, dafür aber sehr viele, Cluster gibt. Wird das Abstandmaß zu groß gewählt, fallen möglicherweise alle Werte in das selbe Cluster. In beiden Fällen ist es dann sehr schwierig bis unmöglich Ausreißer zu identifizieren. Wenn die Ausreißer identifiziert wurden, gilt es diese gesondert zu behandeln. Die Möglichkeiten sind dabei sehr ähnlich zu denen, die bereits zur Behandlung von fehlenden Daten erläutert wurden. Die verrauschten Daten können also beispielsweise durch Werte aus alternativen Datenquellen, Default-Werte oder vorhergesagten Werte ersetzt werden. Sollten die Ausreißer über ein Clusterverfahren identifiziert worden sein, bietet sich hier auch noch die Möglichkeit den verrauschten Wert durch einen aus dem nächstliegenden Cluster zu ersetzen. Allerdings stellen Ausreißer nur eine extreme Form des Rauschens dar. Um die restlichen Werte, die möglicherweise auch verrauscht sind, zu behandeln gibt es Methoden um diese zu *glätten*. Solche Methoden finden hauptsächlich bei numerischen Werten Anwendung. Eines dieser Glättungsverfahren ist das sogenannte *Binning*. Dabei wird der gesamte Wertebereich in einzelnen Intervalle (engl. Bins) unterteilt. Danach wird jeder Wert innerhalb eines Intervalls entweder durch die Invallgrenze ersetzt, zu welcher die Differenz am geringsten ist, oder aber alle Werte werden durch den Mittelwert dieses Intervalls ersetzt. Alternativ zum Binning, bei welchem der gesamte Wertebereich in numerische Intervalle eingeteilt wurde ist es, den Wertebereich in symbolische Werte zu unterteilen. Beispielsweise können die einzelnen Temperaturen einer Heizungssteuerung in die Klassen *warm*, *mittel* und *kalt* eingeteilt werden. Je nachdem in welcher Form die Daten später analysiert und repräsentiert werden sollen, kann eine Darstellung durch symbolische Werte für einen Anwender nachvollziehbarer sein. Ein Nachteil ist allerdings, dass symbolische Werte subjektiv sind und eventuell Informationen verloren gehen. Deswegen muss stets überlegt werden, ob eine Klassifizierung für die angestrebte Genauigkeit der Auswertung genügt, oder nicht. Eine dritte Möglichkeit um Daten zu glätten, ist durch verschiedene Regressionsverfahren gegeben. Auf diese wird in Kapitel 8 unter Abschnitt 8.1.2 eingegangen.

7.3 Datenkonstruktion

Bei der Datenkonstruktion als dritten Aufgabenbereich innerhalb der Datenvorbereitung geht es auf der einen Seite darum aus den vorhandenen Daten neue Werte zu konstruieren, die für die Analyse benötigt werden und auf der anderen Seite die benötigten vorhande-

nen Daten in ein Format zu bringen, welches für die Analyse benötigt wird. Zunächst soll darauf eingegangen werden neue Daten aus bereits vorhandenen zu konstruieren. Wenn in einer Datenbank für Häuser beispielsweise nur die Längen der Wände angegeben sind, ist das für potentielle Kunden wenig aussagekräftig. Allerdings lässt sich aus diesen Werten sehr leicht die Grundfläche berechnen, die wiederum mehr Aussagekraft gegenüber einem potentiellen Kunden hat. Allgemein wird dieses Verfahren, bei welchem mehrere Einzelobjekte logisch zu einem Gesamtobjekt zusammengefasst werden, Aggregation genannt. In dem Beispiel wurden zwei Attribute zu einem einzigen neuen Attribut aggregiert. Neben einzelnen Attributen können auch Entitäten aggregiert werden. Beispielsweise können die Einzelobjekte *Person* und *Hote* zusammen in die Entität *Reservierung* aggregiert werden. Bei dem Zusammenführen von Entitäten ist weiterhin zu unterscheiden zwischen Aggregation und Komposition. Wie in dem Beispiel können die Einzelobjekte bei einer Aggregation weiter bestehen, auch wenn das Gesamtobjekt aufgelöst wurde. Ein Beispiel für die Komposition, bei welcher die Einzelobjekte nicht weiterbestehen können, wenn das Gesamtobjekt aufgelöst wurde, stellt ein Gebäude dar, welches aus den einzelnen Stockwerken aggregiert wurde. Es ist also möglich sowohl Attribute als auch Entitäten zu aggregieren, wenn dies für die gegebene Fragestellung sinnvoll ist. Eine Alternative zu der Aggregation stellt die Generalisierung dar. Bei diesem Vorgehen werden Entitäten, die gemeinsame Attribute aufweisen, einer übergeordneten Enitätsklasse zugeordnet. Beispielsweise können die Entitäten *Kunde* und *Mitarbeiter* in einer Klasse *Partner* zusammengefasst werden, da Name und Adresse für sowohl Kunden als auch Mitarbeiter gespeichert werden und nur diese Werte für eine konkrete Fragestellung relevant sind. Wie auch bei der Aggregation kann auch die Generalisierung neben Entitäten auf einzelne Attribute durchgeführt werden. Ein Beispiel dafür wurde bereits in dem Arbeitsschritt *Datenbereinigung* genannt, in welchem die einzelnen Temperaturwerte einer Heizung in die Klassen *warm*, *mittel* und *kalt* eingeteilt wurden. Sowohl durch die Aggregation als auch die Generalisierung ist es möglich die Anzahl der Attribute zu verringern, die dem Analyseverfahren übergeben werden. Dabei es ist wichtig zu beachten, dass das Verringern der Attribute in der Regel mit einem Informationsverlust einhergeht. In jedem Fall sollte auch hier wieder entschieden werden, ob das Zusammenführen von Werten für die Analyse oder Repräsentation sinnvoll ist oder nicht. Neben der Aggregation von Attributen und Entitäten geht es in diesem Arbeitsschritt aber auch darum einzelne Attribute in ein Format zu bringen, welches die Analyse erwartet. Beispielsweise kann es vorkommen, dass ein Analysetool nicht mit symbolischen Werten umgehen kann und Angaben wie *definitiv ja*, *ja*, *unsicher* und *nein* in numerische Werte umgewandelt werden müssen um von dem Analyseverfahren ausgewertet werden zu können. Auch hier ist es wieder abhängig von dem zu verwendenden Analyseverfahren, in welcher Form die symbolischen Werte umgewandelt werden. Beispielsweise kann eine symbolische Angabe der Monate in die Zahlen 1 bis 12 umgewandelt werden. Allerdings führt diese Art der Repräsentation dazu, dass der Monat „Dezember“ (12) größer ist, als der Monat „Januar“ (1). Wenn alle Monate aber gleich gewichtet sein sollen ist es eine Möglichkeit zwölf neue Attribute, Januar bis Dezember zu erstellen und jeweils auf 1 zu setzen, wenn der Monat zutrifft. Eine andere Art der Transformation ist die Norma-

lisierung. Verschiedene Analyseverfahren erwarten beispielsweise normalisierte Werte im Wertebereich [0..1].

7.4 Datenintegration

Die Datenintegration stellt einen weiteren Arbeitsschritt innerhalb der Datenvorbereitung dar, bei dem es darum geht Daten aus unterschiedlichen Tabellen zusammenzuführen. Innerhalb der Datenbereinigung wurde bereits ein ähnliches Verfahren angewandt um fehlende Daten zu ergänzen. In diesem Arbeitsschritt geht es mehr darum unterschiedliche Daten, allerdings bezüglich des gleichen Objektes zusammenzuführen. Beispielsweise kann es beim Datenimport dazu gekommen sein, dass die Verkaufszahlen für jeden Monat in einer eigenen Tabelle hinterlegt wurden. Durch das Zusammenführen dieser Tabellen kann eine einzige Tabelle erzeugt werden, der alle Verkaufszahlen für alle Monate beinhaltet. Diese Gesamttabelle kann dann einfacher in ein Analyseverfahren eingespeist werden um entsprechende Informationen zu bekommen. Wenn die Tabellen, die zusammengeführt werden sollen allerdings nicht disjunkt sind, entstehen zwangsweise Duplikate. Ein Beispiel dafür stellt das Zusammenfügen von unterschiedlichen Kundentabellen dar, wobei der selbe Kunde in beiden Tabellen gelistet ist. Nun kann es dazu kommen, dass die Adresse des Kunden in beiden Tabellen unterschiedlich ist, sodass entschieden werden muss, welcher Wert als *wahr* angenommen wird. Eine solche Entscheidung kann aufgrund verschiedener Aspekte durchgeführt werden. Beispielsweise kann aufgrund der letzten Aktualisierung entschieden werden, welche Adresse aktueller und somit als richtig anzunehmen ist. Eine andere Möglichkeit ist es aufgrund der Glaubwürdigkeit der Urheber der Quelle zu entscheiden, welche Adresse die korrekte ist. Für einige Attribute, wie in dem Beispiel die Adressdaten, gibt es auch öffentliche Datenbanken, die als Referenztabellen verwendet werden können. In dem Beispiel könnte ein Telefonbuch eine relativ glaubwürdige Referenztabelle darstellen. Aber nicht nur Duplikate stellen ein Problem beim Zusammenführen von Tabellen dar. Sollen beispielsweise Tabelle mit Werten aus dem metrischen und dem angloamerikanischen System zusammengeführt werden muss entschieden werden, welches System in der Gesamttabelle verwendet werden soll. Ähnlich verhält es sich mit Werten die in einer Tabelle in beispielsweise Zentimetern und in einer anderen Tabelle in Metern angegeben werden. Auch hier muss entschieden werden, welche Maßeinheit für die Gesamttabelle sinnvoll ist. Um solche Probleme schon vorher zu beheben, sollten alle Werte bereits im Arbeitsschritt *Datenkonstruktion* normiert werden. Wie auch in jedem anderen Arbeitsschritt kann auch während der Datenintegration auffallen, dass bestimmte Daten fehlen. In einem solchen Fall ist es möglich in den letzten Prozessschritt, das *Datenverständnis*, zurück zu wechseln und die benötigten Daten zu erheben.

7.5 Datentransformation

Der letzte Arbeitsschritt, die *Datentransformation*, beschäftigt sich damit die nun vollständigen, bereinigten, zusammengefügten, normalisierten und normierten Daten in die finale Form zu bringen, um sie in das Analyseverfahren einzuspeisen. Diese letzte Formatierung bezieht sich dabei nicht mehr auf die einzelnen Werte sondern lediglich auf das Format der

Tabellen. Beispielsweise kann das Analyseverfahren verlangen, dass die erste Spalte der eindeutige Bezeichner ist oder in die letzte Spalte das Ergebnis der Analyse geschrieben werden soll. Wenn dieser letzte Arbeitsschritt durchgeführt wurde sind die Daten in einem Format, welches von dem jeweiligen Analyseverfahren akzeptiert wird und es kann mit dem nächsten Prozessschritt, dem *Modelling* fortgefahren werden.

7.6 Vorbereitung der Busdaten

ID	LINE	NODE	NAME	POS	LAT	LONG
1	301a	262.597.256	Eversten	1	53.1303905	8.1559062
2	301a	784.347.967	Ludwig-E...	2	53.1316989	8.1602954
3	301a	365.573.152	Otto-Suh...	3	53.1332822	8.1641666
4	301a	365.573.154	Florianstr...	4	53.1348572	8.1679007
5	301a	352.848.293	Billunger...	5	53.1348780	8.1720453

Abbildung 26: Auszug aus HANA Datenbanktabelle BUS_BASE

Die im Datenverständnis beschriebenen Excel-Tabellen zum Fahrplan und den Abfahrtszeiten sollen in die HANA importiert werden. Vorab muss die Zuordnung der Koordinaten aus den OSM Daten mit den VWG-Daten vorgenommen werden. Hierzu wird die bereits aus den OSM Daten erstellte CSV Datei in Excel geöffnet, die sämtliche Node IDs mit den jeweiligen Koordinaten enthält. In der erstellten Datei mit den Fahrplandaten wird die *SVERWEIS* Funktion genutzt, um die Node ID der Haltestelle in der Node-Koordinaten Tabelle ausfindig zu machen und die Angaben zu Längen- und Breitengrad in die Fahrplantabelle zu übernehmen. Im Anschluss werden die Tabellen *BUS_BASE* und *BUS_TIMETABLE* angelegt. Die BUS_BASE Tabelle spiegelt die erstellte Tabelle zum Fahrplan wieder. Neben einer ID als Primärschlüsselattribut wird die Linie angegeben, die Node ID der Bushaltestelle, der Name der Station, die Position der Haltestelle im Liniennetz sowie die Koordinaten zur Verortung der Haltestelle in der Karte. Die Tabelle *BUS_TIMETABLE* bildet den Fahrplan mit den Abfahrtszeiten ab, es sind wie in der Datenerhebung beschrieben Angaben zu Uhrzeit, Gruppe (für die Unterscheidung von Wochentag, Samstag sowie Sonn- und Feiertag), Haltestelle und Linie enthalten.

7.7 Vorbereitung der Wetterdaten

Zur Integration der Wetterdaten in die HANA Datenbank muss zunächst die Datenbanktabelle in der HANA erstellt werden. Als Tabellenname wird *WEATHERDATA* gewählt. Die unterschiedlichen Aspekte, die vom Wetterdienst aufgezeichnet und bereitgestellt werden sind in der Datenerhebung untersucht worden. Für jeweils potentiell relevante Informationen werden Spalten angelegt. Die in Klammern angegebenen Bezeichner sind vom DWD vergeben und sind Bestandteil der Dateinamen. Als Primärschlüsselattribut dient

der Zeitstempel, wie er aus den Dateien des DWD entnommen wird. Für die Lufttemperatur (TU) liegt ein Wert vor, die Bodentemperatur (EB) wird für die Höhen 0,05 Meter, 0,1 Meter, 0,2 Meter sowie 0,5 Meter erhoben. Weitere Werte sind der Bewölkungsgrad (N), die Niederschlagshöhe (RR), der Luftdruck (P0), die Sonnendauer (SD), für die ein relativer Wert des Anteils an Sonnenschein in der jeweiligen Stunde erhoben wird und die Windgeschwindigkeit sowie –Richtung (FF).

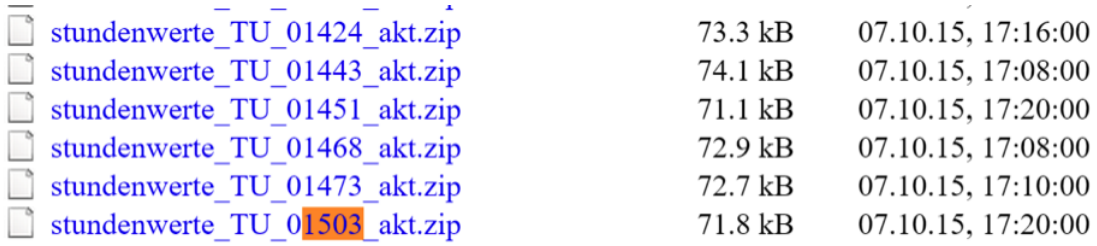


Abbildung 27: Dateiübersicht zur Lufttemperatur auf dem Server des DWD

Für Import der Daten wird ein PHP Skript geschrieben, welches auf unserem Server ausgeführt wird. Das Skript wird nacheinander für die einzelnen Wetterparameter durchlaufen, es stellt zunächst eine Verbindung zum FTP-Server des Deutschen Wetterdienstes her und lädt die ZIP Datei auf den Server des Skripts. Die Server URL ist auf der Internetseite des DWD angegeben, da es sich um einen öffentlichen Zugang handelt werden keine Benutzerdaten benötigt, der Dateiname wird in einer Variable gespeichert. Im String zur Herstellung der Verbindung wird anonymous als User angegeben, gefolgt von der Server URL, dem Pfad zur Datei und der Variable, die den Dateinamen enthält. Nachdem die Datei heruntergeladen wurde wird diese entpackt, die ZIP Datei wird gelöscht, die Wetterdaten befinden sich in der nun vorliegenden TXT-Datei. Zur Verarbeitung wird die Dateiendung von .txt auf .csv geändert, sodass die PHP Methoden zum Umgang mit CSV Dateien genutzt werden können. Der Inhalt der CSV Datei wird in ein Array gespeichert. Anschließend wird die Zeilenanzahl um 25 subtrahiert, da das nun vorliegende Resultat die erste Zeile des zuletzt hinzugefügten Tages darstellt. Per ODBC wird eine Verbindung zur HANA Datenbank hergestellt. Eine For-Schleife mit einer Laufzeitbedingung von $i ¡25 generiert *INSERT-Statements* für die letzten 24 Zeilen. Da als Primärschlüssel in der HANA Datenbanktabelle der Zeitstempel der Wetterdaten-Zeile angegeben ist wird der Insert nur durchgeführt, wenn noch keine Zeile für diesen Zeitpunkt importiert wurde. Ansonsten wird ein Fehler ausgegeben und der Datensatz wird nicht importiert. Zur Prüfung auf neue Datensätze wird ein Cronjob angelegt, der das Skript jeweils im 30 Minuten Takt ausführt. Dadurch wird gewährleistet, dass die aktuellsten Daten direkt nach Aktualisierung der Datei durch den Deutschen Wetterdienst in die HANA Datenbank importiert werden.

7.8 Vorbereitung der ADAC-Daten

Die Datenbereinigung beinhaltet unterschiedliche Methoden zum Korrigieren und Entfernen von Fehlern. Die Fehler können u.a. aus redundanten, inkonsistenten, falsch formatierten oder inkorrekten Daten entstehen. Die Vorgehensweise der Datenbereinigung besteht aus zwei Phasen:

- Standardisierung
- Bereinigung (vgl. [Gmb15]).

7.8.1 Bereinigung der Testdaten

Die Bereinigung der Testdatensätze wurde mit SPSS durchgeführt. Während der Bereinigung der Testdaten wurden leere Felder, sowie unbekannte Werte durch ein „-“ Pufferzeichen ersetzt. Anschließend wurden doppelt und mehrfach vorkommende Fahrzeugmodelle wie in Abbildung 28 dargestellt, zusammengefasst.

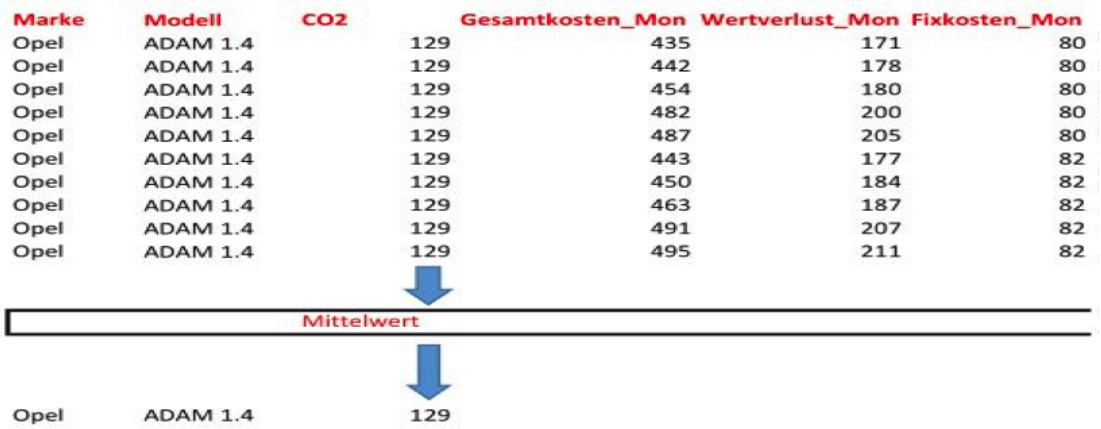


Abbildung 28: Zusammenfassung mehrfach vorkommende Fahrzeugmodelle

In Abbildung 28 sind Informationen über die Fahrzeugmarke, das Fahrzeugmodell, die CO2-Emissionen in Gramm pro KM, die Gesamtkosten in Euro pro Monat, den Wertverlust in Euro pro Monat und die gesamten Fixkosten in Euro pro Monat dargestellt. Wie in der Abbildung zu erkennen ist, kommt das Fahrzeugmodell ADAM 1.4 der Marke Opel mehrfach vor. Ebenso ist der CO2-Wert bei allen vorkommenden ADAM 1.4 Modellen identisch. Eine Unterscheidung liegt vor allem bei den Werten für die Gesamtkosten in Euro pro Monat, Wertverlust in Euro pro Monat und die gesamten Fixkosten in Euro pro Monat vor. An dieser Stelle wurde ein Durchschnittswert nur für den CO2-Wert ausgerechnet. Eine Berechnung des Durchschnittswertes für die Gesamtkosten in Euro pro Monat, den Wertverlust in Euro pro Monat und die gesamten Fixkosten in Euro pro Monat war für die Schadstoffbelastung irrelevant und sind daher aus der Liste entfernt wurden. Durch diese Vorgehensweise konnte der Testdatensatz auf ca. 300 Daten optimiert werden.

7.8.2 Bereinigung der Gesamtdaten

Die Bereinigung des ADAC-Gesamtdatensatzes erfolgt analog anhand des Testdatensatzes. Zunächst wurden leere Felder, sowie fehlerhafte Angaben durch ein „-“ Pufferzeichen ersetzt. Dann wurden Umlaute wie (Flüssiggas durch Fluessiggas, Geländewagen durch Gelaendewagen, Schrägheck durch Schraegheck) ersetzt und fehlende Klammern eingefügt. Im Anschluss wurden diese Daten in eine neue Tabelle übertragen. Da nicht alle Daten für das Projekt relevant sind, wurden die folgenden Daten ausgeschlossen:

- Eindeutige ADAC Fahrzeug-ID
- Kennzeichnung auf Modellaktualität
- Fahrzeugtyp
- Produktionsstart des einzelnen Modells
- Produktionsende des einzelnen Modells
- Baureihe
- Produktionsstart der Baureihe
- Produktionsende der Baureihe
- Karosserieart
- Gesamtaufpreis für übliche Extras
- Fahrzeug-Listenpreis inkl. Extras
- Getriebeart
- Kraftstoffart bzw. Qualität
- Leergewicht in kg
- Zulässiges Gesamtgewicht in kg
- Anzahl der Türen
- Anzahl der Sitze
- Anzahl der Sitze maximal
- 2. EU-Verbrauch gesamt
- 2. EU-Verbrauch innerorts
- 2. EU-Verbrauch außerorts
- Gesamtkosten in Euro pro Monat

- Gesamtkosten in Cent pro Kilometer
- Wertverlust in Euro pro Monat
- Gesamte Betriebskosten in Euro pro Monat
- Gesamte Fixkosten in Euro pro Monat
- Gesamte Werkstatt- und Reifenkosten pro Monat
- Jahresbeitrag für die Haftpflichtversicherung
- Jahresbeitrag für die Teilkaskoversicherung
- Jahresbeitrag für die Vollkaskoversicherung

Von einer hohen Bedeutung sind hingegen Daten und Informationen, die für eine Berechnung von Schadstoff-Werten benötigt werden. Folgende Daten wurden dafür übernommen:

- Fahrzeugmarke
- Fahrzeugmodell
- Fahrzeugklasse
- Grundpreis
- Motoren-Art
- Hubraum in ccm
- Leistung in kW
- Leistung in PS
- CO2-Emission in Gramm pro KM
- CO2-Effizienzklasse
- Kraftstoffart
- Schadstoffklasse
- EU-Verbrauch gesamt
- EU-Verbrauch innerorts
- EU-Verbrauch außerorts

In der Datenverständnisphase wurde festgestellt, dass bei mehreren Fahrzeugmodellen der Grundpreis, der CO2-Wert, die CO2-Effizienzklasse, der Gesamtverbrauch, den Gesamtverbrauch Innerorts und der Gesamtverbrauch Außerorts fehlen. Durch die Kombination der Datenbasis konnte die Qualität der Daten verbessert werden. Damit die Erhöhung der Qualität weiter gesteigert wird, wurden im Internet einige Recherchen durchgeführt und auf der Webseite der deutsche Handwerks Zeitung festgestellt, dass der CO2-Wert vom Gesamtverbrauch des Fahrzeugs abhängig ist. Für die Berechnung des CO2-Wertes, wurde die folgende Formel verwendet:

$$CO_2Wert = \frac{Verbrauch * Ausstoßwert}{100km} *$$

Der Ausstoßwert ist abhängig von der Kraftstoffart. So wird für die Kraftstoffart Benzin der Ausstoßwert „23,8“, Diesel „26,5“, Autogas „17,8“ und Erdgas „27,4“ verwendet (vgl. [DHZ05]). Im Folgenden wird ein Beispiel für die Berechnung des CO2-Wertes bei einem Benzinfahrzeug dargestellt:

$$x_{arith} = \frac{1}{n} \sum_{i=1}^{n} x_i = \frac{x_1 + x_2 + ... + x_n}{n}$$

Nachdem die Bereinigung der Daten abgeschlossen werden konnte, wurden die ADAC-Daten nach den vier Kriterien, „Fahrzeugmarke“, „Fahrzeugmodell“, „Kraftstoffart“ und „Fahrzeugklasse“ sortiert. Anschließend wurde ein Mittelwert für die einzelnen Fahrzeugmodelle berechnet, um eine Aussage über den Kraftstoffverbrauch, CO2-Ausstoß, Grundpreis, den Hubraum und die PS-Anzahl der einzelnen Modelle zu treffen. Für die Berechnung des Mittelwerts wurde die folgende Formel verwendet:

$$x_{prozent} = \frac{x}{n} * 100$$

Bei der Optimierung der Daten wurden, die Fahrzeugmodelle nach der Kraftstoffart getrennt, um deren offensichtlichen Einfluss auf den Kraftstoffverbrauch und die CO2-Belastung zu berücksichtigen. Im Anschluss daran, konnten alle Fahrzeugmarken in eine neue Tabelle zusammengeführt und in die SAP HANA importiert werden. Die ADAC-Tabelle umfasste ursprünglich ca. 61.000 Datensätze und wurde durch die Optimierung auf knapp 6150 Datensätze reduziert. Im Verlauf des Projekts war es leider nicht möglich, genauere Informationen über zugelassene Fahrzeuge im Stadtgebiet von Oldenburg zu erlangen. Um dennoch eine Verwendungszweck zu finden, wurden die ADAC-Daten zusammen mit den Daten des Kraftfahrt Bundesamtes verknüpft um für jede Fahrzeugmarke den CO2- Durchschnittswert zu berechnen. Hierfür wurde zunächst der Durchschnittswert der einzelnen Fahrzeugmarken in der optimierten Tabelle mit 6150 Datensätze berechnet. Diese Werte wurden anschließend in eine neue Tabelle übertragen. Durch die veröffentlichen Daten des Kraftfahrt-Bundesamtes (Bestand an Personenkraftwagen am 1. Januar 2014 nach Herstellern, Handelsnamen und Bundesländern), konnte eine abgeleitete, aber überaus konkrete Anzahl der zugelassenen Fahrzeuge in Niedersachsen ermittelt werden.

In der neuen Tabelle wurden Fahrzeugmarken, die in Niedersachsen angemeldet sind und in der ADAC-Tabelle nicht vorhanden waren, übertragen. Im nächsten Schritt wurde eine neue Spalte mit der Bezeichnung „CO2-Werte“ eingefügt. In dieser Spalte konnte anhand der bereits vorhandenen ADAC-Daten, die CO2-Werte abgeleitet und eingetragen werden. Nachdem die CO2-Werte abgeleitet wurden, konnte eine weitere Spalte mit der Bezeichnung „Prozentanteil der Fahrzeuge in Niedersachsen“ hinzugefügt werden. Der Prozentanteil der einzelnen Marken wurde mit der folgenden Formel berechnet:

$$x_{prozent} = \frac{x}{n} * 100$$

Abschließend wurden die optimierten Daten aus beiden Listen in einer neuen Tabelle zusammengeführt, um daraus einen CO2-Durchschnittswert zu berechnen. Der CO2-Wert wurde anschließend in die SAP HANA übertragen.

7.9 Vorbereitung der OpenStreetMap-Daten

Im Kapitel Vorbereitung der OpenStreetMap-Daten wurde die zugrunde liegende Darstellungsbasis für Verkehrsdaten im Großraum Oldenburg zusammengetragen. Die OSM Datenmenge umfasst rund 1.359.707 loc (Lines of Code) die im folgenden Kapitel bereinigt werden. Als Grundgedanke dazu, wird der Vorgang der Datenbereinigung in zwei Schritte aufgeteilt. Zunächst wird die generierte OSM-XML Datei mittels eines eigens dafür entwickelten Parsers im Unterkapitel Parser aufgesplittet und bereinigt. Im Anschluss daran werden die vorhandenen Abschnitte konvertiert und in ein datenbankfreundliches Dateiformat, dem CSV Format, gebracht. Dies wird im Unterkapitel Konverter beschrieben.

7.9.1 Parser

Bevor die Bereinigung der Datenbasis durchgeführt werden konnte, wurde die OSM Datenmenge in insgesamt 45 kleinere Abschnitte eingeteilt. Die Aufsplittung der OSM Datenmenge bot mehrere Vorteile. Zum einen konnte dadurch ein Performancegewinn des Parsers erzielt und zum anderen fehlerhafte Arbeitsschritte auf einzelne Bereiche eingeschränkt und identifiziert werden. Des Weiteren wurde die vorhandene Hardware weniger beansprucht, sodass etwaige Programmabstürze vermieden werden konnten. Im weiteren Verlauf war es wichtig, alle nicht benötigten Elemente wie „Tags“, „Relations“, Versionsinformationen und Zusatzinformationen an den „Ways“ zu bestimmen und als Input für den Parser zu kennzeichnen. Die Tabellen 3 - 5 geben einen Überblick über aussortierte Elemente.

Tabelle 3: Nicht benötigte „Tags“ – vorwiegend „Amenitys“

Element	Beschreibung
animal_boarding	Tierpension
animal_shelter	Tierheim
atm	Standort eines Geldautomaten
arts_centre	Kunstzentrum
bank	Geldinstitut
baby_hatch	Babyklappe
bbq	Grillplatz
bench	Parkbank
biergarten	Biergarten
bicycle_parking	Fahrrad-Parkplatz
bicycle_repair_station	Öffentliche Fahrrad -Reparaturstation
bicycle_rental	Fahrrad-Verleih
bureau_de_change	Wechselstube
boat_sharing	Boatsharing
casino	Kasino
car_rental	Autovermietung
car_sharing	Carsharing
car_wash	Waschanlage
charging_station	Stromtankstelle
clock	Uhr
community_centre	Gemeindezentrum
courthouse	Gericht
coworking_space	Geteilter Arbeitsbereich
crematorium	Krematorium
crypt	Mausoleum
doctors	Arzt
dojo	Trainingsplatz für japanische Kampfkünste
drinking_water	Öffentlich zugängliche Trinkwasserstelle
embassy	Botschaft
ferry_terminal	Fähranleger
firepit	Stelle im Freien zum Feuermachen
food_court	Eine Vielzahl von Restaurants
fountain	Springbrunnen
grave_yard	Friedhof
grit_bin	Spezieller Streucontainer
gambling	Platz für Glücksspiel
game_feeding	Fütterplatz

hunting_stand	Jagdstand
kindergarten	Kindergarten
library	Bibliothek
nursing_home	Altenheim
pharmacy	Apotheke
photo_booth	Stand für sofort-Bilder
place_of_worship	Gedenkstätte/Andachtstätte
planetarium	Planetarium
recycling	Recycling-Ort
rescue_station	Rettungsstation
police	Polizei
post_box	Briefkasten
post_office	Post-Station
prison	Gefängnis
public_bookcase	Öffentlicher Bücherkasten
public_building	Öffentliches Gebäude
ranger_station	Öffentliches Gebäude/Polizei und Information
register_office	Registrierungsbüro
sauna	Sauna
shelter	Wetterschutz
shower	Dusche
social_facility	Soziale Einrichtung
social_centre	Soziale Einrichtung
studio	Ton- oder Fernsehstudio
taxi	Taxistand
theatre	Theater
toilets	Öffentliche Toilette
townhall	Rathaus
veterinary	Tierarzt
vending_machine	Gerät zum Verkauf von Waren
watering_place	Tränke

Tabelle 4: Nicht benötigte „Relations“:

Element	**Beschreibung**
power	Hochspannungsleitungsmast
Wahlkreis	Wahlkreis

Mein Fernbus	Station von MeinFernbus
MeinFernbus	Station von MeinFernbus

Tabelle 5: Nicht benötigte Versionsinformationen:

Element	Beschreibung
changeset	Hochspannungsleitungsmast
timestamp	Wahlkreis
uid	Station von MeinFernbus
user	Station von MeinFernbus
version	Aktuelle Änderungsversion

Nicht benötigte Zusatzinformationen an „Ways“:
Bei der Berücksichtigung von Zusatzinformationen an den „Ways“, wurden nur diejenigen Wege berücksichtigt, welche den „Tag“ „Highway“ mit Ausprägungen und folgenden Informationen enthielten:

- Highway=motorway
- Highway=trunk
- Highway=primary
- Highway=secondary
- Highway=tertiary
- Highway=unclassified
- Highway=residential
- Highway=service

Um die geeignete Bearbeitung der OSM XML Datenbasis durchzuführen, werden die Java Bibliotheken javax.xml.* und org.w3c.dom* benötigt. Zunächst wird mit Hilfe der „DocumentBuilderFactory“ und dem „DocumentBuilder“ eine XML Datei mit DOM eingelesen und vollständig in den Hauptspeicher als Baumstruktur geladen.

```
public class osm_parser {
   public static void main(String[] args) {
       // start parsing document
       System.out.println("start parsing document");
       try {
           //break down XML structure to DOM tree
           DocumentBuilderFactory documentBuilderFactory =
              DocumentBuilderFactory
                    .newInstance();
           DocumentBuilder documentBuilder =
              documentBuilderFactory
                    .newDocumentBuilder();
           Document document = documentBuilder
                    .parse("insert XML document to parse");
```

Der daraus resultierende Gesamtbaum ermöglicht die individuelle Bearbeitung des eingelesenen Dokuments und erleichtert den Zugriff auf einzelne Knoten. Die Architektur des Parsers ist so aufgebaut, dass nacheinander sämtliche „Nodes“, „Ways“ und „Relations“ abgearbeitet werden. Im ersten Schritt wird zunächst die Bearbeitung der Nodes dargestellt, welche bereits die Implementierung aller nicht benötigten Elemente initiieren.

```
           // editing nodes
           System.out.println("editing nodes..");
           // show ROOT element
           Element theRoot = document.getDocumentElement();
           // get element node
           NodeList nodes = document.getElementsByTagName("
              node");
           // userdefined changeinformations
           String[] delAttr = {[...]};
           // get acces to amenity on v tag
           String[] amenity = {[...]};
           // get acces to k or v tag
           String[] deta = {[...]};
           // get acces to k or v tag without deleting father
              root
           String[] reduData = {[...]};
           // ceating JAVA HashSet for target elements
           Set<Element> targetElementNodes = new HashSet<
              Element>();
           // first for loop for node access
           for (int r = 0; r < nodes.getLength(); r++) {
               Node delTagCreate = (Node) document
               .getElementsByTagName("node").item(r);
               NodeList helpNodes = delTagCreate.getChildNodes
                  ();
               Node innerNode = nodes.item(r);
               // search for userdefined changesets and remove
```

```
 for (int e = 0; e < delAttr.length; e++) {
     String value = delAttr[e];
     Element help = (Element) innerNode;
        if (help.hasAttribute(value)) {
            help.removeAttribute(value);
        }
    }
    // set actual element node
    Element spectNode = (Element) innerNode;
    // search for child nodes
    if (innerNode.hasChildNodes()) {
        for (int k = 0; k < helpNodes.getLength
           (); k++) {
            Node element = helpNodes.item(k);
            // deleting amenity data
             for (int f = 0; f < amenity.length;
                 f++) {
                String value = amenity[f];
    }   }    }
// trim white space in case of deleted elements
for (Element e : targetElementNodes) {
    Node prev = e.getPreviousSibling();
    if (prev != null && prev.getNodeType() ==
       Node.TEXT_NODE
            && prev.getNodeValue().trim().length
               () == 0) {
        theRoot.removeChild(prev);
    }
    e.getParentNode().removeChild(e);
}
```

Die Bearbeitung der Nodes beginnt zunächst mit der Definition der nicht benötigten Elemente. Dazu wurden die Arrays „dellAttr“, „Amenity“, „deta und „ReduData“ vom Typ String nacheinander erstellt. Im Anschluss daran beginnt eine for-Schleife mit der Identifizierung aller Node Elemente die als Namen „Tag“ angegeben haben. Mit Hilfe der Methode getElementsByTagName() aus dem importierten org.w3c.dom Paket kann ein spezielles Element ohne weiteres benannt werden. Im gleichen Verarbeitungsschritt werden die nicht benötigten Versionsinformationen identifiziert und gelöscht. Die nachfolgende If -Anweisung überprüft ob der betrachtete Node weitere Unterelemente hat. Dazu wird die Methode hasChildNodes() verwendet. Sind Unterelemente vorhanden, durchläuft eine weitere for Schleife alle Unterelemente des Nodes mit Hilfe der getLenth() Methode. Weist ein betrachtetes Element eine Übereinstimmung mit den in den Arrays hinterlegten nicht benötigten Elementen auf, so wird dieses gelöscht. Um den so genannten „Whitespace“ nach dem löschen zu vermeiden, wird abschließend eine for Schleife ausgeführt die Text Nodes ohne Inhalt aufspürt und mit Hilfe der Methode removeChild() vom Root Element löscht. Die Hilfs- Methode clearChildNodes() nimmt eine rekursive Löschung aller Unter-

elemente eines Nodes vor.

```
private static void clearChildNodes(Node n) {
    Node node = n;
    while (node.hasChildNodes()) {
        NodeList nList = node.getChildNodes();
        int index = node.getChildNodes().getLength() - 1;
        Node no = nList.item(index);
        node.removeChild(no);
        clearChildNodes(no);
}   }
```

Der übergebene Node wird im internen Node node gespeichert. Die While Schleife durchläuft daraufhin alle Unterelemente des Nodes. Über eine zweite NodeListe mit Index und dem erneuten rekursiven Aufruf der Methode können alle Unterelemente entfernt werden. Im zweiten Schritt werden die „Ways“ betrachtet. Dabei ist die Bestimmung der relevanten Informationen mit dem „Tag“ „Highway“ essentiell wichtig.

```
// editing ways
System.out.println("editing ways...");
// creating nodelist with all way data
NodeList delNodeList = document.
   getElementsByTagName("way");
//creating JAVA hashset for target elements
Set<Element> targetElementWays = new HashSet<
   Element>();
    // counter for while loop
int z = 0;
// pointer for extern method
boolean pointer = false;
while (z < delNodeList.getLength()) {
   Node delNode = delNodeList.item(z);
// get special way element with item z
   Node way =(Node)document.getElementsByTagName("
      way").item(z);
   NodeList helpNodes = way.getChildNodes();
   Element spectWay = (Element) delNode;
   for (int k = 0; k < helpNodes.getLength(); k++)
      {
      Node element = helpNodes.item(k);
      // identify node name of tag
      if ("tag".equals(element.getNodeName())) {
         NamedNodeMap attribute = helpNodes.item(k)
            .getAttributes();
         Node nodeAttr = attribute.getNamedItem("k"
            );
         // identify tag highway
```

```
                if(nodeAttr.getTextContent().matches("
                    highway")) {
                    // set pointer true if highway
                    pointer = true;
             } } }
            // if pointer position false delete all child
               nodes of node
            if (pointer == false) {
                clearChildNodes(delNode);
                targetElementWays.add(spectWay);
            }
            Node delWayAttr = delNode;
            // delete userdefined informations if possible
            for (int i = 0; i < delAttr.length; i++) {
                String value = delAttr[i];
                Element help = (Element) delWayAttr;
                if (help.hasAttribute(value)) {
                    help.removeAttribute(value);
            }   }

            pointer = false;
            z++;
        }
        // trim white space in case of deleted elements
        [...]
```

Zunächst wird die Gesamtanzahl aller „Way“ Elemente bestimmt, welche als Referenz für den Counter der while Schleife dienen. Innerhalb der Schleife werden dann alle Elemente und Unterelemente mit dem „Tag“ Namen „Way“ ermittelt. Die nachfolgende for Schleife bestimmt die Position des Elements und setzt den Pointer vom Typ boolean auf „true“ falls der „Tag“ „Highway“ vorhanden ist. Ist dies nicht der Fall bleibt die Position des Pointers auf false und die Hilfs- Methode clearChildNodes() löscht alle Unterlemente des Nodes. Auch innerhalb der „Ways“ müssen gesondert Versionsinformationen und „Whitespaces“ erkannt und gelöscht werden. Dies passiert analog zur bereits vorgestellten Vorgehensweise in den „Nodes“. Im dritten Schritt müssen die Relations angepasst werden. Dazu ist es erforderlich alle Relations zu identifizieren die zum einen durch gelöschte „Ways“ nicht mehr referenziert werden und zum anderen nicht benötigte Elemente aus dem definierten Array deta wiederspiegeln.

```
// editing relations
System.out.println("editing relations...");
//creating JAVA hashset for target elements
Set<Element> targetElementRela = new HashSet<Element
   >();
// creating nodelist with all relation data
NodeList relations = document.getElementsByTagName("
   relation");
```

```
// for loop for node access with name relation
for (int i = 0; i < relations.getLength(); i++){
    Node innerRela = relations.item(i);
    // get special element relation with position i
    Node delTagRela = (Node) document.
       getElementsByTagName("relation").item(i);
    NodeList helpRela = delTagRela.getChildNodes();
    // delete all userdefined informations if
       possible
    for (int e = 0; e < delAttr.length; e++) {
        String value = delAttr[e];
        Element help = (Element) innerRela;
        if (help.hasAttribute(value)) {
            help.removeAttribute(value);
    }   }
    Element spectRel = (Element) innerRela;
    // if spectRel innerRela has childs
    if (innerRela.hasChildNodes()) {
        for(int j = 0; j < help.getLength(); j++) {
            Node element = helpRela.item(j);
            // search for element with name tag
            if("tag".equals(element.getNodeName()){
                NamedNodeMap attribute = helpRela.
                   item(j).getAttributes();
                // get attribute k and v
                Node nodeAttrK = attribute.
                   getNamedItem("k");
                Node nodeAttrV = attribute.
                   getNamedItem("v");
                for (int g = 0; g < deta; g++) {
                    String value = deta[g];
                  // delete relation information
                     which are unnecessary
                  if(nodeAttrV.getTextContent().
                     matches(value)){
                     clearChildNodes(innerRela);
                     targetElementRela.add(spectRel);
}   }  }   }    } }
// trim white space in case of deleted elements
[...]
```

Ähnlich wie bei der Bereinigung der „Ways“ werden eine HashSet sowie eine NodeList mit allen „Tags“ gebildet, die den Namen „Relation“ beinhalten. Die erste for Schleife durchläuft wie gewohnt die NodeList und liefer mit Hilfe des Indexes i die aktuelle Position zurück. In umgekehrter Reihenfolge werden in der zweiten for Schleife zunächst nicht benötigte Versionsinformationen erkannt und gelöscht. Die nachfolgende if Schleife überprüft ob der betrachtete Node an der Position des Indexs i Unterelemente besitzt. Ist dies der Fall, durchläuft eine for Schleife alle Unterelemente und löscht diese anhand der in

dem Array deta angegebenen Informationen. Die letzte for Schleife nimmt eine erneute Löschung von entstandenen „Whitespaces“ vor. Abschließend muss das bereinigte OSM-XML Dokument in die Ausgangsform zurück gebracht werden.

```
            // Build new XML Document
            System.out.println("build new document...");
            // building new instance
            TransformerFactory transformerFactory =
               TransformerFactory.newInstance();
            Transformer transformer = transformerFactory.
               newTransformer();
            // create dom source from document
            DOMSource domSource = new DOMSource(document);
            // stream result to new file
            StreamResult streamResult = new StreamResult(new File("
               new source .xml"));
            // transform result and stream
            transformer.transform(domSource, streamResult);
            System.out.println("done writing document");
        } catch (ParserConfigurationException pce) {
            pce.printStackTrace();
        } catch (TransformerException tfe) {
            tfe.printStackTrace();
        } catch (IOException ioe) {
            ioe.printStackTrace();
        } catch (SAXException sae) {
            sae.printStackTrace();
}   }
```

Zunächst stellt eine neue Instanz der TransformerFactory() aus dem Paket javax.xml.* einen Transformer bereit. Das geänderte DOMDocument wird mittels DOMSource als Input für den Transformer definiert. Der Output kann als StreamResult realisiert werden und gibt die bereinigte OSM-XML Datei aus, welche im nachfolgenden Kapitel mit Hilfe eines Konverters in das CSV Format überführt wird.

7.9.2 Konverter

Nach der Bereinigung der OSM-XML Daten ist es wichtig, die Fülle der neuen Informationen in ein datenbankfreundliches Format zu bringen. Hier bietet sich besonders das CSV Format an, da ein schneller und einfacher Import in die SAP HANA Datenbank durchgeführt werden kann. Zu diesem Zweck wird neben dem bereits vorgestellten Parser auch ein Konverter benötigt, der die Informationen adäquat überführt und darstellt. Um die Übersichtlichkeit zu verbessern, werden die Informationen in vier verschiedenen CSV Dateien aufgespalten:

- „Cords“ (Node)

- „Meta“ (Node)
- „Way“
- „Relation“

Der Aufbau der unterteilten CSV Dateien wird mit Hilfe von individuellen XSL Stylesheets vorgenommen. Die bereits erwähnten Java Klassen javax.xml.* und org.w3c.dom.* stellen Konzepte bereit, die als Grundlage für die Umsetzung des Konverter dienen.

```
public class xmlcsv_converter {
    public static void main(String[] args) {
        String xmlPath = "path to xml file";
        String csvNodes = "path for new csv file";
        try {
            System.out.println("starting converter");
            // path of stylesheet form for csv file
            File stylesheetNodes = new File(
                    "path to stylesheet for file");
```

Zunächst wird der Pfad zu einer bestehenden OSM-XML Datei im String xmlPath eingetragen. Der String csvNodes repräsentiert wiederum den Pfad der neuen CSV Datei. Der Konverter beginnt mit dem Einlesen der jeweils zuständigen XSL Stylesheets, welche den Aufbau der neuen CSV Datei wiederspiegeln. Im Folgenden werden vier individuelle XSL Stylesheets vorgestellt, die in Anlehnung an der erwähnten Aufteilung in „Cords“, „Meta“, „Way“ und „Relation aufgeteilt sind. Der Stylesheet „Cords“ extrahiert koordinatenbezogene Informationen aller Nodes einer bestehenden OSM-XML Datei.

```
<xsl:template match="/">
     node_id;latitude;longitude
    <xsl:for-each select="//node">
        <xsl:value-of select="@id" />
        <xsl:text>;</xsl:text>
        <xsl:value-of select="@lat" />
        <xsl:text>;</xsl:text>
        <xsl:value-of select="@lon" />
        <xsl:text>;</xsl:text>
        <xsl:text>&#10;</xsl:text>
   </xsl:for-each>
</xsl:template>
```

Mittels einer for-each Schleif werden zunächst alle Elemente bestimmt, die in einem Node vorhanden sind. Zu jedem identifiziertem Node werden die dazugehörigen Koordinaten sowie eine einmalige ID herausgefiltert. Die Tabelle 6 stellt eine abschließende Übersicht aller verwendeten Elemente bereit und zeigt den Aufbau der CSV Datei „Cords“. Das Element „Id“ fungiert als Primärschlüssel zur tabellenübergreifenden Identifikation von Informationen eines Nodes.

Tabelle 6: Aufbau der CSV Datei „Cords“

Element	Beschreibung
Id	Node_ID
lat	Latitude
lon	Longitude

Neben den Koordinaten eines Nodes sind auch Metainformationen vorhanden die gesondert durch den Stylesheet „Meta“ betrachtet werden.

```
<xsl:template match="/">
    <xsl:for-each select="//node">
        <xsl:value-of select="@id" />
        <xsl:text>;</xsl:text>
        <xsl:if test="tag[@k='TMC:cid_58:tabcd_1:Class']">
        <xsl:value-of select="tag[@k='TMC:cid_58:tabcd_1:Class']/@v
            [1]" />
        </xsl:if>
        <xsl:text>;</xsl:text>
        [...]
        <xsl:text>&#10;</xsl:text>
   </xsl:for-each>
</xsl:template>
```

Die for-each Schleife betrachtet für die Extraktion der Metainformationen auch in diesem Fall jeden einzelnen Node. Durch aufeinanderfolgende If Abfragen können gefundene Informationen lokalisiert und für den Aufbau der neuen CSV Datei verwendet werden. In der Tabelle 7 werden alle Elemente dargestellt, die das Stylesheet verarbeitet. Das Elment „Id“ repräsentiert erneut einen Primärschlüssel um koordinatenbezogene Informationen mit Metainformationen zu verknüpfen.

Tabelle 7: Stylesheets Nodes

Element	Beschreibung
Id	Node_ID
TMC:cid_58:tabcd_1:Class	TMC-Navigation- Klasse
TMC:cid_58:tabcd_1:Direction	TMC-Navigation-Richtung
TMC:cid_58:tabcd_1:LCLversion	TMC-Navigation-Version
TMC:cid_58:tabcd_1:LocationCode	TMC-Navigation- Aktuelle Area
TMC:cid_58:tabcd_1:NextLocationCode	TMC-Navigation- Nächste Area

TMC:cid_58:tabcd_1:PrevLocationCode	TMC-Navigation- Vorherige Area
Addr:city	Name Stadt wie in postalischer Adresse angegeben
Addr:country	Ländercode wie in postalischer Adresse angegeben
Is_in	Ortsangabe
Addr:housenumber	Hausnummer wie in postalischer Adresse angegeben
Addr:postcode	Postleitzahl wie in postalischer Adresse angegeben
Addr:street	Straßenname wie in postalischer Adresse angegeben
Addr:suburb	Ortsteil wie in postalischer Adresse angegeben
Place	Area
Name	Name einer Area, eines Ortes oder eines Geschäfts
Shop	Geschäft
Operator	Geschäftsbetreiber
Service	Geschäftsart
Sports	Spezifizierung eines Sportgeschäfts
Building	Gebäude aller Art
Information	Informationsstätte
Office	Büro
Emergency	Erste Hilfe Equipment
Tourism	Tourismusplätze
Leisure	Freizeitort/Möglichkeit
Highway	Straßen und Wege aller Art
Name	Straßen- und Wegenamen
Ref	Refferenznummer von Straßen und Wegen
Traffic_sign	Verkehrszeichen
Noexit	Ausfahrt nicht möglich
Crossing_ref	Straßenübergang, Art des Übergangs
Laterne	Straßenlaterne
Material	Material aus dem das betrachtete Objekt besteht
Bdouble	Erlaubnis für EuroCombi LKWs
Railway	Transportformen die auf Metallschienen basieren
Name	Name/Bezeichnung der Schienennutzung

Disused	Stillgelegter Schienenabschnitt
Crossing:barrier	Zeigt Beschrankung eines Bahnübergangs an
Crossing:light	Zeigt an ob ein Warnlicht an einem Bahnübergang vorhanden ist
Wheelchair	Zeigt ob der Bahnzugang mit einem Rollstuhl möglich ist
Railway:position	Positionsangabe entlang von Bahnlinien
Railway:position:exact	Exakte Positionsangabe
Railway:switch	Eisenbahnweiche
Railway:turnout_side	Gibt die Richtung der Abzweigung an
Access	Rechtliche Zugangsregelung eines Elements
Ref	Referenznummer einer Zugangsregelung
Hgv	Zugangsregelung für LKWs über 3,5 Tonnen
Cargo	Gibt die Güterart an
Entrance	Position eines Eingangs
Private	Gibt an ob die Nutzung im allgemeinen öffentlich zugänglich oder nur privat möglich ist
Barrrier	Barriere
Bicycle	Zugangserlaubnis für Fahrräder
Foot	Zugangserlaubnis für Fußgänger
Horse	Zugangserlaubnis für Reiter
Noexit	Sackgasse
Amenity	Markierung nützlicher und wichtiger Einrichtungen
Name	Name der Einrichtung
Operator	Betreiber der Einrichtung
Fee	Gebühr für die Nutzung der Einrichtung
School	Schule
Name	Name einer Busverbindung
Public_transport	Spezifiziert öffentliche Verkehrsmittel
Network	Verkehrsverbund
Network:zone	Zone eines Verkehrsverbundes
Operator	Betreiber des Verkehrsverbundes oder Busverbindung
Ref	Referenznummer des Betreibers
Route	Route einer Busverbindung
Towards	Durchgangsstation einer Busverbindung

Wheelchair	Zeigt ob der Buszugang mit einem Rollstuhl möglich ist
Tactile_paving	Blindenstock Tastpunkte an Fußwegen
Waterway	Wasserweg
Aeroway	Luftweg
Obstacle	Hindernis
Obstacle_name	Name des Hindernisses
Seamark:name	Name eines Navigations-Objekts aus OpenSeaMap
Seamark:type	Typ eines Navigations-Objekts aus OpenSeaMap
Seamark:bridge:category	Klassifizierung einer Brückenkategorie die einen Seeweg kreuzt
Seamark:bridge:clearance_height	Vertikaler Abstand in Metern gemessen für feste/nicht bewegliche Brücken
Seamark:bridge:clearance_height_closed	Vertikaler Abstand in Metern gemessen für bewegliche Brücken
Seamark:harbor:category	Spezifiziert die Art eines Anlegepunktes
Seamark:small_craft_facility:category	Spezifiziert die Ausstattung eines Anlegepunktes
Seamark:buoy_lateral:category	Gibt die Kategorie einer Boje an
Seamark:buoy_lateral:colour	Zeigt die Leuchtfarbe bei Nacht einer Boje an
Seamark:buoy_lateral:shape	Form einer Boje
Seamark:buoy_lateral:system	Regionsabhängige Nutzung von Systemen

Ein essentiell wichtiger Part übernimmt der Stylesheet „Way“. Hier werden alle Informationen zusammengetragen, die einen Weg identifizieren. Durch die Festlegung des Primärschlüssels „Id“ kann eine spätere Zuordnung in den Relations erfolgen. Zudem werden Routen definiert, die über bereits festgelegte Primärschlüssel aus den Nodes sukzessive aufgebaut werden können.

```
<xsl:template match="/">
    <xsl:for-each select="//way">
        <xsl:value-of select="@id" />
        <xsl:text>;</xsl:text>
        <xsl:if test="tag[@k ='highway']">
            <xsl:value-of select="tag[@k= 'highway']/@v[1]" />
        </xsl:if>
        <xsl:text>;</xsl:text>
        [...]
```

```
        <xsl:text>;</xsl:text>
        <xsl:for-each select="nd">
            <xsl:value-of select="@ref" />
            <xsl:text>;</xsl:text>
        </xsl:for-each>
        <xsl:text>&#10;</xsl:text>
    </xsl:for-each>
</xsl:template>
```

Zunächst durchläuft die erste for- each Schleife alle Elemente, die mit einem „Way“ Tag gekennzeichnet sind. Analog zu den bereits vorgestellten Stylesheets werden mit Hilfe der If Abfragen relevante Informationen gekennzeichnet und in die Struktur der neuen CSV Datei übertragen. Da einem Way standardmäßig mehrere Nodes zugeordnet sein können, extrahiert eine zweite for-each Schleife alle korrespondierenden Nodes, die dem in der ersten for-each Schleife betrachtetem Node entsprechen. Die nachfolgende Tabelle 8 gibt eine Übersicht über Elemente, die im Stylesheet verarbeitet werden.

Tabelle 8: Stylesheets Ways

Element	**Beschreibung**
Id	Way_ID
Highway	Straßen und Wege aller Art
Railway	Transportformen die auf Metallschienen basieren
Restriction	Sammlung von Abbiegevorschriften
Access	Rechtliche Zugangsregelung eines Elements
Destination	Zugangsregelung für Anliegerverkehr
Vehicle	Zugangsbeschränkung für Kraftfahrzeuge aller Art
Motorcar	Offizielle Zugangserlaubnis für PKW
Motorcycle	Offizielle Zugangserlaubnis für Motorräder
Motor_vehicle	Zugangsregelung für Kraftfahrzeuge aller Art Durchfahrt verboten
Hgv	Zugangsbeschränkung für LKWs über 3,5 Tonnen
Psv	Straße für öffentliche Verkehrsmittel
Psv:lanes	Anzahl der Fahrbahnen für öffentliche Verkehrsmittel
Lit	Straßenlaterne
Traffic_sign	Verkehrszeichen
Zone:traffic	Geschlossene Ortschaft

Bdouble	Erlaubnis für EuroCombi LKWs
Bridge	Hinweis auf eine Brücke
Bridge_name	Brückenname
Tunnel	Hinweis auf einen Tunnel
Junction	Kreuzung
Lanes	Anzahl Fahrbahnen
Lanes:backward	Fahrstreifen, deren Fahrtrichtung in entgegengesetzter Richtung des OSM Weges liegen
Lanes:forward	Fahrstreifen, deren Fahrtrichtung in Richtung des OSM Weges liegt
Destination:lanes	Richtung einzelner Fahrspuren des OSM Weges
Destination:lanes:forward	Richtung aller Fahrspuren die in Richtung des selben OSM Weges führen
Destination:lanes:backward	Richtung aller Fahrspuren die in entgegengesetzter Richtung des selben OSM Weges führen
Turn	Folgerichtung eines Weges, Abbiegespuren
Turn:lanes	Gibt die Anzahl an Abbiegespuren in Richtung desselben OSM Weges an
Turn:lanes:backward	Gibt die Anzahl an entgegengesetzter Abbiegespuren desselben OSM Weges an
Turn:lanes:forward	Gibt die Anzahl an Abbiegespuren des selben OSM Weges an
Change:lanes:forward	Gibt an ob ein Spurenwechsel in der selben Richtung des OSM Weges möglich ist
Change:lanes:backward	Gibt an ob ein Spurwechsel in entgegengesetzter Richtung des OSM Weges möglich ist
Layer	Verschiedene Darstellungsschichten des Map Layers
Covered	Beschreibt die Verdeckung einer Fläche
Maxspeed	Maximalgeschwindigkeit
Source:maxspeed:wet	Maximalgeschwindigkeit bei Nässe
Source:maxspeed:backward	Maximalgeschwindigkeit der entgegengesetzten Richtung des selben OSM Weges
Source:maxspeed:forward	Maximalgeschwindigkeit in Richtung des selben OSM Weges

Maxspeed:forward	Maximalgeschwindigkeit in Richtung des selben OSM Weges
Maxweight	Maximalgewicht in Tonnen
Width	Breite eines Weges oder einer Straße
Incline	Steigung eines Weges oder einer Straße
Oneway	Einbahnstraße
Ref	Referenznummer der Einbahnstraße
Reg_name	Regulärer Name der Straße (Eigennamen)
Bicycle	Hinweis auf eine Fahrradverbindung
Bicycle_road	Fahrradstraße
Cycleway	Fahrradweg
Cycleway:both	Fahrradweg auf beiden Straßenseiten
Cycleway:both:surface	Fahrradweg auf beiden Straßenseiten mit speziellem Untergrund
Cycleway:left:surface	Fahrradweg auf der linken Straßenseite mit speziellem Untergrund
Cycleway:right	Fahrradweg auf der rechten Straßenseite
Foot	Zugangsbeschränkung für Fußgänger
Footway:left	Fußweg auf der linken Straßenseite
Footway:right	Fußweg auf der rechten Straßenseite
Footway:both	Fußweg auf beiden Straßenseiten
Footway:both:surface	Fußweg auf beiden Straßenseiten mit speziellem Untergrund
Footway:right:surface	Fußweg auf der rechten Straßenseite mit speziellem Untergrund
Footway:left:surface	Fußweg auf der linken Straßenseite mit speziellem Untergrund
Sidewalk	Bürgersteig
Sidewalk:both	Bürgersteig auf beiden Straßenseiten
Sidewalk:left	Bürgersteig auf der linken Straßenseite
Sidealk:right	Bürgersteig auf der rechten Straßenseite
Sidewalk:left:surface	Bürgersteig auf der linken Straßenseite mit speziellem Untergrund
Sidewalk:right:surface	Bürgersteig auf der rechten Straßenseite mit speziellem Untergrund
Sidewalk:both:surface	Bürgersteig auf beiden Straßenseite mit speziellem Untergrund
Parking:condition:both	Parkmöglichkeiten auf beiden Straßenseiten

Parking:condition:both:default	Parkmöglichkeiten auf beiden Straßenseiten mit Standardbelegwert
Parking:condition:both:time_interval	Parkmöglichkeit auf beiden Straßenseiten nur zu bestimmten Zeiten möglich
Segregated	Zeigt an, ob es sich um einen getrennten oder gemeinsamen Rad- und Fußweg handelt
Service	Gibt zusätzliche Informationen zu einer Straße an.
Surface	Gibt den Untergrund einer Straße an
Tracktype	Güteklasse eines Wirtschaftsweges
Trail_visibility	Beschreibt die Erkennbarkeit eines Weges
Area	Gibt eine Fläche oder ein Gebiet an
Addr:city	Name Stadt wie in postalischer Adresse angegeben
Addr:postcode	Postleitzahl wie in postalischer Adresse angegeben
Addr:street	Straßenname wie in postalischer Adresse angegeben
Is_in	Ortsangabe
Smoothness	Beschreibt die Oberläche von Wegen an
Amenity	Markierung nützlicher und wichtiger Einrichtungen
Ref	Referenznummer einer nützlichen Einrichtung

Um eine Beziehung zwischen "Ways" herzustellen, werden spezielle Relations benötigt. In diesem Schritt können individuelle Routen erstellt und mit nötigen Zusatzinformationen verknüpft werden. Der dazugehörige Stylesheet "Relations" übernimmt diese Aufgabe und überträgt die neu gesammelten Informationen in die gewünschte Darstellungsform. Die Identifizierung der Ways wird über dem Primärschlüssel Way-ID im "ref" Tag sichergestellt.

```
<xsl:template match="/">
    <xsl:for-each select="//relation">
        <xsl:value-of select="@id" />
        <xsl:text>;</xsl:text>
        <xsl:if test="tag[@k='TMC:cid_58:tabcd_1:Class']">
        <xsl:value-of select="tag[@k='TMC:cid_58:tabcd_1:Class']/
           @v[1]" />
        </xsl:if>
```

```
            <xsl:text>;</xsl:text>
            [...]
            <xsl:text>;</xsl:text>
            <xsl:for-each select="member">
                <xsl:value-of select="@ref" />
                <xsl:text>;</xsl:text>
                <xsl:value-of select="@role" />
                <xsl:text>;</xsl:text>
                <xsl:value-of select="@type" />
                <xsl:text>;</xsl:text>
            </xsl:for-each>
            <xsl:text>&#10;</xsl:text>
        </xsl:for-each>
</xsl:template>
```

Die Arbeitsweise des Stylesheets ähnelt der, dem bereits vorgestellten Stylesheet ”Ways”. Zunächst durchläuft die erste for-each Schleife alle Relation Tags und extrahiert diejenigen Elemente, die in der unten stehenden Tabelle 9 aufgeführt sind. Die zweite for-each Schleife betrachtet Member-Tags die über das Element ”ref” eine Verbindung zwischen Relations und Ways herstellen. Die Elemente ”role” und ”type” spezifizieren die Art der Verbindung.

Tabelle 9: Stylesheets Relations

Element	**Beschreibung**
TMC:cid_58:tabcd_1:Class	TMC-Navigation- Klasse
TMC:cid_58:tabcd_1:Direction	TMC-Navigation-Richtung
TMC:cid_58:tabcd_1:LCLversion	TMC-Navigation-Version
TMC:cid_58:tabcd_1:LocationCode	TMC-Navigation- Aktuelle Area
TMC:cid_58:tabcd_1:NextLocationCode	TMC-Navigation- Nächste Area
TMC:cid_58:tabcd_1:PrevLocationCode	TMC-Navigation- Vorherige Area
Admin_level	Gibt Landesgrenzen an
De:amtlicher_gemeindeschluessel	Zuweisung eines individuellen Schlüssels zu Gemeinden
De:reginalschluessel	Zuweisung eines individuellen Schlüssels an Regionen
Description	Kurze und verständliche Beschreibung
Description:en	Kurze und verständliche Beschreibung in Englisch
Boundary	Grenzen
Postal_code	Postleitzahl
Postal_code_level	Postleitzahl mit spezifizierte Verwaltungsart

Is_in:township	Spezialisiert den Ortsbezug – hier Stadteil
Building	Gebäude aller Art
Section	Gibt Einteilungen in Bereichsintervalle an
Line	Gibt die Linie einer Busverbindung an
Name	Gibt den Namen einer Busverbindung an
Ref	Legt eine Referenznummer der Busverbindung oder Route fest
From	Gibt den Startort einer Busverbindung an
To	Gibt den Zielort einer Busverbindung an
Via	Gibt eine Zwischenstation einer Busverbindung an
Destination	Zeigt eine Zugangsbeschränkung an
Service	Gibt zusätzliche Informationen zu einer Busverbindung an
Network	Beschreibt das Netzwerk oder den Verkehrsverbund zu dem die Route gehört
Operator	Name des Betreibers von z.B. Busverbindungen
Public_transport	Spezifiziert öffentliche Verkehrsmittel
Public_transprt :version	Gibt die Versionsnummer des Erfassungsschemas für öffentliche Verkehrsmittel an
Type	Gibt den Typ einer Relation an
Main_road	Gibt an ob es sich um eine Hauptverbindung/Hauptstraße handelt
Route	Zeigt eine spezifische Zuordnung zu einer Route an
Route_master	Fasst mehrere Relationen einer Linie des öffentlichen Verkehrs in einer zusammen
Detour	Gibt eine Umleitungsstrecke an
Restriction	Gibt Vorschriften und Beschränkungen zu Routen an
Except	Schließt bestimmte Routenteilbereiche von Relations aus
Site	Fasst Objekte zusammen die in direktem Kontakt zueinander stehen
Lanes	Gibt eine einzelne Spur einer Straße an

Lanes:extra	Spezifiziert die Art einer Spur
Lengths:left	Gibt die Länge eines linken -Teilweges zwischen Relationspunktes an
Lengths:right	Gibt die Länge eines rechten –Teilweges zwischen Relationspunktes an
Name:nds	Namensspezifizierung in plattdeutscher Sprache
Enforcement	Geräte, um Verkehrsverstöße festzustellen
Roundtrip	Spezifiziert die Verbindungsweise sodass der Start und Endpunkt einer Verbindung gleich sind
Distance	Distanz in Kilometern
Railway	Transportformen die auf Metallschienen basieren
Aeroway	Zeigt Flugrouten an
Waterway	Spezifiziert Wasserstraßen
Water	Wasserfläche
Natural	Landschaften
Landuse	Landnutzung, menschliche Benutzung von Ländereien
Amenity	Markierung nützlicher und wichtiger Einrichtungen
Interval	Spezifiziert ein Zeitintervall zur Nutzung nützlicher und wichtiger Einrichtungen
Leisure	Beschreibt Ortschaften, an dem Menschen Freizeitmöglichkeiten vorfinden
Sport	Zeigt die Sportart an, die an einem Element betrieben wird
Ref	Gibt die Way_ID eines Weges an
Role	Beschreibt die Rolle des betrachteten Weges
Type	Zeigt den Typ einer Relation an

```
// read xml path
File xmlSourceNodes = new File(xmlPath);
System.out.println("building new csv nodes...");
// create new instance of documentbuilderfactory
DocumentBuilderFactory factoryNodes =
   DocumentBuilderFactory.newInstance();
DocumentBuilder builderNodes = factoryNodes.
   newDocumentBuilder();
```

```
            // parse xml file
            Document documentNodes = builderNodes.parse(
                xmlSourceNodes);
            // read stylesheet for new csv file
            StreamSource stylesourceNodes = new StreamSource(
                stylesheetNodes);
            // create new instance for transformer
            Transformer transformerNodes = TransformerFactory.
                newInstance().newTransformer(stylesourceNodes);
            // create new domsource from existing xml source
            Source sourceNodes = new DOMSource(documentNodes);
            // stream result as output for new csv file
            Result outputTargetNodes = new StreamResult(new File(
                csvNodes));
            // use transformer to create the new csv file
            transformerNodes.transform(sourceNodes,
                outputTargetNodes);
            System.out.println("done writing csv nodes");
        } catch (ParserConfigurationException e) {
            e.printStackTrace();
        } catch (SAXException e) {
            e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        } catch (TransformerConfigurationException e) {
            e.printStackTrace();
        } catch (TransformerFactoryConfigurationError e) {
            e.printStackTrace();
        } catch (TransformerException e) {
            e.printStackTrace();
    }}}
```

Die vorgestellten Stylesheets übernehmen als Aufgabe die Bildung einer CSV Struktur und überführen alle Elemente aus dem XML Source der OSM Datei in das gewünschte CSV Format. Im nächsten Schritt wird eine Instanz der „DocumentBuilderFactory“ gebildet. Mit Hilfe eines neuen Documents sowie des in dem javax.xml.* Paket zur Verfügung stehenden XML Parsers, können alle Elemente der XML OSM Datei in eine Baumstruktur überführt werden. Zusätzlich werden die erstellten Stylesheets mittels eines StreamSource geladen und der Transformerinstanz als Input übergeben. Dadurch bindet der Transformer die Struktur der Stylesheets an die erstellte Instanz. Die modifizierte Methode transform() aus der TranformerFactory liest abschließend als Input die im DOMSource übergebene Baumstruktur der XML OSM Datei ein und schreibt dies unter Berücksichtigung des neuen Formats in die neu erstellte CSV Datei.

7.10 Vorbereitung der Zählspuldaten

Wie in Abschnitt 6.6 beschrieben wurde, hat sich die Projektgruppe dazu entschieden nur die Daten der Zählspulen für Autos zu verwenden. Mit Hinblick auf diese Entscheidung gab

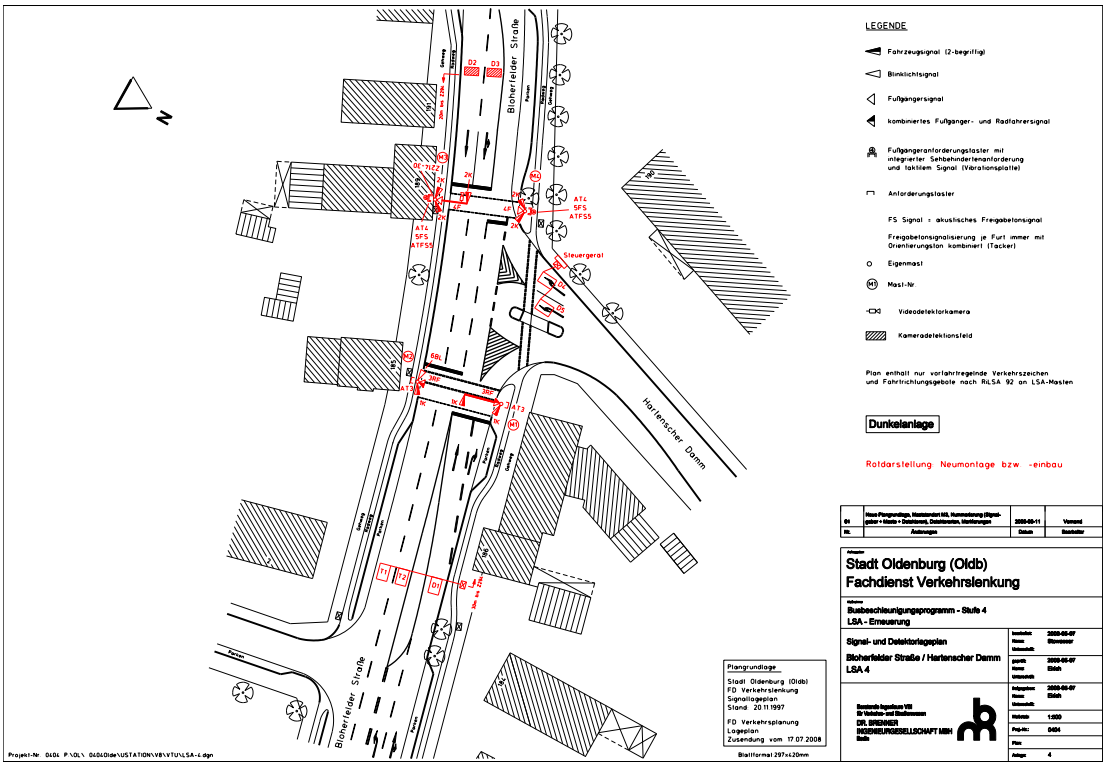


Abbildung 29: Beispiel für einen Lageplan mit den Zählspulen

es zwei Datensätze, die für das weitere Vorgehen vorbereitet werden mussten. Zum einen lagen der Projektgruppe die Lagepläne der Zählspulen vor und zum anderen gab es mehrere Tabellen mit den jeweils erfassten Daten für die einzelnen Zählspulen. Im Folgenden wird darauf eingegangen, wie diese beiden Datenquellen für eine spätere Analyse aufbereitet wurden.

7.10.1 Vorbereitung der Lagepläne

Zur Wiederholung ist in Abbildung 29 ein Beispiel für einen Lageplan gegeben. Darauf sind unter anderem die Zählspulen und die Straßen mit Namen dargestellt.

Um die Position der Zählspulen allerdings in dem Portal darstellen zu können, wurden die jeweiligen Koordinaten benötigt. Da die Lagepläne als PDF-Datein vorlagen und somit sehr schwer automatisiert ausgewertet werden konnten, wurden die Koordinaten händisch bestimmt. Dabei wurde folgendes Schema für die Erfassung der Zählspulen verwendet.

```
<node id="4000014000" lat="53.12736" lon="8.2363">
  <tag k="name" v="Bremer Heerstrasse"/>
  <tag k="type" v="counting_loop"/>
  <tag k="counting_loop" v="VSA191_D1_T1"/>
```

```
</node>

<relation id="4598000">
  <member type="node" ref="2870915348" role="from"/>
  <member type="node" ref="74048675" role="to"/>
  <member type="node" ref="4000014000" role="via"/>
  <tag k="type" v="counting_loop"/>
  <tag k="counting_loop" v="VSA191_D1_T1"/>
  <tag k="counting_lanes" v="1"/>
  <tag k="counting_lanes_max" v="2"/>
  <tag k="counting_lanes_position" v="1"/>
  <tag k="turn" v="011"/>
</relation>
```

Für jede Zählspule wurden zwei Datensets erfasst. Zum einen die Zählspule selber, als *node* bezeichnet und eine *relation* die weitere Informationen über die Zählspule enthält.

Der *node* wurde dabei durch folgende Attribute spezifiziert:

- **id:** Die *id* ist eine fortlaufende Zahl um die Zählspulen später eindeutig in der Datenbank wiederfinden zu können.
- **lat, lon:** Diese beiden Werte geben den Breiten-(engl. latitude, lat) und den Längengrad (engl. longitude, lon) und somit die Koordinaten der Zählspule wieder. Diese Angaben werden benötigt um die Zählspule auf der Karte an der richtigen Position darzustellen.
- **name:** Gibt den Namen der Straße wieder, auf welcher die Zählspule zu verordnen ist.
- **type:** Gibt den Typ der Schaltanlage wieder. Dieses Attribut wurde eingeführt um bei einer Erweiterung der Datenbasis um beispielsweise die Ampeltaster zwischen den einzelnen Schaltanlagen unterscheiden zu können.
- **counting_loop:** Dieses Attribut referenziert den Namen der Zählspule über welchen die erfassten Werte aus der Wertetabelle den einzelnen Zählspulen zugeordnet werden können.

Parallel zu dem *node* wurde für jede Zählspule eine *relation* mit folgenden Attributen erfasst:

- **id:** Auch der *relation* wurde eine fortlaufende Nummer zugewiesen um sie später in der Datenbank eindeutig identifizieren zu können.
- **node ... from:** Wie in Abschnitt 7.9 beschrieben, werden die Straßen in *Open Street Map* durch Knoten beschrieben. Dieses Attribut gibt den letzten OSM Knoten an, der in Fahrtrichtung vor der zu betrachtenden Zählspule liegt. In Verbindung mit den nächsten beiden Attributen soll dadurch die Richtung, in welcher die Zählspule passiert wird, abgebildet werden.

- **node ... to:** Dieses Attribut gibt den OSM Knoten an, der in Fahrtrichtung nach der zu betrachtenden Zählspule kommt.
- **node ... via:** Dieses Attribut gibt den OSM Knoten an, der am nächsten an der zu betrachtenden Zählspule liegt.
- **type:** Gibt den Typ der Schaltanlage wieder.
- **counting_loop:** Gibt den Namen der Zählspule wieder, unter welchem die Zählspule in der Wertetabelle referenziert wird.
- **counting_lanes:** Dieses Attribut gibt an, wie viele Straßenspuren die angegebene Zählspule erfasst. Dieses und die folgenden drei Attribute sind für eine Erweiterung ausgelegt in welcher aus den Zählspulen beispielsweise ein Netz erstellt werden könnte.
- **counting_lanes_max:** Dieses Attribut gibt an, wie viele Straßenspuren insgesamt in Fahrtrichtung um die Zählspule herum vorhanden sind.
- **counting_lanes_position:** Wenn um die Zählspule herum mehrere Straßenspuren vorhanden sind muss festgehalten werden, auf welcher Spur sich die zu betrachtende Zählspule befindet. Die Nummerierung findet dabei in Fahrtrichtung vom Straßenrand mit der Nummer 1 an. Die in Fahrtrichtung der Straße am weitest rechts liegende Spur wird also mit der 1 beziffert.
- **turn:** Da die Zählspulen meistens vor Ampeln positioniert sind, gibt dieses Attribut an, welche Möglichkeiten der Weiterfahrt einem Autofahrer auf der Straßenspur der zu betrachtenden Zählspule vorliegen. *100* bedeutet, dass ein Autofahrer nur nach links abbiegen kann. *010* heißt, dass der Autofahrer nur geradeaus weiterfahren kann und *001* bedeutet, der Autofahrer muss rechts abbiegen. Diese einzelnen Möglichkeiten können weiterhin kombiniert werden, sodass der Wert *011* bedeutet, dass der Autofahrer entweder geradeaus weiterfahren oder rechts abbiegen kann.

Anhand der erfassten Koordinaten konnten die Zählspulen im späteren Verlauf in die Darstellung aufgenommen werden.

7.10.2 Vorbereitung der Messwerte

Der zweite Datensatz der für das weitere Vorgehen vorbereitet werden musste war die Tabelle mit den jeweils gemessenen Werten für die Zählspulen. Wie in Abschnitt 6.6 beschrieben wurde, wurden uns die Daten in 12 verschiedenen Tabellen zur Verfügung gestellt. Um in der späteren Analyse nicht auf 12 verschiedene Tabellen zugreifen zu müssen, sollten die Tabellen zu einer Tabelle zusammengeführt werden. Bei der Vorbereitung dazu fiel auf, dass die Zeitstempel der einzelnen Tabellen nicht übereinstimmten. Zum einen gab es größere zeitliche Abstände zwischen zwei aufeinanderfolgenden Werten (vgl. Abschnitt 6.6), die in einer Tabelle vorhanden waren und in einer anderen nicht und zum anderen

kam es vor, dass die Erfassung der Daten nach einer größeren Pause um 30 oder 60 Sekunden zum üblichen 90 Sekunden Takt verschoben war. Andere Verschiebungen als 30 oder 60 Sekunden konnten nicht gefunden werden, die Werte wurden also immer zu xx:xx:00 oder xx:xx:30 Stunden erfasst. Um diese Schwierigkeit zu beheben wurde beschlossen in der Gesamttabelle eine Spalte *TIMESTAMP* anzulegen, welche in 30 Sekunden Schritten den gesamten Monat März abbildet. Die Werte aus den Einzeltabellen wurden dann entsprechend dem richtigen Zeitstempel eingefügt.

Spalten: TIMESTAMP WOCHENTAG GRUPPE Zählspulen...

8 Modellierung

Im Prozessschritt *Datenverständnis*, vgl. Kapitel **6**, wurden bereits benötigte Daten erhoben und beschrieben. Im Prozessschritt *Datenvorbereitung*, vgl. Kapitel **7**, wurden diese Rohdaten weiter vorbereitet indem Fehler behoben und die Daten in ein Format gebracht wurden, welches von späteren Analyseverfahren akzeptiert wird. In diesem Prozessschritt, der *Modellierung*, geht es jetzt darum die Daten hinsichtlich einer konkreten Problemstellung zu analysieren um neue Informationen zu generieren.

Der erste Arbeitsschritt der Modellierung, die *Auswahl der Modellbildungsverfahren*, wird in Abschnitt 8.1 näher beleuchtet. Dabei geht es darum geeignete Verfahren zu bestimmen, mit welchen die vorhandenen Daten analysiert werden können um die konkrete Problemstellung zu behandeln. Um das ausgewählte Verfahren später evaluieren zu können, gilt es im zweiten Arbeitsschritt, dem *Generieren eines Test-Designs*, unter anderem geeignete Kriterien zu definieren um das Modell auf seine Güte zu testen. Auf diesen Arbeitsschritt wird in Abschnitt 8.2 näher eingegangen. Im dritten Arbeitsschritt, dem *Erstellen der Modelle*, werden dann die einzelnen Modelle, die vorher ausgewählt wurden, auf die Daten angewandt um sichtbare Ergebnisse zu schaffen. In Abschnitt 8.3 wird nochmal näher auf diesen Arbeitsschritt eingegangen. Der vierte und letzte Arbeitsschritt, die *Bewertung des Modells*, auf welche in Abschnitt 8.4 näher eingegangen wird, beschäftigt sich damit, die Ergebnisse der einzelnen Modelle untereinander oder hinsichtlich der vorher definierten Gütekriterien zu vergleichen.

Da während des Prozessschrittes *Modellierung* verschiedene Analyseverfahren ausprobiert werden, unter Umständen auch welche die während der vorherigen Prozessschritte nicht bedacht wurden, kann es vorkommen, dass die Daten nicht passend vorbereitet wurden. In einem solchen Fall kann es nötig sein wieder in den Prozessschritt *Datenvorbereitung* zu wechseln um die Daten entsprechend für ein anderes Modell vorzubereiten. Wenn während der Modellierung auffällt, dass für ein bestimmtes Analyseverfahren benötigte Daten nicht, oder nicht ausreichend vorhanden sind, kann es zum Teil nötig sein in den Prozessschritt *Datenverständnis* zurück zu wechseln.

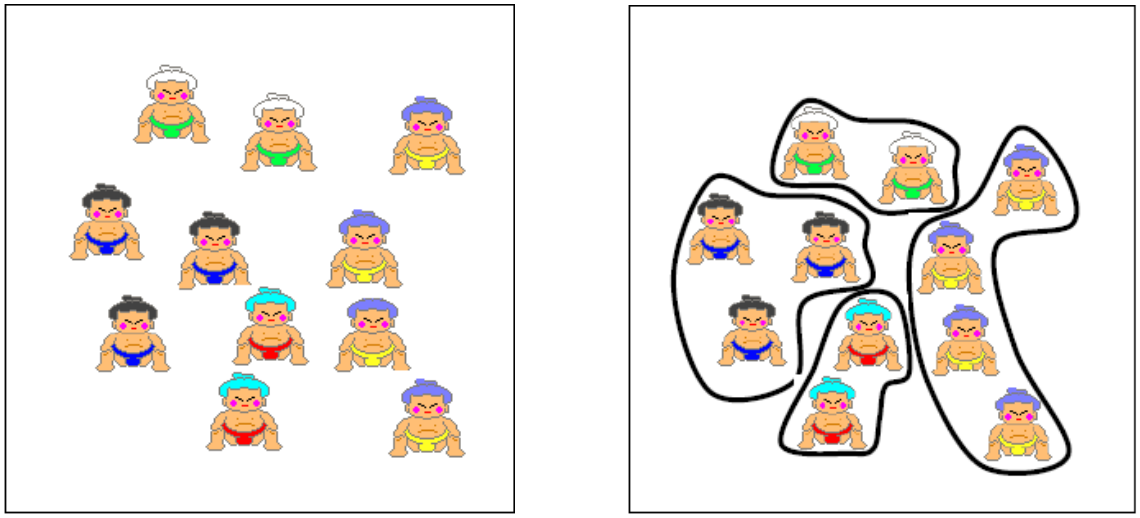
8.1 Auswahl der Modellbildungsverfahren

Wie in der Einleitung bereits beschrieben geht es in dem Prozessschritt *Modellierung* darum die vorbereiteten Daten hinsichtlich einer konkreten Fragestellung zu analysieren. Der erste Schritt dahin ist die *Auswahl der Modellbildungsverfahren*. Die Auswahl der richtigen Verfahren ist unter anderem deswegen von großer Bedeutung, da nicht jedes Verfahren für jede Fragestellung verwendet werden kann. Wenn die Aufgabenstellung beispielsweise vorsieht für ein Unternehmen die Kundenbasis in Untergruppen aufzuteilen, die jeweils ähnliche Attributausprägungen (Alter, Geschlecht, Einkommen, etc.) haben, dann kann eine Clusteranalyse ein geeignetes Verfahren sein. Wenn die selbe Problemstellung allerdings versucht werden soll mit einer Regressionsanalyse beantwortet zu werden, könnte dies nahezu unmöglich sein, da Regressionsanalysen eher verwendet werden können, wenn die Problemstellung eine Vorhersage erfordert. Was Clusteranalysen und Regressionsanalysen sind, wird in den Abschnitten 8.1.1 und 8.1.2 näher erläutert. Neben diesen beiden

Verfahren gibt es aber noch viele weitere wie beispielsweise künstliche neuronale Netze, regelbasierte Methoden oder Entscheidungsbäume einsetzen. Neben den technischen Anforderungen an das Verfahren kann es auch noch „politische Anforderungen“ geben. Beispielsweise kann dem Unternehmen nur ein bestimmtes Tool zur Analyse zur Verfügung stehen, welches nur eine begrenzte Anzahl an Analyseverfahren bereitstellt. Aufgrund dieser Aspekte gilt es dann das bestmögliche Verfahren für die konkrete Fragestellung zu bestimmen.

8.1.1 Cluster Analyse

Das Ziel der Cluster Analyse ist es eine Menge von Datenobjekte in Gruppen (engl. Cluster) zu unterteilen, wobei die Mitglieder einer Clusters ähnliche Eigenschaften haben. Ein einfaches Beispiel ist in Abbildung 30 dargestellt. Auf der linken Seite sind die Datenobjekte ohne Zugehörigkeit zu einem Cluster dargestellt. In diesem Fall ist ein Datenobjekt einer der Sumoringer, welches im wesentlichen durch die Attribute *Haarfarbe* und *Hosenfarbe* beschrieben wird. Aufgrund dieser beiden Attribute können die Datenobjekte in vier Cluster eingeteilt werden, wobei jedes Cluster die Sumoringer mit der gleichen Haar- und Hosenfarbe beinhaltet.



(a) Sumo-Ringer ohne Zugehörigkeit (b) Sumo-Ringer mit Clusterzugehörigkeit

Abbildung 30: Einfaches Beispiel für Cluster-Analyse. Sumo-Ringer mit gleicher Haar- und Hosenfarbe gehören zu einem Cluster. Quelle:

In diesem Beispiel wird das Datenobjekt durch zwei Attribute beschrieben, die zur Bestimmung der Cluster verwendet werden. Im allgemeinen ist es aber auch möglich und üblich mehrere Attribute für ein Datenobjekt zu verwenden, die als Vektor in den Analysealgorithmus eingespeist werden. Wie in dem Beispiel muss dann anhand der Attribute bestimmt werden, ob zwei Datenobjekte zueinander ähnlich sind um entsprechend zu-

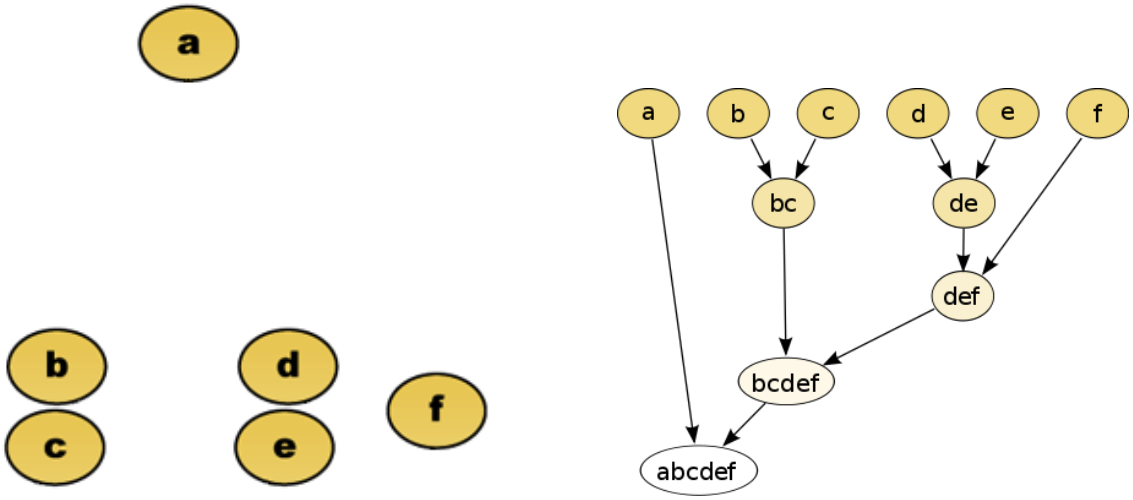
geordnet werden zu können. Um die Ähnlichkeit zu bestimmen ist ein weit verbreitetes Verfahren die L_2-Norm. Bei der L_2-Norm wird im wesentlichen für jedes Attribut die quadratische Differenz zwischen den beiden Ausprägungen der Datenobjekte berechnet. Die Einzeldifferenzen werden dann summiert und aus der Summe wird die Wurzel gezogen. Der Wert, der dabei berechnet wird, ist dann eine numerische Repräsentation der Nähe zwischen zwei Datenobjekten. Da für die Berechnung numerische Werte verlangt werden wird vorausgesetzt, dass auch alle verwendeten Attribute numerisch repräsentiert werden. Dies ist ein Beispiel dafür, dass die Datenvorbereitung ein wichtiger Schritt ist und auf das verwendete Verfahren abgestimmt sein muss. Wird dieser Schritt für jede Paarung der Datenobjekte ausgeführt, kann daraus eine Distanzmatrix[2] berechnet werden, die dann im späteren Verlauf für die Bestimmung der Cluster verwendet werden kann.

Um jetzt Cluster zu erstellen sind zwei häufig verwendete Verfahren das *hierarchische Verfahren* oder das *partitionierende Verfahren*. Bei den hierarchischen Verfahren gibt es im wesentlichen zwei Möglichkeiten, das *Top-down* Verfahren und das *Bottom-Up* Verfahren. Zur Veranschaulichung dient Abbildung 31. Auf der linken Seite in Abbildung 31(a) sind die einzelnen Datenobjekte so dargestellt, dass sie die berechnete Nähe zu den jeweils anderen Datenobjekten repräsentieren. Die Objekte *b* und *c* haben beispielsweise eine geringe Distanz und werden entsprechend nah zueinander dargestellt. Auf der rechten Seite in Abbildung 31(b) wird dargestellt, wie die einzelnen Datenobjekte zu Clustern zusammengefasst werden. Das *Bottom-Up* Verfahren betrachtet zunächst jedes Datenobjekt als eigenes Cluster. Im Laufe des Verfahrens werden dann Cluster zusammengefasst, die die geringste Distanz zueinander haben. In dem Beispiel werden im ersten Schritt die Datenobjekte *b* und *c* zu einem Cluster zusammengefasst, ebenso wie *d* und *e*. Wenn das Verfahren lange genug läuft, bildet sich am Ende ein einziges Cluster mit allen Datenobjekten, in dem Beispiel das Cluster *abcdef*. Das *Top-Down* Verfahren setzt hingegen bei diesem Cluster an, welches alle Datenobjekte umfasst. Im Laufe des Verfahrens werden die Cluster dann in kleinere Cluster aufgespaltet, die jeweils die größte Distanz zueinander haben. In dem Beispiel würde das Gesamtcluster *abcdef* im ersten Schritt in die Cluster *a* und *bcdef* aufgeteilt werden.

Ein Nachteil von hierarchischen Verfahren ist, dass Datenobjekte, die einmal in einem Cluster zusammengefasst werden, nicht mehr getrennt werden können. Es ist also nicht möglich einzelne Datenobjekte im Laufe des Verfahrens einem anderen Cluster zuzuordnen, obwohl die Distanz zu diesem anderen Cluster möglicherweise geringer ist, als zu dem aktuell Zugeordneten. Auf der anderen Seite sind bei hierarchischen Verfahren die Anzahl der Cluster nicht vorgegeben. Das bedeutet, wenn festgestellt wird, dass die Aufteilung nicht fein oder grob genug ist, kann eine Ebene höher oder tiefer im Hierarchiebaum gesucht werden.

Anders verhält es sich bei *partitionierenden Verfahren*, bei welchen die Anzahl der Cluster (k) von vornherein festgelegt werden muss. Auf der anderen Seite ist es möglich das Datenobjekte im Laufe des Algorithmus das zugehörige Cluster ändern. Da dies im schlimmsten Fall unendlich oft passieren kann, muss beim *partitionierenden Verfahren* teil-

[2] In einer Distanzmatrix wird für jedes Datenobjekt x die Distanz zu jedem anderen Datenobjekt y ($y \neq x$) gespeichert.



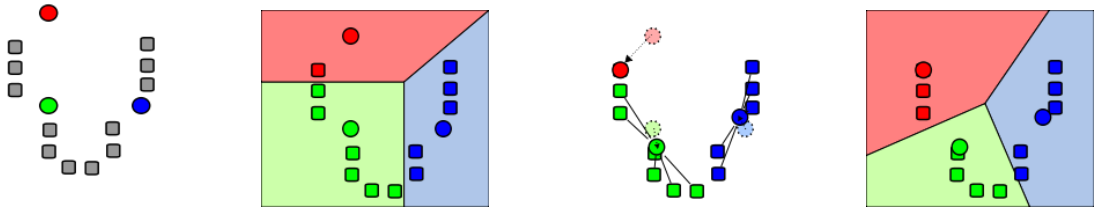
(a) Die einzelnen Datenobjekte. Die Distanz zwischen bspw. *b* und *c* ist sehr gering und wird entsprechend dargestellt

(b) Darstellung der hierarchischen Gruppierung der Datenobjekte

Abbildung 31: Einfaches Beispiel für ein hierarchisches Clustern von Datenobjekten. **[Eas00]**

weise die Anzahl der Iterationen als Abbruchbedingung des Algorithmus gegeben werden. Ein weit verbreitetes Verfahren innerhalb der Partitionierenden Verfahren ist der *k-Means-Algorithmus*. Vereinfacht funktionieren k-Means-Algorithmen indem sie zunächst zufällig k Punkte aus dem gesamten Datenraum auswählen und diese als Zentrum für die Cluster annehmen. In Abbildung 32(a) sollen drei Cluster gebildet werden. Die zufällig ausgewählten Datenpunkte werden als Kreis dargestellt und die Datenobjekte (Kästen) sind zunächst keinem Cluster zugeordnet. Im zweiten Schritte werden die Datenobjekte entsprechend ihrer Nähe, welche auch hier durch die L_2-Norm berechnet werden kann, einem der drei Cluster zugeordnet. Dieser Vorgang ist in Abbildung 32(b) dargestellt. Entsprechend den zugeordneten Datenobjekten wird der Schwerpunkt des Clusters verschoben, dargestellt in Abbildung 32(c). Diese beiden Schritte, das Zuordnen der Datenobjekte und das Verschieben der Schwerpunkte, werden iterativ so lange fortgeführt, bis die Datenobjekte nicht mehr neu zugeordnet werden oder bis die vorgegebene Anzahl an Iterationen überschritten wird. Das Ergebnis ist eine Zuordnung der Datenobjekte zu einem der Cluster. Für das Beispiel ist das Endergebnis in Abbildung 32(d) dargestellt.

Wichtig bei der Clusteranalyse ist, dass zum einen die Ergebniscluster keine Interpretation der Gruppen beinhalten und zum anderen, dass durch die Analyse Datenobjekte zu Clustern zusammengeführt werden, die von vornherein nicht erwartet wurden. So kann eine Clusteranalyse auf eine Menge von Fahrzeugen beispielsweise ergeben, dass ein gefundenes Cluster Fahrzeuge beinhaltet, die ein Analyst als LKWs bezeichnen würde. Diese Interpretation aber wurde nicht von der Clusteranalyse vorgegeben und ist zudem keine neue Erkenntnis. Auf der anderen Seite kann eine Clusteranalyse, wieder auf eine Menge



(a) Es werden zufällig drei Datenpunkte (Kreise) ausgewählt, die als erster Schwerpunkt genutzt werden

(b) Aufgrund der Schwerpunkte wird der Datenraum in drei Bereiche aufgeteilt

(c) Die Schwerpunkte werden aufgrund der zugeordneten Datenobjekte neu justiert

(d) Die Aufteilung der Datenobjekte wird mit den neuen Schwerpunkten neu berechnet

Abbildung 32: Beispiel für Gruppierung durch einen k-Means-Algorithmus. [**Eas00**]

von Fahrzeugen angewendet, ergeben, dass es eine Gruppe von Fahrzeugen gibt, die „rot“ sind. Es zeigt sich also, dass eine Clusteranalyse sinnvoll ist um innerhalb einer großen Datenmenge zusammenhängende Objekte zu finden, allerdings muss stets berücksichtigt werden, dass die Berechnung rein mathematisch ist und die Interpretation der Cluster mit Hinblick auf die konkrete Problemstellung zumeist händisch durchgeführt werden muss. Bei der Interpretation der Cluster kann es dann unter Umständen dazu kommen, dass keine neuen Gruppierungen gefunden wurden, die nicht bereits vorher bekannt waren, wie im Fall der Gruppe LKWs. Im allgemeinen ist es deswegen sinnvoll mehrere Clusteranalysen, wenn gegeben auch mit mehreren verschiedenen Algorithmen, mit verschiedenen Attributen die betrachtet werden sollen, durchzuführen.

8.1.2 Regression

Die Regressionsanalyse die dazu die Werte bestimmter Datenobjekte durch eine Funktion bestmöglich zu approximieren. Auf diese Weise lässt sich ein gerichteter Zusammenhang zwischen den betrachteten Variablen erforschen. Die Regressionsanalyse zielt auf das Generieren einer Funktion, bei der es in der Summe der Fehlerquadrate (Summe der Abweichungen zwischen der durch die Funktion berechneten und den tatsächlichen Werten) zur kleinstmöglichen Abweichung kommt. Durch weitere statistische Kennzahlen (f-Test, t-Test,...) kann zudem getestet werden, wie hoch die Signifikanz zwischen den unabhängigen und abhängigen Variablen ist [Sta15].

8.2 Generieren eines Test-Designs

Nachdem ein (oder mehrere) Modellbildungsverfahren ausgewählt wurden, gilt es im zweiten Arbeitsschritt ein Test-Design für das (oder die) entsprechende(n) Verfahren zu erstellen. Dieses Test-Design wird dann später in Abschnitt 8.4 dazu genutzt, das implementierte Verfahren auf seine Qualität zu testen. Wenn als Modell beispielsweise ein Klassifikations-

verfahren[3] verwendet wird, kann ein mögliches Qualitätskriterium die prozentuale Anzahl der richtigen Zuordnungen sein. Auch hier gilt es wieder die Qualitätskriterien abhängig von der konkreten Fragestellung zu definieren. Um das Modell anhand der erstellten Kriterien später testen zu können ist es weiterhin erforderlich, dass die gesamte Menge der Datenobjekte in zwei Gruppen aufgeteilt wird, die Trainingsdaten und die Testdaten. Die Trainingsdaten werden verwendet um das Modell zu trainieren, also beispielsweise über eine Clusteranalyse neue Gruppen zu bestimmen. Bei der Auswahl der Trainingsdaten sind zwei Punkte zu beachten. Zum einen sollten die Trainingsdaten möglichst den kompletten Datenraum abdecken, zum anderen aber sollte die Auswahl weitestgehend zufällig sein, damit nicht bestimmte Bereiche des Datenraums stärker ausgeprägt sind, als andere. Weiterhin stellt auch die Quantität der Trainingsdaten eine Schwierigkeit dar. Wie bei der Auswahl sollten auf der einen Seite genügend Trainingsdaten vorhanden sein um den gesamten Datenraum abdecken zu könne, auf der anderen Seite kann es durch zu viele Daten dazu kommen, dass sich das Modell zu stark an den Trainingsdaten orientiert und nicht mehr von diesen abstrahieren kann. Für dieses Problem wird teilweise auch der Begriff *overfitting* verwendet.

Die Testdaten werden dann entsprechend dazu verwendet das trainierte Modell zu testen, ob es also mit unbekannten Daten zum einen allgemein umgehen kann und zum anderen die vorher definierten Qualitätskriterien auch mit unbekannten Daten erfüllt. Um hierbei eine neutrale Bewertung zu gewährleisten sollten die Test- und Trainingsdaten disjunkt sein.

Neben den Qualitätskriterien, die später verwendet werden um die Gültigkeit des Modells zu testen und der Aufteilung der gesamten Datenmenge in die beiden Gruppen Test- und Trainingsdaten müssen für einige Analyseverfahren weitere Parameter festgelegt werden, wie beispielsweise die Anzahl der Iterationen (vgl. Abschnitt 8.1.1), sodass am Ende dieses Arbeitsschrittes ein Plan vorliegt, wie und unter welchen Bedingungen das zu erstellende Modell zu testen und bewerten ist.

8.3 Erstellen der Modelle

In dem dritten Arbeitsschritt, dem *Erstellen der Modelle*, geht es jetzt darum die in Abschnitt 8.1 ausgewählten Modelle mit den in Abschnitt 8.2 definierten Testdaten auszuführen. Da es bei vielen der Modelle Einstellungsparameter gibt, empfiehlt es sich das gleiche Modell mit verschiedenen Einstellungen auszuprobieren und jeweils die Einstellungen und das Ergebnis festzuhalten. Alleine dadurch, dass bei beispielsweise einer Clusteranalyse (vgl. Abschnitt 8.1.1) die initialen Schwerpunkte zufällig ausgewählt wurden kann das Ergebnis stärker oder weniger stark variieren, auch wenn die Einstellungen gleich behalten wurden. Sollten die Ergebnisse dann zu stark unterschiedlich sein ist es unter Umständen nötig einen Schritt zurück zu gehen und Parameter anzupassen, wie beispielsweise die Anzahl der Iterationen bei der Clusteranalyse.

[3] Bei Klassifikationsverfahren sind die Cluster bereits vorgegeben und es gilt die Datenobjekte den richtigen Clustern zuzuordnen.

Das Ergebnis dieses Arbeitsschrittes sind zum einen Modelle, die aus den Trainingsdaten erstellt wurden und zum anderen die Aufzeichnungen darüber, welche Einstellungen jeweils verwendet wurden. Ein weiterer wichtiger Teil in den Aufzeichnungen ist die Begründung für die einzelnen Parameter um später nachvollziehen zu können, wie ein bestimmtes Modell entstanden ist, falls es im späteren Verlauf Probleme geben sollte.

8.4 Bewerten des Modells

Der vierte und letzte Arbeitsschritt innerhalb der *Modellierung* ist das *Bewerten des Modells*. Dabei werden die in Abschnitt 8.3 erstellten Modelle hinsichtlich der in Abschnitt 8.2 definierten Qualitätskriterien mit den ebenfalls in Abschnitt 8.2 definierten Testdaten überprüft. Da meistens mehrere Modelle erstellt wurden, mit entweder unterschiedlichen Einstellungen oder gänzlich unterschiedlichen Algorithmen, ist auch hier ein wichtiges Ergebnis des Arbeitsschrittes die Dokumentation. Weiterhin sollten die einzelnen Modelle miteinander verglichen werden, welches Modell die Qualitätskriterien am besten erfüllt. Aber nicht nur der Vergleich der Modelle untereinander ist zu dokumentieren, sondern auch wie die einzelnen Modelle interpretiert werden können. Wenn ein Modell bei einer Clusteranalyse alle Testdaten richtig zuordnet, die Klassen, die vorher gefunden wurden allerdings nicht sinnvoll interpretiert werden können oder mit Hinblick auf die konkrete Problemstellung unsinnig sind, dann sollte auch ein hohes Maß an Qualität nicht ausschlaggebend sein. Neben der Interpretation gilt es auch die Plausibilität des Modells zu überprüfen. Anhand dieser verschiedenen Bewertungsgrundlagen werden dann die für die konkrete Problemstellung am besten geeigneten Modelle ausgewählt und näher betrachtet. Diese nähere Betrachtung könnte beispielsweise von Experten durchgeführt werden, oder die Modelle werden dahingehend untersucht ob die gefundenen Informationen neu und für die konkrete Problemstellung hilfreich sind. Sollte in diesem Arbeitsschritt herausgefunden werden, dass keines der erstellten Modelle den Qualitätskriterien entspricht oder für das Problem nicht hilfreich ist, kann dies bedeuten in einen früheren Arbeitsschritt zu wechseln. Beispielsweise könnten die Parameter für die Modelle neu bestimmt oder die Qualitätskriterien angepasst werden. Teilweise ist es auch nötig ein komplett anderes Modell neu zu entwerfen und zu testen.

8.5 Prognose

Eine Kernaufgabe der Projektgruppe ist es ein Modell zu entwickeln, welches aufgrund von vorhandenen Daten einen Wert in der Zukunft prognostiziert. Während der Recherche, welche Verfahren es gibt um Werte zu prognostizieren, hat es sich als sinnvoll erwiesen die Prognose in zwei Bereiche aufzuteilen. Zum einen soll es eine *kurzfristige Prognose* geben, die Werte für 15, 30 und 60 Minuten in der Zukunft berechnet. Welche Algorithmen dabei verwendet werden wird in Abschnitt 8.5.1 näher erklärt. Zum anderen ist es möglich durch die *langfristige Prognose* einen beliebigen Tag in der Zukunft zu wählen und sich für diesen die Verkehrssituation anzeigen zu lassen. Auf diesen zweiten Bereich wird in Abschnitt 8.5.2 näher eingegangen.

8.5.1 Kurzfristige Prognose

Für die kurzfristige Prognose haben sich während der Recherche zwei Verfahren als sinnvoll erwiesen. Die Grundidee für das erste Verfahren ist es, für den angegebenen Wert in der Zukunft, beispielsweise 12:15 Uhr (es ist aktuell 12:00 Uhr), die Werte von den letzten 15 Tagen zu verwenden die für die gleiche Zeit, also 12:15 Uhr, erfasst wurden. Diese sehr einfache Betrachtung hat sich aber als zu ungenau erwiesen, da das Verkehrsaufkommen an den Tagen der Woche unterschiedlich ist. An einem Sonntag fahren beispielsweise weniger Autos, als an einem Montag, weil die Leute nicht zur Arbeit müssen. Aus ähnlichen Gründen ist nicht nur das grundsätzliche Verkehrsaufkommen unterschiedlich, sondern auch der zeitliche Verlauf des Verkehrsaufkommen. Während an Wochentagen ein erhöhtes Verkehrsaufkommen morgens und nachmittags erfasst wird, wird an Sonntagen eher Nachmittags/Abends ein höheres Aufkommen beobachtet. Daraus ergab sich die Notwendigkeit, die Wochentage in Gruppen einzuteilen, sodass für die Prognose nur noch Tage verwendet werden würden, die grundsätzlich ähnlich zu dem Tag sind für welchen die Prognose ausgeführt werden soll. Eine Clusteranalyse wie sie in Abschnitt 8.1.1 beschrieben wurde, hätte für den Umfang von 31 Tagen, der der Projektgruppe zur Verfügung stand, keine hinreichend gute Aufteilung ergeben, deswegen wurde eine logische Aufteilung vorgenommen die vier Gruppen vorsieht:

- Gruppe 1: Montag, Dienstag, Mittwoch, Donnerstag

  Diese Gruppe bildet den klassischen Arbeitstag ab.

- Gruppe 2: Freitag

  An Freitagen fahren Arbeitnehmer eventuell früher nach Hause. Außerdem verlassen einige Leute über das Wochenende die Stadt.

- Gruppe 3: Samstag

  An Samstagen haben die Geschäfte noch geöffnet, sodass Leute eventuell in die Stadt fahren um Besorgungen zu tätigen. Außerdem werden Sport- oder andere Veranstaltungen öfter am Wochenende ausgetragen.

- Gruppe 4: Sonntag

  An Sonntagen haben die Geschäfte nicht mehr geöffnet. Aber es werden dennoch Sport- oder andere Veranstaltungen ausgetragen.

Diese Einschätzung wurde auch durch eine Arbeit von Roland Chrobok **[Chr05]** gestützt, der in seiner Arbeit den Verkehr auf Autobahnen untersucht und aufgrund eines wesentlich größeren Datensatzes eine ähnliche Zuordnung vorgenommen hat.

In seiner Arbeit wurden darüber hinaus Feiertage wie Ostersonntag und auch verschiedene Zeiten im Jahr, wie Ferien, Sommer und weitere betrachtet. Eine solch präzise Betrachtung war unserer Projektgruppe aufgrund der Datenbasis nicht möglich. Im Ausblick, in Kapitel *11*, wird auf dieses Thema aber nochmal näher eingegangen.

Durch die Aufteilung der Wochentage in die einzelnen Gruppen konnte die Prognose präzisiert werden. Die einzelnen Schritte der Prozedur, welche dieses Vorgehen in dem SAP HANA System implementiert wird im Folgenden erläutert.

Wenn die Prozedur aufgerufen wird, wird ihr der aktuelle Zeitstempel (`now_origin`) übergeben. Da die Daten nur in 30 Sekunden Abständen vorliegen ist die erste Aufgabe von dem übergebenden Zeitstempel aus den letzten vorhandenen Zeitstempel in der Datenbank zu finden. Dafür wird `now_origin` so lange um jeweils eine Sekunde dekrementiert, bis ein vorhandener Zeitstempel (`now_new`) gefunden wurde.

```
WHILE c = 0 DO
  SELECT "GRUPPE" into group_id
    FROM "RAPID_OPERATOR"."ZAEHLSPULEN"
    WHERE "TIMESTAMP" = now_new;
  IF group_id = NULL
    THEN
      now_new := ADD_SECONDS(now_new, (-1));
    ELSE
      now_origin := now_new;
      c := 1;
  END IF;
END WHILE;
```

Da ein zukünftiger Wert berechnet werden soll, werden anschließend auf den gefundenen Zeitstempel 15 Minuten (= 900 Sekunden) addiert um den letztendlich benötigten Zeitstempel zu erzeugen. In einer Schleife werden dann die letzten 15 Tage, die der selben Kategorie angehören in die Tabelle `PREDICTION_15` geschrieben um sie dort zwischen zu speichern und später weiter verwenden zu können. Statt der fest definierten Tabelle wurde zwischenzeitlich auch eine temporäre Tabelle verwendet, weil diese innerhalb der Prozedur auch wieder gelöscht werden kann, allerdings hatte die Verwendung der temporären Tabelle einen negativen Einfluss auf die Performance, sodass am Ende wieder die fest definierte Tabelle verwendet wurde.

```
FOR k IN 0 .. m DO
  IF :checker <= 14
    THEN
      INSERT INTO "RAPID_OPERATOR"."PREDICTION_15"
        SELECT *
        FROM "RAPID_OPERATOR"."ZAEHLSPULEN"
        WHERE "TIMESTAMP" = ADD_DAYS(now_origin, k*(-1)) AND GRUPPE
            = group_id;

      SELECT GRUPPE into check_group
        FROM "RAPID_OPERATOR"."ZAEHLSPULEN"
        WHERE "TIMESTAMP" = ADD_DAYS(now_origin, k*(-1));

      IF :check_group = group_id
        THEN
          checker := checker+1;
        ELSE
```

```
            continue;
        END IF;
      ELSE
    END IF;
END FOR;
```

Innerhalb einer weiteren Schleife werden dann alle Spalten der vorher befüllten Tabelle `PREDICTION_15` durchlaufen und jeweils der Mittelwert für die einzelnen Zählspulen gebildet. Da der Zugriff auf die Spalten dynamisch erfolgt kann der Mittelwert nicht direkt in einer Variablen gespeichert werden, sondern muss in einer temporären Tabelle (`#STORAGE`) zwischengespeichert werden. Aus dieser kann der Wert ausgelesen und in die endgültige Tabelle `PREDICTION` in die entsprechende Zelle eingefügt werden.

```
FOR i IN 4 .. max_columns DO
  SELECT COLUMN_NAME INTO column_name
    FROM TABLE_COLUMNS
    WHERE SCHEMA_NAME = 'RAPID_OPERATOR'
    AND TABLE_NAME = 'PREDICTION_15'
    AND POSITION = i;

  EXEC 'INSERT INTO #STORAGE (FOR_SUM)
          SELECT AVG(' ||column_name|| ')
          FROM "RAPID_OPERATOR"."PREDICTION_15"';
  SELECT SUM(FOR_SUM) INTO temp FROM #STORAGE;

  UPDATE "RAPID_OPERATOR"."PREDICTION"
    SET "15_MINUTES" = ROUND(TO_DOUBLE(temp), 0)
    WHERE "SPULE" = column_name;

  temp := 0;
  TRUNCATE TABLE "#STORAGE";
END FOR;
```

Die gleiche Prozedur wurde auch für die Prognosewerte für 30 und 60 Minuten verwendet, wobei zum einen die entsprechenden Werte auf den am Anfang ermittelten Zeitstempel `now_new` addiert wurden und zum anderen wurden die Ergebnisse in der Tabelle `PREDICTION` in andere Zellen geschrieben. Das Ergebnis, wenn alle drei Prozeduren ausgeführt wurden war, dass die Tabelle `PREDICTION` für alle Zählspulen jeweils den Prognosewert für 15, 30 und 60 Minuten zu einem spezifischen Zeitpunkt beinhaltet, der zu Beginn dem Programm übergeben wurde.

Ein möglicher Nachteil dieses Verfahrens ist, dass es nicht auf die kürzlich gemessenen Werte der Zählspule eingeht. Dadurch werden Ereignisse, die zwischen dem aktuellen Tag und dem letzten Tag, der der gleichen Gruppe zugehört, nicht berücksichtigt. Für die Tage Freitag, Samstag und Sonntag ist das jeweils eine volle Woche. Deswegen wurde ein anderes Verfahren ausgewählt, welches statt der letzten 15 Tage die letzten 15 gemessenen Werte

betrachtet und daraus die Prognose berechnet. Die Formel die der Berechnung zugrunde liegt ist die für das exponentielle Glätten

$$S_t(x) = A \sum_{i=0}^{14} (1 - A)^i x_{t-i}$$

Durch einige Tests hat sich für den Parameter A ein Wert von $0,2$ als am besten hinsichtlich der Genauigkeit des Ergebnisses erwiesen. Der neue Wert S_t wird dann aus den letzten 15 gemessen Werten berechnet, wobei die Gewichtung abnimmt, umso weiter der Wert in der Vergangenheit liegt. Da die Zeitabstände in der Wertetabelle für die Zählspulen 30 Sekunden betragen, liegt der berechnete Wert für S_t auch nur 30 Sekunden in der Zukunft. Das Ziel ist es allerdings einen Wert für 15 Minuten in der Zukunft zu berechnen, weswegen die Formel insgesamt 30 mal mit den jeweils vorher berechneten Werten ausgeführt werden muss. Das selbe Verfahren kann theoretisch für beliebige Zeitpunkte in der Zukunft ausgeführt werden, allerdings haben Vergleiche zwischen berechneten und tatsächlich gemessen Werten ergeben, dass die Werte für mehr als 15 Minuten in der Zukunft zu stark unterschiedlich sind. Deswegen ist eine Berechnung mit diesem Verfahren für mehr als 15 Minuten in der Zukunft nicht sinnvoll.

Die Prozedur, welche in dem SAP HANA System für diese Funktionalität implementiert wurde, wird im Folgenden erläutert.

Auch dieser Prozedur wird der aktuelle Zeitstempel übergeben, welcher analog zu der obigen Prozedur auf einen Zeitstempel zurückgerechnet wird, der in der Wertetabelle existiert. Von diesem Zeitstempel ausgehend werden die letzten 15 Werte aus der Wertetabelle in die Tabelle `PREDICTION_15` übertragen. Da entweder diese Prozedur oder die weiter oben beschriebene ausgeführt werden sollte, aber nicht beide gleichzeitig, und die Tabelle `PREDICTION_15` nach jedem Aufruf der Prozedur zurückgesetzt wurde, ergeben sich auch keine Konflikte.

```
FOR i IN 0 .. 14 DO

  INSERT INTO "RAPID_OPERATOR"."PREDICTION_15"
    SELECT *
    FROM "RAPID_OPERATOR"."ZAEHLSPULEN"
    WHERE "TIMESTAMP" = ADD_SECONDS(now_origin, i*(-30));

END FOR;
```

Durch drei geschachtelte Schleifen wird dann der exponentiell geglättete Wert berechnet, wobei die äußerste Schleife dafür da ist um über alle Zählspulen zu iterieren und deswegen keiner weiteren Erklärungen bedarf.

Die innersten Schleife dagegen berechnet den geglätteten Wert. Dafür wird zunächst der Wert für den aktuell zu betrachtenden Zeitstempel (`now_reverse`) aus der vorher befüllten Tabelle `PREDICTION_15` geladen. Auch hier muss das Speichern in eine Variable (`temp_raw`) wieder den Umweg über eine temporäre Tabelle (`#STORAGE_15`) gehen. Die Variable `temp_raw` wird dann gewichtet in der Variable `temp` gespeichert und anschließend

der bereits berechneten Summe `sum_temp` hinzugefügt. Abschließend muss die temporäre Tabelle zurückgesetzt werden und der Zeitstempel wird um 30 Sekunden verringert.

```
FOR j IN 0 .. 14 DO
    EXEC 'INSERT INTO "#STORAGE_15" (TEMP)
            SELECT '||:column_name||'
            FROM "RAPID_OPERATOR"."PREDICTION_15"
            WHERE "TIMESTAMP" = ''''||:now_reverse||'''';

    SELECT TEMP INTO temp_raw FROM "#STORAGE_15" ;

    temp := (POWER(0.8, j) * temp_raw);
    sum_temp := (sum_temp + temp);
    TRUNCATE TABLE "#STORAGE_15";
    now_reverse := ADD_SECONDS(now_reverse, (-30));
  END FOR;
```

Dieser Vorgang muss, wie weiter oben beschrieben, 30 mal wiederholt werden um den Wert für 15 Minuten in der Zukunft zu liefern. Dafür ist die mittlere Schleife da, welche zum einen nach jedem Durchlauf der innersten Schleife den als aktuell angenommen Zeitpunkt (`now_forward`) um 30 Sekunden erhöht und zum anderen die in der inneren Schleife berechnete Summe nach der Formel gewichtet. Zuletzt wird das Ergebnis der Berechnung (`sum_temp`) in die entsprechende Tabelle `PREDICTION_15` geschrieben und die Summe für den nächsten Durchlauf zurückgesetzt.

```
FOR i IN 1 .. 30 DO
  now_reverse := now_forward;

  FOR j IN 0 .. 14 DO
    BERECHNUNG DES GEWICHTETEN
  END FOR;

  now_forward := ADD_SECONDS(now_forward, (+30));
  sum_temp := (0.2 * sum_temp);

  EXEC 'INSERT INTO "RAPID_OPERATOR"."PREDICTION_15" ("TIMESTAMP",
     '||:column_name||')
          VALUES ('||:now_forward||', '||:sum_temp||')';

  sum_temp := 0;
END FOR;
```

Zuletzt wird die insgesamt berechnete Summe einer jeden Zählspule in der Tabelle `PREDICTION` in die entsprechende Zelle zu schreiben. Obwohl dieses Verfahren genauere Werte für den Prognosewert in 15 Minuten liefert, als das weiter oben beschriebene Verfahren, konnte es aufgrund seiner Laufzeit nicht übernommen werden. Trotz umfangreicher Optimierungsversuche war es nicht möglich die Prozedur für alle Spalten in unter 30 Sekunden auszuführen. Diese 30 Sekunden Zeitschranke wurde von der Projektgruppe

festgelegt, weil für alle 30 Sekunden neue Werte vorliegen und die Prozedur dann neu gestartet werden muss.

8.5.2 Langfristige Prognose

Die langfristige Analyse versucht das Verkehrsaufkommen in Oldenburg innerhalb eines längeren Zeitraums zu berechnen. Dazu werden hauptsächlich die Daten über die Zählspuren der Verkehrsleitzentrale in Oldenburg verwendet. Weitere Datenquellen, wie Wetterdaten oder Eventdaten werden zu einem späteren Zeitpunkt eingebunden, um die Vorhersageergebnisse weiter zu präzisieren bzw. eine Aussage über deren Sicherheit/Genauigkeit zu treffen. Im Folgenden soll die langfristige Analyse genauer erläutert werden.

Schleifendurchlauf und Befüllung der Ziel-Tabelle

Die For-Schleife iteriert durch jede Tageszeit mit einem Abstand von 30 Sekunden. Dadurch ergibt sich ein eine Menge von 2878 Iterationen (ursprünglich 2879 abzüglich einer Iteration, da die Iteration erst bei der Tageszeit '00:00:30' beginnt). In der Schleife wird mit Hilfe von dynamischen SQL ein Update der 'AVERAGES-GROUP-X' Tabelle durchgeführt indem die Selektoren Die Durchschnittswerte bilden und gleichzeitig die Parameter wie den richtigen Timestamp und die richtige Gruppe in Betracht ziehen. Anschließend wird die Zeit um 30 Sekunden erhöht, um die nächste Tageszeit zu berechnen.

```
CREATE PROCEDURE "RAPID_OPERATOR"."averager_group_2"
LANGUAGE SQLSCRIPT AS
BEGIN

DECLARE max_columns INT;
DECLARE column_name VARCHAR(50);
DECLARE zeit TIME := '00:00:30';
DECLARE i INT := 0;
DECLARE j INT := 0;
DECLARE temp DOUBLE;


SELECT MAX(POSITION) INTO max_columns
        FROM TABLE_COLUMNS
        WHERE SCHEMA_NAME = 'RAPID_OPERATOR'
        AND TABLE_NAME = 'AVERAGES_GROUP_2';

        FOR i IN 4 .. max_columns DO

                SELECT COLUMN_NAME INTO column_name
                FROM TABLE_COLUMNS
                WHERE SCHEMA_NAME = 'RAPID_OPERATOR'
                AND TABLE_NAME = 'CLEANING'
                AND POSITION = i;
```

```
                FOR j IN 0 .. 2878 DO

                        EXEC 'UPDATE "RAPID_OPERATOR"."
                           AVERAGES_GROUP_2" SET '||column_name||'
                           = (SELECT ROUND(AVG('||column_name||'),
                           0) FROM "RAPID_OPERATOR"."CLEANING"
                           WHERE "ZEIT" = '''||zeit||''' AND GRUPPE
                            = 2) WHERE "TIMESTAMP" = '''||zeit||'''
                           ';

                        zeit := Add_SECONDS(zeit, 30);

                END FOR;
                zeit := '00:00:30';
        END FOR;
END;
```

Die Variablen

Die Deklaration der Variable "zeit" und deren Festlegung auf '00:00:30' ist die Basis der ausgeführten For-Schleife. Diese Variable wird ebenfalls nach jedem kompletten Schleifendurchlauf wieder auf diesen Default-wert zurückgesetzt. Darüber hinaus gibt es einen Variable, die bei jedem Schleifensuchlauf den Namen der Zählschleife speichert und an das dynamische SQL übergibt.

Iterierung durch die Namen der Zählschleifen

Um sämtliche Zählschleifen der Tabellen zu berücksichtigen, werden zwei Select Abfragen benötigt, die mit Hilfe der Systemtabelle 'TABLE-COLUMNS' einerseits die Anzahl der maximalen Spalten, andererseits auch die Namen der entsprechenden Spalten heraussucht und diese anschließend an den ausführenden Teil des dynamischen SQLs übergibt. Die Quell-Tabelle hierbei ist 'CLEANING' wobei die Ziel-tabelle 'AVERAGES-GROUP-X' ist. Hierbei steht das 'X' für die entsprechende Gruppe, in welche die einzelnen Tage eingeordnet wurden.

Schleifendurchlauf und Befüllung der Ziel-tabelle

Die For-Schleife iteriert durch jede Tageszeit mit einem Abstand von 30 Sekunden. Dadurch ergibt sich ein eine Menge von 2878 Iterationen (ursprünglich 2879 abzüglich einer Iteration, da die Iteration erst bei der Tageszeit ‚00:00:30' beginnt). In der Schleife wird mit Hilfe von dynamischen SQL ein Update der 'AVERAGES-GROUP-X' Tabelle durchgeführt indem die Selektoren die Durchschnittswerte bilden und gleichzeitig die Parameter wie den richtigen Timestamp und die richtige Gruppe in Betracht ziehen. Anschließend wird die Zeit um 30 Sekunden erhöht, um die nächste Tageszeit zu berechnen.

Regressionsfunktion

Auf Grundlage der Zählspuren-Werte wird ein Berechnungsmodell realisiert (Regressionsfunktion), welches versucht die Werte der Zählspuren (gekennzeichnet in den Tabellen mit der Bezeichnung „ZAEHL“) zu jedem Zeitpunkt möglichst nah an den tatsächlich auftretenden Werten vorherzusagen. Die Funktion soll individuell auf 4 konkrete Klassen von Tagen angewendet werden:

- Klasse 1: Wochentage (Montag bis Donnerstag)
- Klasse 2: Wochenendtage (Freitag)
- Klasse 3: Samstag
- Klasse 4: Feiertage und Sonntage (fällt ein Tag in diese Klasse wird er nicht mehr einer der 3 anderen Klassen zugeordnet)

Zunächst werden die einzelnen Tabellen (Input- und Output-Tabellen) und Tabellentypen mittels der Prozedur CREATETABLES erzeugt. Zu diesem Zeitpunkt enthalten sie noch keine Werte. Bei einer Gruppe der erzeugten Tabellen handelt es sich um diejenigen, die die Werte aus den oben beschriebenen Tabellen AVERAGER_GROUP1-4 erhalten. Sie werden weitergegeben an die Prozedur ZAEHL, in der der Regressionsalgorithmus auf sie angewendet wird. Die erste Spalte ist ein Platzhalter für alle Tageszeiten in 30 Sekunden-Abständen („00:00:30“, „00:01:00“ usw.). In allen weiteren Spalten der Tabelle befinden sich die Bezeichner der einzelnen Zählschleifen, in denen in einem der folgenden Schritte die Mittelwerte aus den AVERAGER_GOUP-Tabellen eingetragen werden. Zudem werden auch die leeren Tabellen für die Vorhersagewerte erzeugt (PREDVAL1-4). Die Struktur dieser Tabelle ist dieselbe.

```
CREATE PROCEDURE CREATEDATATABLES()
LANGUAGE SQLSCRIPT AS
BEGIN

DROP TABLE AVERAGEVAL1;

CREATE TABLE AVERAGEVAL1 AS
(SELECT * FROM "RAPID_OPERATOR"."AVERAGES_GROUP_1" WHERE Timestamp
   BETWEEN '00:00:30' AND '23:59:30');

DROP TABLE AVERAGEVAL2;

CREATE TABLE AVERAGEVAL2 AS
(SELECT * FROM "RAPID_OPERATOR"."AVERAGES_GROUP_2" WHERE Timestamp
   BETWEEN '00:00:30' AND '23:59:30');
```

Anschließend werden die Tabellen PREDVAL1-4 ein einziges Mal mit den beschriebenen Zeitwerten befüllt. Dazu werden die Prozeduren FILLTIME1-4 aufgerufen. Damit ist die

erste Spalte mit Werten gefüllt und es entsteht eine Tabelle, welche alle Zeitwerte eines Tages aber noch keine berechneten ZAEHL-Werte enthält.

```
CREATE PROCEDURE "RAPID_OPERATOR"."FILLTIME1"
LANGUAGE SQLSCRIPT AS
BEGIN

        DECLARE k INT := 0;
        FOR k in 0 .. 2879
                DO
                        INSERT INTO "RAPID_OPERATOR"."PREDVAL" ("
                           TIMEV") VALUES (((k+1)*0.5));
        END FOR;
END;
```

Diese Tabellen mit den Werten der Tabellen AVERAGER_GROUP1-4 werden dann dem Regressionsalgorithmus ZAEHLDATA, als Input-Parameter übergeben. Die Tabellentypen PREDVALTYPE1-4 werden dem Algorithmus als Output übergeben.

```
zaehl_Data1 = SELECT * FROM "RAPID_OPERATOR"."AVERAGEVAL1";
CALL ZAEHL(:zaehl_Data1, "RAPID_OPERATOR"."PREDICTION_LONG_COF";
```

Innerhalb der Prozedur „ZAEHL“ wird zu jeder Uhrzeit mittels des Regressionsalgorithmus der korrespondierende Vorhersagewert berechnet. Dieser wird dann von dem Algorithmus als Output-Parameter zurückgegeben.

```
fPVs <- function(scattData) {
  xWerte <-data.frame()
  yWerte <- data.frame()
  xWerte <- scattData[,1]
  yWerte <- scattData[,2]
  regMod <- lm(yWerte ~ poly(xWerte,5,raw=TRUE))
  predVs <- predict(regMod,newData=TIMEV)
  return(predVs)
}
```

Durch die Rückgabe-Tabellen der Prozedur ZAEHL (PREDVAL1-4) und den Durchschnittswerten (AVERAGERGROUP1-4) ist es u.a. möglich im Frontend einen Scatterplot über den Durchschnittswerten einer jeden Zählschleife zu bilden und zu jedem Zeitpunkt die Abweichungen der vorhergesagten Werte von denen der tatsächlichen Werte zu bestimmen. In der folgenden Abbildung wird noch einmal der Ausschnitt des ER-Diagramms gezeigt, der die Tabellen, Prozeduren und deren Zusammenwirken übersichtlich darstellt, die in der langfristigen Prognose eine Rolle spielen (siehe Abbildung 33.

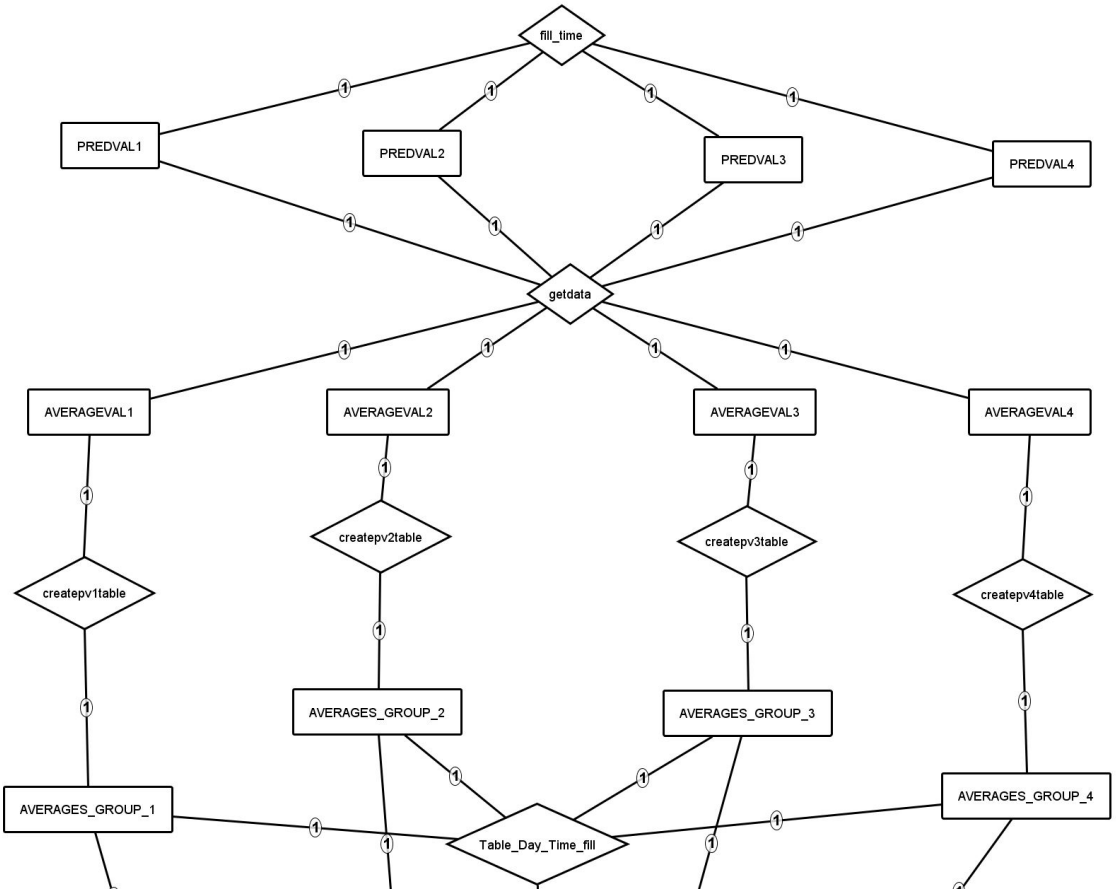


Abbildung 33: Übersicht über die Tabellen und Prozeduren, der langfristigen Prognose

8.6 CO2-Ausstoß Berechnung

Der Mittelwert des CO2 Verbrauchs (187 mikrogramm CO2 pro Auto) wird nun für die Berechnung des CO2 Verbrauchs in den verschiedenen Regionen in Oldenburg verwendet. Dafür wird der Mittelwert mit dem Verkehrsaufkommen, der über die Spulen gebietsweise in Oldenburg aufgenommen wird, multipliziert. Für Neufahrzeuge ist momentan durchschnittlich ein CO2 Ausstoß von 95 Gramm pro Kilomerter also ca. 95000 Mikrogramm pro Meter als Grenze zugelassen. Dieser Wert bestimmt bei der Darstellung der Ergebnisse den Grenzwert und macht sichtbar, welche Regionen in Oldenburg durch eine hohen bzw. niedrigen CO2 Ausstoß auffallen. Für die Berechnung des CO2 Ausstoßes für jede Zählspule zu einem bestimmten Zeitpunkt ist in dem nachfolgenden Code kenntlich gemacht. Im ersten Schritt wurde eine neue Tabelle erstellt. Die mittels des Prozedur gefüllt wurde.

```
CREATE PROCEDURE "RAPID_OPERATOR"."CO2Berechnung" (IN now timestamp
    )
LANGUAGE SQLSCRIPT AS
```

```
BEGIN

        DECLARE co2Wert INT := 187;
        DECLARE column_name VARCHAR(50);
        DECLARE i INT;
        DECLARE c INT := 0;
        DECLARE now_new TIMESTAMP := now;
        DECLARE now_origin TIMESTAMP;
        DECLARE group_id INT;
        DECLARE max_columns INT;

        WHILE c = 0 DO
                SELECT "GRUPPE" into group_id FROM "RAPID_OPERATOR"
                   ."ZAEHLSCHLEIFEN" WHERE "TIMESTAMP" = now_new;
                IF group_id = NULL
                        THEN
                                now_new := ADD_SECONDS(now_new,
                                   (-1));
                        ELSE
                                now_origin := now_new;
                                c := 1;
                END IF;
        END WHILE;

        UPDATE "RAPID_OPERATOR"."Co2_ZAEHLSCHLEIFEN" SET "TIMESTAMP
           " = now_new;

SELECT MAX(POSITION) INTO max_columns FROM TABLE_COLUMNS
WHERE "SCHEMA_NAME" = 'RAPID_OPERATOR' AND "TABLE_NAME" = '
   ZAEHLSCHLEIFEN';

        FOR i IN 4 .. max_columns DO

                SELECT COLUMN_NAME INTO column_name
                FROM TABLE_COLUMNS
                WHERE SCHEMA_NAME = 'RAPID_OPERATOR'
                AND TABLE_NAME = 'ZAEHLSCHLEIFEN'
                AND POSITION = i;

                EXEC 'UPDATE "RAPID_OPERATOR"."Co2_ZAEHLSCHLEIFEN"
                   SET '||column_name||' = (SELECT '||column_name||
                   ' FROM "RAPID_OPERATOR"."ZAEHLSCHLEIFEN" WHERE "
                   TIMESTAMP" = '''||now_new||''') * '||co2Wert||'
                   WHERE "TIMESTAMP" = '''||now_new||'''';

                END FOR;
END;
```

Zusätzlich wurde der CO2 Verbrauch ebenfalls mit den Werten aus der kurzfristigen Prognose für jeweils 15, 30 und 60 Minuten berechnet. Somit können Prognosen gegeben werden, wie sich der CO2 Ausstoß in der nächsten Stunde verändert. Für die Berechnung wurde folgender Code erstellt:

```
CREATE PROCEDURE "RAPID_OPERATOR"."Co2Prog"
LANGUAGE SQLSCRIPT AS
BEGIN

DECLARE Co2Wert INT:= 187;

INSERT INTO "RAPID_OPERATOR"."Co2Prognose" ("SPULE", "15_MINUTES",
   "30_MINUTES", "60_MINUTES")
SELECT "SPULE", "15_MINUTES", "30_MINUTES" , "60_MINUTES" FROM "
   RAPID_OPERATOR"."PREDICTION";

EXEC 'UPDATE "RAPID_OPERATOR"."Co2Prognose" SET "15_MINUTES" = "15
   _MINUTES" * '''||Co2Wert||''', "30_MINUTES" = "30_MINUTES" * '''
   ||Co2Wert||''', "60_MINUTES" = "60_MINUTES" * '''||Co2Wert||''''
   ;

END;
```

9 Darstellung der Ergebnisse

Zur Darstellung der Projektergebnisse wurde ein Onlineportal entwickelt. Als Grundlage für das Portal dient das aufgestellte Anwendungsszenario, aus welchem die Akteure und Funktionen abgeleitet werden. Zur Entwicklung des Portals werden APIs und Frameworks auf dem Webserver der Projektgruppe eingerichtet. Das Produkt wird in Form einer Webapplikation realisiert und ermöglicht den Zugriff durch die Anwender über den Browser. Die Verkehrsleitzentrale der Stadt Oldenburg als Experte nutzt weitere Tools zur Verkehrsanalyse über diesen Weg, wodurch die Anwender in einer vertrauten Umgebung arbeiten.

9.1 Anforderungen

Innerhalb des folgenden Kapitels, werden die Anforderungen an das zu implementierende Webportal formuliert. Dabei teilen sich die Anforderungen in funktionale und nicht funktionale Anforderungen auf. Die funktionalen Anforderungen sind Anforderungen die das Webportal in jedem Fall bieten muss. Die nicht funktionalen Anforderungen definieren Anforderungen an das Gesamtportal und nicht an einzelne Komponenten. Zusätzlich wurde eine Einteilung des Portals in drei Management Bereiche vorgenommen:

- Information Management – Dient der Verarbeitung von externen Kommunikationsanfragen
- User-Management – Behandelt alle Benutzer spezifischen Angelegenheiten
- Portal-Management – Stellt die Basis-Funktionalität des Portals dar

9.1.1 Funktionale Anforderungen

Die funktionalen Anforderungen gliedern sich laut [Bel07] in die Prioritäten: Muss, Soll und Kann. Muss Anforderungen müssen in jedem Fall erfüllt sein. Soll Anforderungen sind Features die zusätzlich eingebaut werden sollten. Die Kann Anforderungen dienen beispielsweise dem nächsten Release als Einstiegspunkt für Verbesserungen.

Priorität Muss

1. Das Information-Management muss den Kontakt mit den Administratoren des Portals sicherstellen.
   a) Das Information-Management muss einen Kontaktbereich zur Verfügung stellen.
   b) Das Information-Management muss den Namen des Kontaktnehmers erfragen.
   c) Das Information-Management muss die E-Mail Adresse des Kontaktnehmers erfragen.
   d) Das Information-Management muss ein Nachrichtenfeld zur Verfügung stellen.

e) Das Information-Management muss die Nachricht übermitteln.

2. Das Information-Management muss einen Impressum-Bereich zur Verfügung stellen.
3. Das User-Management muss Rollen verteilen.
   a) Das User-Management muss einem registrierten aber nicht freigeschaltetem User die Rolle 3 (Future Customer) zuweisen.
   b) Das User-Management muss einem registrierten und freigeschalteten User die Rolle 1 (Customer) zuweisen.
   c) Das User-Management muss die Rolle 2 (Administrator) bereitstellen.
      i. Das User-Management muss dem Administrator eine geeignete Übersicht über Administratoren, Customer und Future Customer zur Verfügung stellen.
      ii. Das User-Management muss dem Administrator die Möglichkeit geben Future Customer freizuschalten.
      iii. Das User-Management muss dem Administrator die Möglichkeit geben nicht autorisierte Future Customer zu löschen.
      iv. Das User-Management muss dem Administrator die Möglichkeit geben Customer zu deaktivieren.
      v. Das User-Management muss dem Administrator die Möglichkeit geben Customer zu aktivieren.
      vi. Das User-Management muss dem Administrator die Möglichkeit geben deaktivierte Customer zu löschen.
4. Das User-Management muss die Registrierung eines nicht registrierten Users ermöglichen.
   a) Das User-Management muss bei der Registrierung die E-Mail-Adresse erfragen.
   b) Das User-Management muss bei der Registrierung die Wiederholung der E-Mail-Adresse erfragen.
   c) Das User-Management muss bei der Registrierung das Passwort erfragen.
   d) Das User-Management muss bei der Registrierung die Wiederholung des Passworts erfragen.
   e) Das User-Management muss bei der Registrierung den Vornamen erfragen.
   f) Das User-Management muss bei der Registrierung den Nachnamen erfragen.
   g) Das User-Management muss bei der Registrierung die Organisation erfragen.
   h) Das User-Management muss zum Abschluss der Registrierung dem nicht registrierten User die Möglichkeit bieten, die Richtigkeit seiner Angaben mittels einer Checkbox zu bestätigen.
   i) Das User-Management muss die Rolle Future Customer eintragen.

5. Das User-Management muss die Anmeldung (Login) eines Customers ermöglichen.
   a) Das User-Management muss bei der Anmeldung den Usernamen erfragen.
   b) Das User-Management muss bei der Anmeldung das Passwort erfragen.
   c) Das User-Management muss bei der Anmeldung die Speicherung des Passworts für weitere Sessions ermöglichen.
6. Das User-Management muss die Abmeldung (Logout) eines Customers ermöglichen.
   a) Das User-Management muss die Session zerstören.
7. Das User-Management muss die Funktion „Passwort vergessen“ bereitstellen
   a) Das User-Management muss die E-Mail Adresse des betroffenen Customers erfragen.
   b) Das User-Management muss sicherstellen, dass die angegebene E-Mail Adresse bereits registriert ist.
   c) Das User-Management muss einen einzigartigen Link mit Weiterleitungsfunktion generieren.
   d) Das User-Management muss eine E-Mail an die eingegebene Adresse schicken.
   e) Das User-Management muss aus Sicherheitsgründen den Hash Wert des einzigartigen Links vergleichen und überprüfen.
   f) Das User-Management muss automatisch im vorgesehenen Feld des Passwort erneuern Formulars die E-Mail Adresse eintragen.
   g) Das User-Management muss das neue Passwort erfragen.
   h) Das User-Management muss die Wiederholung des neuen Passworts erfragen.
8. Das Portal-Management muss dem Customer eine Hauptmenüzeile zur Verfügung stellen.
   a) Das Portal-Management muss dem Customer innerhalb der Hauptmenüzeile den Logout ermöglichen.
   b) Das Portal-Management muss dem Customer innerhalb der Hauptmenüzeile die Analyseübersicht anzeigen.
   c) Das Portal-Management muss dem Customer innerhalb der Hauptmenüzeile die Profilübersicht anzeigen.
   d) Das Portal-Management muss dem Administrator innerhalb der Hauptmenüzeile das Administrations-Menü anzeigen.
9. Das Portal-Management muss dem Customer eine Karte über den Stadt- und Nahbereich Oldenburgs bereitstellen.
10. Das Portal-Management muss dem Customer eine Widget Übersicht bereitstellen.

a) Das Portal-Management muss dem Customer die Auswahl des Widgets Wetter ermöglichen.
   i. Das Portal-Management muss dem Customer innerhalb des Widgets Wetter eine aktuelle Übersicht über Lufttemperatur und Bodentemperatur am betrachteten Tag bereitstellen.

b) Das Portal-Management muss dem Customer die Auswahl des Widgets Veranstaltungen ermöglichen.
   i. Das Portal-Management muss dem Customer innerhalb des Widgets Veranstaltungen eine aktuelle Übersicht über Veranstaltungen am betrachteten Tag bereitstellen.

c) Das Portal-Management muss dem Customer die Auswahl des Widgets Nahverkehr-Bus ermöglichen.
   i. Das Portal-Management muss dem Customer innerhalb des Widgets Nahverkehr-Bus das Stadtliniennetz Oldenburgs auf der Karte anzeigen.
   ii. Das Portal-Management muss dem Customer innerhalb des Widgets Nahverkehr-Bus alle Haltestellen des Stadtliniennetzes Oldenburgs auf der Karte anzeigen

d) Das Portal-Management muss dem Customer die Auswahl des Widgets Nahverkehr-Zug ermöglichen.
   i. Das Portal-Management muss dem Customer innerhalb des Widgets Nahverkehr-Zug eine aktuelle Übersicht über alle ankommenden Züge am betrachteten Tag und zur betrachteten Uhrzeit mit einer Stunde Vorlaufzeit bereitstellen.

e) Das Portal-Management muss dem Customer die Auswahl des Widgets Ampelanlagen ermöglichen.
   i. Das Portal-Management muss dem Customer innerhalb des Widgets Ampelanlagen alle in Oldenburg vorhandenen Ampelanlagen anzeigen.

f) Das Portal-Management muss dem Customer die Auswahl des Widgets Zählspulen ermöglichen.

11. Das Portal-Management muss dem Customer die Möglichkeit bieten die ausgewählten Widgets auf dem Kartenausschnitt anzuzeigen.

12. Das Portal-Management muss einen Widget Teilanzeigebereich zur Verfügung stellen.
    a) Das Portal-Management muss bei Auswahl das Widget Wetter im Teilanzeigenbereich anzeigen.
    b) Das Portal-Management muss bei Auswahl das Widget Veranstaltungen im Teilanzeigenbereich anzeigen.

c) Das Portal-Management muss bei Auswahl das Widget Naverkehr-Zug im Teilanzeigenbereich anzeigen.

13. Das Portal-Management muss eine Möglichkeit bereitstellen, den Widget Teilanzeigenbereich auszublenden.

14. Das Portal-Management muss den Verkehrsdurchsatz der eingetragenen Zählschleifen als Heatmap auf der Karte darstellen.

15. Das Portal-Management muss jede spezifisch ausgewählte Zählschleife interaktiv anzeigen.
    a) Das Portal-Management muss zu einer ausgewählten Zählschleife einen klickbaren Unterbereich darstellen.
        i. Das Portal-Management muss den Unterbereich der ausgewählten Zählschleife in die vorhandene Darstellung einbetten.
        ii. Das Portal-Management muss innerhalb des Unterbereiches der ausgewählten Zählschleife die kurzfristige Prognose als Graph anzeigen.
        iii. Das Portal-Management muss innerhalb des Unterbereichs der ausgewählten Zählschleife die kurzfristige Prognosewerte im 15 Min., 30 Min. und 60 Min. Takt anzeigen.
        iv. Das Portal-Management muss innerhalb des Unterbereichs der ausgewählten Zählschleife die langfristige Prognose als Graph anzeigen.
        v. Das Portal-Management muss innerhalb des Unterbereichs der ausgewählten Zählschleife die vom Customer ausgewählten Widgets anzeigen.

Priorität Soll

1. Das User-Management soll dem Customer die Möglichkeit bieten, sein Userprofil einzusehen.
    a) Das User-Management soll den vom Customer bei der Registrierung eingegebenen Vornamen anzeigen.
    b) Das User-Management soll den vom Customer bei der Registrierung eigegebenen Nachnamen anzeigen.
    c) Das User-Management soll die vom Customer bei der Registrierung eingegebene Organisation anzeigen.
    d) Das User-Management soll die vom Customer bei der Registrierung eingegebene E-Mail Adresse anzeigen.
    e) Das User-Management soll die Rolle des Customers darstellen.

2. Das Portal-Management soll das aktuelle Datum und die aktuelle Uhrzeit anzeigen.

Priorität Kann

1. Das User-Management kann dem Customer die Möglichkeit bieten sein Userprofil zu bearbeiten.
2. Das User-Management kann dem Customer die Möglichkeit bieten sein Userprofil zu löschen.
3. Das Portal-Management kann dem Customer innerhalb des Widgets Nahverkehr-Zug den Standort des Bahnhofs per Symbol anzeigen.
4. Das Portal-Management kann dem Customer innerhalb des Widgets Nahverkehr-Bus alle nach dem offiziell vorgegebenen Nahverkehr-Fahrplan fahrenden Busse als Symbol anzeigen.
5. Das Portal-Management kann dem Customer innerhalb des Widgets Veranstaltungen alle Veranstaltungen am betrachteten Tag als Symbol anzeigen.

9.1.2 Nicht-funktionale Anforderungen

Die nichtfunktionalen Anforderungen gliedern sich in Qualitätsanforderungen und technische Anforderungen. Die Qualitätsanforderungen sind laut [Bal08] in Funktionalität, Zuverlässigkeit, Benutzbarkeit, Effizienz, Wartbarkeit und Portabilität unterteilt. Die Einteilung in den Prioritäten Muss, Soll und Kann bleibt bestehen, verwischt aber im weiteren Verlauf.

Qualitätsanforderungen

Funktionalität

Sicherheit

1. Das User-Management muss gewährleisten, dass nur Customer Zugriff auf die Inhalte des Portals erhalten.
2. Das User-Management muss beim Login SQL Einschleusungen (SQL Injection) verhindern.
3. Das User-Management muss während des Passwort-Vergessen Vorgangs SQL Einschleusungen (SQL Injection) verhindern.
4. Das User-Management muss das Passwort während der Registrierung verschlüsselt in der Datenbank speichern.
5. Das User-Management muss sicherstellen, dass nur Administratoren mit der Rolle 2 Zugriff auf den Administrationsbereich haben.

Zuverlässigkeit
Fehlertoleranz

1. Das User-Management muss bei fehlerhaften Eingaben einen eindeutigen Rückgabewert liefern.
2. Das Portal-Management muss bei einer Fehlfunktion den Customer zurück auf die Analysesicht leiten.

Wiederherstellbarkeit

1. Das Portal-Management kann eigenständig Backups durchführen.
2. Das Portal-Management kann eigenständig Backups übergangsweise in das Portal einführen.

Benutzbarkeit
Erlernbarkeit

1. Das Portal-Management muss einem Customer eine schnelle und leichte Einarbeitung in das Portal ermöglichen.
2. Das Information-Management kann während der Nutzung von Portalfunktionalitäten Hilfestellungen bereitstellen.

Bedienbarkeit

1. Das Portal-Management muss einem Customer die Bedienung ohne technische Vorkenntnisse ermöglichen.

Effizienz
Zeitverhalten

1. Das Portal-Management kann durch Benutzung der SAP HANA Datenbank einen Zeitvorteil gegenüber herkömmlichen Datenbanken in den Analysealgorithmen erzielen.

Wartbarkeit
Stabilität

1. Das Portal-Management muss eine stabile Laufweise des Portals ermöglichen.
2. Das User-Management muss eine fehlerfreie Benutzerverwaltung ermöglichen.

Portabilität
Installierbarkeit

1. Das Portal-Management muss einem Customer die Möglichkeit bieten, dass Portal mit dem Internetbrowser Mozilla Firefox ab der Version 38.x zu nutzen.
2. Das Portal-Management muss einem nicht Customer die Möglichkeit bieten, dass Portal mit dem Internetbrowser Internet Explorer ab der Version 11.x zu nutzen.
3. Das Portal-Management muss einem Customer die Möglichkeit bieten, dass Portal mit dem Internetbrowser Google Chrome ab der Version 42.x zu nutzen.
4. Das Portal-Management muss einem Customer die Möglichkeit bieten, dass Portal mit dem Internetbrowser Safari ab der Version 10.x zu nutzen.
5. Das Portal-Management muss einem Customer die Möglichkeit bieten, dass Portal mit dem Internetbrowser Microsoft Edge zu nutzen.

Technische Anforderungen

1. Das Gesamtportal muss als Webanwendung auf einem Abteilungsinternen Server realisiert werden.
2. Das Gesamtportal muss im Backend eine SAP HANA Datenbank für Analysealgorithmen verwenden.
3. Das Portal-Management muss einem Customer die Bedienung des Portals in einer Auflösung von 1024 x 768 ermöglichen.
4. Das Portal-Management muss automatisch eine Umstellung des Portals auf mobilen Endgeräten vornehmen.

9.2 Anwendungsfälle

ID	AF01
Bezeichnung	Registrierung (Muss)
Akteure	Future Customer
Anfangszustand	Der User befindet sich auf der Portalstartseite
Ereignisfluss	1. Der User klickt auf den Link „Registrieren“ 2. Das System liefert dem User ein übergeordnetes Fenster zurück mit den Feldern „E-Mail“, „E-Mail wiederholen“, „Passwort“, „Passwort wiederholen“, „Vorname“, „Nachname“, „Organisation“ zurück 3. Der User fällt die Felder aus und bestätigt darüber hinaus die allgemeinen Geschäftsbedingungen durch Anklicken der Checkbox 4. Das System überprüft die Eingabe des Benutzernamens mit den bereits vorhandenen Einträgen der Datenbank. Falls dieser Benutzername vorhanden ist, wird eine Fehlermeldung ausgegeben. Andernfalls wird ein neuer Customer angelegt 5. Der Administrator schaltet diesen Customer dann frei 6. Das System versendet daraufhin eine Bestätigungsemail an den Customer
Endzustand	Der Customer kann sich mit seinen Anmeldedaten auf der Portalseite anmelden
Referenzierte Anforderungen	F1.1,F1.2,F1.3,F3.1, F3.2, F3.3, F4, F16, F18, F19, NF12,

ID	AF02
Bezeichnung	Anmelden (Muss)
Akteure	Customer
Anfangszustand	Der Customer befindet sich auf der Startseite des Portals

Ereignisfluss	1. Der Customer klickt auf den Link „Login“ 2. Das System gibt eine Maske zurück in der der Customer seine Anmeldedaten eintragen soll 3. Der Customer trägt in die Formularfelder „Benutzername“ und “Passwort“ seine Benutzerdaten ein 4. Das System prüft die vom Customer eingegebenen Anmeldedaten mit den Datensätzen in der Tabelle „Benutzerdaten“ 5. Nach erfolgreicher Prüfung gibt das System dem Customer die Kartenansicht zurück
Endzustand	Der Customer ist eingeloggt und befindet sich auf der Kartenansicht
Referenzierte Anforderungen	F5, NF1-4

ID	AF03
Bezeichnung	Fehlerhaftes Anmelden (Muss)
Akteure	Customer
Anfangszustand	Der Customer befindet sich auf der Startseite des Portals
Ereignisfluss	1. Der Customer klickt auf den Link „Login“ 2. Das System gibt eine Maske zurück in der der Customer seine Anmeldedaten eintragen soll 3. Der Customer trägt in die Formularfelder „Benutzername“ und “Passwort“ seine Benutzerdaten ein 4. Das System prüft die vom Customer eingegebenen Anmeldedaten mit den Datensätzen in der Tabelle „Benutzerdaten“ 5. Das System liefert dem Customer nach der Nicht-Übereinstimmung die Anzeige: „Sie haben einen falschen Benutzernamen oder ein falsches Passwort eingegeben! Bitte versuchen Sie es ein weiteres Mal oder wenden Sie sich an den Administrator¡‘
Endzustand	Der Customer ist eingeloggt und befindet sich auf der Ansicht „Dashboard“
Referenzierte Anforderungen	F5, NF12

ID	AF04
Bezeichnung	Abmelden (Muss)
Akteure	Customer (angemeldet)
Anfangszustand	Der Customer ist angemeldet und befindet sich auf einer der Portalseiten
Ereignisfluss	1. Der Customer klickt auf den oben rechts befindlichen Link „Abmelden“ 2. Das System verweist den Customer zurück auf die Portalstartseite auf der die Meldung „Sie haben sich erfolgreich abgemeldet“ angezeigt wird

Endzustand	Der Customer befindet sich auf der Portalstartseite
Referenzierte Anforderungen	F6, NF12

ID	AF05
Bezeichnung	Passwort vergessen (Muss)
Akteure	Customer (nicht angemeldet)
Anfangszustand	Der Customer befindet sich auf der Portalstartseite
Ereignisfluss	1. Der Customer wählt die Funktion „Passwort vergessen“ aus 2. Das System bietet dem Customer ein übergeordnetes Fenster an, in dem die E-Mail Adresse eingetragen wird 3. Der Customer übersendet seine Anfrage dem System 4. Das System überprüft, ob ein Customer mit dieser E-Mail Adresse vorhanden ist und sendet ihm ein temporäres Passwort zu. Zugleich wird in der Datenbank das alte Passwort des Customers durch das temporäre Passwort ersetzt. 5. Der Customer meldet sich beim nächsten Login mit dem temporären Passwort an. 6. Das System leitet den Customer auf ein übergeordnetes Fenster weiter, in dem ein neues Passwort angelegt wird. 7. Der Customer legt ein neues Passwort an und bestätigt dies. 8. Das System setzt das neue Passwort in die Datenbank ein.

Endzustand	Der Customer hat ein neues Passwort angelegt und kann sich damit anmelden.
Referenzierte Anforderungen	F3, F7, NF2, NF3, NF12

ID	AF06
Bezeichnung	Customer sperren (Muss)
Akteure	Administrator (angemeldet)
Anfangszustand	Der Administrator befindet sich in der Ansicht „Benutzerverwaltung“
Ereignisfluss	1. Der Administrator deaktiviert den Customer durch Anklicken des Deaktivieren-Buttons hinter dem Benutzerobjekt
Endzustand	Der Administrator befindet sich in der Ansicht „Benutzerverwaltung“ und der Customer ist deaktiviert und kann sich somit nicht mehr mit seinen alten Anmeldedaten im Portal anmelden
Referenzierte Anforderungen	F3.3.4, F3.3.5

ID	AF07
Bezeichnung	Customer entsperren (Muss)
Akteure	Administrator (angemeldet)
Anfangszustand	Der Administrator befindet sich Ansicht „Benutzerverwaltung“
Ereignisfluss	1. Der Administrator aktiviert den Customer durch Anklicken des Aktivieren-Buttons hinter dem Benutzerobjekt
Endzustand	Der Administrator befindet sich in der Ansicht „Benutzerverwaltung“ und der Customer ist deaktiviert und kann sich somit nicht mehr mit seinen alten Anmeldedaten im Portal anmelden
Referenzierte Anforderungen	F3, F8.4, NF1-5, NF15

ID	AF08
Bezeichnung	Benutzerprofil ansehen (Muss)
Akteure	Customer
Anfangszustand	Der Customer befindet sich auf der Portalstartseite und die Kartendarstellung wird angezeigt

Ereignisfluss	1. Der Customer klickt auf den oben rechts stehenden Link „<Benutzername>“ 2. Das System gibt ein Dropdown-Menü zurück mit den Auswahlmöglichkeiten „Profil“ und “Logout“ 3. Der Customer klickt auf den Link „Profil“ 4. Das System liefert die Ansicht mit den Benutzerangaben zurück
Endzustand	Der Customer befindet sich auf der Ansicht Profil
Referenzierte Anforderungen	F8.3, NF10, NF12

ID	AF09
Bezeichnung	AGBs ansehen (Muss)
Akteure	Customer
Anfangszustand	Der Customer befindet sich auf der Portalstartseite
Ereignisfluss	1. Der Customer klickt auf den Link „AGBs“ 2. Das System liefert die Ansicht „AGBs“ zurück
Endzustand	Der Customer befindet sich auf der Ansicht „AGBs“
Referenzierte Anforderungen	NF10, NF12

ID	AF10
Bezeichnung	Impressum ansehen (Muss)
Akteure	Customer
Anfangszustand	Der Customer befindet sich auf der Portalstartseite
Ereignisfluss	1. Der Customer klickt auf den Link „Impressum“ 2. Das System liefert die Ansicht „Impressum“ zurück

Endzustand	Der Customer befindet sich auf der Ansicht „Impressum“
Referenzierte Anforderungen	F2, NF10, NF12

ID	AF11
Bezeichnung	Kontakte ansehen (Muss)
Akteure	Customer
Anfangszustand	Der Customer befindet sich auf der Portalstartseite
Ereignisfluss	1. Der Customer klickt auf den Link „Kontakte“ 2. Das System liefert die Ansicht „Kontakte“ zurück
Endzustand	Der Customer befindet sich auf der Ansicht „Kontakte“
Referenzierte Anforderungen	F2, NF10, NF12

ID	AF12
Bezeichnung	Widgets anzeigen – Allgemein (Muss)
Akteure	Customer (angemeldet)
Anfangszustand	Der Customer befindet sich in der Hauptansicht des Portals
Ereignisfluss	1. Der Customer klickt auf das Zahnradsymbol oben rechts in der Darstellung 2. Das System gibt dem Customer eine Übersicht über die einzelnen Widgets (Wetter, Veranstaltungen, Nahverkehr – Bus, Nahverkehr – Zug, Schadstoffbelastung, Ampelanlagen, Zählspulen, Routenplaner) zurück die angezeigt werden können 3. Der Customer klickt auf eines der Widgets die nicht durch ein Häkchen-Symbol gekennzeichnet sind und somit noch nicht angezeigt werden auf der Kartendarstellung 4. Das System kennzeichnet das Widget mit einer ausgewählten Checkbox an 5. Der Benutzer klickt auf den Button „Einstellungen ändern“ 6. Das System übernimmt die Einstellungen, lädt die Kartendarstellung neu und zeigt, die vom Benutzer ausgewählten Widgets nun an
Endzustand	Der Customer befindet sich auf der Startseite der Analyse, die Kartendarstellung ist aufgerufen und die von ihm ausgewählten Widgets werden nun angezeigt
Referenzierte Anforderungen	F10, F11, F12, NF13

ID	AF13
Bezeichnung	Wetter anzeigen lassen (Muss)
Akteure	Customer (angemeldet)
Anfangszustand	Der Customer befindet sich auf der Startseite der Analyse und die Kartendarstellung ist aufgerufen
Ereignisfluss	1. Der Customer klickt auf das Zahnrad-Symbol am rechten oberen Bildrand 2. Das System liefert eine Auswahl mit den Elementen zurück, die auf der Karte anzuzeigen sind 3. Der Customer klickt mit der Maus auf die Checkbox hinter dem Label „Wetter“ 4. Das System kennzeichnet diese Checkbox mit einem Häkchen 5. Anschließend klickt der Customer auf den Button „Einstellungen ändern“ um die Einstellungen zu übernehmen 6. Das System aktualisiert die Kartendarstellung im unteren Bildrand und zeigt dort ein Diagramm an, welches die Temperatur und den Luftdruck des aktuell eingestellten Tages anzeigt
Endzustand	Der Customer befindet sich in der Kartendarstellung und die Wetterinformationen werden am unteren Bildrand angezeigt
Referenzierte Anforderungen	F10, F11, F12, NF13

ID	AF14
Bezeichnung	Veranstaltungen anzeigen lassen (Muss)
Akteure	Customer (angemeldet)
Anfangszustand	Der Customer befindet sich auf der Startseite der Analyse und die Kartendarstellung ist aufgerufen
Ereignisfluss	1. Der Customer klickt auf das Zahnrad-Symbol am rechten oberen Bildrand 2. Das System liefert eine Auswahl mit den Elementen zurück, die auf der Karte anzuzeigen sind 3. Der Customer klickt mit der Maus auf die Checkbox hinter dem Label „Veranstaltungen“ 4. Das System kennzeichnet diese Checkbox mit einem Häkchen 5. Anschließend klickt der Customer auf den Button „Einstellungen ändern“ um die Einstellungen zu übernehmen 6. Das System aktualisiert die Kartendarstellung im unteren Bildrand und zeigt dort die Veranstaltung mit deren Uhrzeit zum eingestellten Tag ein
Endzustand	Der Customer befindet sich in der Kartendarstellung und die Veranstaltungsinformationen werden am unteren Bildrand angezeigt
Referenzierte Anforderungen	F10, F11, F12, NF13

ID	AF15
Bezeichnung	Nahverkehr Bus anzeigen lassen (Muss)
Akteure	Customer (angemeldet)
Anfangszustand	Der Customer befindet sich auf der Startseite der Analyse und die Kartendarstellung ist aufgerufen
Ereignisfluss	1. Der Customer klickt auf das Zahnrad-Symbol am rechten oberen Bildrand 2. Das System liefert eine Auswahl mit den Elementen zurück, die auf der Karte anzuzeigen sind 3. Der Customer klickt mit der Maus auf die Checkbox hinter dem Label „Nahverkehr Bus“ 4. Das System kennzeichnet diese Checkbox mit einem Häkchen 5. Anschließend klickt der Customer auf den Button „Einstellungen ändern“ um die Einstellungen zu übernehmen 6. Das System aktualisiert die Kartendarstellung und zeigt auf der Karte alle Busstrecken, Haltestellen und aktuelle Haltepositionen der Busse an
Endzustand	Der Customer befindet sich in der Kartendarstellung und die oben beschriebenen Businformationen sind auf der Karte angezeigt
Referenzierte Anforderungen	F10, F11, F12, NF13

ID	AF16
Bezeichnung	Nahverkehr–Bahn anzeigen lassen
Akteure	Customer (angemeldet)
Anfangszustand	Der Customer befindet sich auf der Startseite der Analyse und die Kartendarstellung ist aufgerufen
Ereignisfluss	1. Der Customer klickt auf das Zahnrad-Symbol am rechten oberen Bildrand 2. Das System liefert eine Auswahl mit den Elementen zurück, die auf der Karte anzuzeigen sind 3. Der Customer klickt mit der Maus auf die Checkbox hinter dem Label „Nahverkehr-Bahn“ 4. Das System kennzeichnet diese Checkbox mit einem Häkchen 5. Anschließend klickt der Customer auf den Button „Einstellungen ändern“ um die Einstellungen zu übernehmen 6. Das System aktualisiert die Kartendarstellung und zeigt auf der Karte den Bahnhof und am unteren Bildrand Informationen zu den aktuellen Abfahrt-Zeiten an
Endzustand	Der Customer befindet sich in der Kartendarstellung und die oben beschriebenen Bahn-Informationen sind auf der Karte angezeigt
Referenzierte Anforderungen	F10, F11, F12, NF13

ID	AF17
Bezeichnung	Schadstoffbelastung anzeigen lassen
Akteure	Customer (angemeldet)
Anfangszustand	Der Customer befindet sich auf der Startseite der Analyse und die Kartendarstellung ist aufgerufen
Ereignisfluss	1. Der Customer klickt auf das Zahnrad-Symbol am rechten oberen Bildrand 2. Das System liefert eine Auswahl mit den Elementen zurück, die auf der Karte anzuzeigen sind 3. Der Customer klickt mit der Maus auf die Checkbox hinter dem Label „Schadstoffbelastung“ 4. Das System kennzeichnet diese Checkbox mit einem Häkchen 5. Anschließend klickt der Customer auf den Button „Einstellungen ändern“ um die Einstellungen zu übernehmen 6. Das System aktualisiert die Kartendarstellung und zeigt auf im unteren Bildrand Informationen der Schadstoffbelastung an
Endzustand	Der Customer befindet sich in der Kartendarstellung und die oben beschriebenen Schadstoff-Informationen sind auf der Karte angezeigt
Referenzierte Anforderungen	F10, F11, F12, NF13

ID	AF18
Bezeichnung	Ampelanlagen anzeigen lassen
Akteure	Customer (angemeldet)
Anfangszustand	Der Customer befindet sich auf der Startseite der Analyse und die Kartendarstellung ist aufgerufen
Ereignisfluss	1. Der Customer klickt auf das Zahnrad-Symbol am rechten oberen Bildrand 2. Das System liefert eine Auswahl mit den Elementen zurück, die auf der Karte anzuzeigen sind 3. Der Customer klickt mit der Maus auf die Checkbox hinter dem Label „Ampelanlagen“ 4. Das System kennzeichnet diese Checkbox mit einem Häkchen 5. Anschließend klickt der Customer auf den Button „Einstellungen ändern“ um die Einstellungen zu übernehmen 6. Das System aktualisiert die Katendarstellung und zeigt auf der Karte die Ampelanlagen in Form spezieller Elemente an
Endzustand	Der Customer befindet sich in der Kartendarstellung und die Ampelanlagen werden mittels Symbol auf der Karte dargestellt
Referenzierte Anforderungen	F10, F11, F12, NF13

ID	AF19
Bezeichnung	Zählspulen anzeigen lassen
Akteure	Customer (angemeldet)
Anfangszustand	Der Customer befindet sich auf der Startseite der Analyse und die Kartendarstellung ist aufgerufen
Ereignisfluss	1. Der Customer klickt auf das Zahnrad-Symbol am rechten oberen Bildrand 2. Das System liefert eine Auswahl mit den Elementen zurück, die auf der Karte anzuzeigen sind 3. Der Customer klickt mit der Maus auf die Checkbox hinter dem Label „Zählspulen“ 4. Das System kennzeichnet diese Checkbox mit einem Häkchen 5. Anschließend klickt der Customer auf den Button „Einstellungen ändern“ um die Einstellungen zu übernehmen 6. Das System aktualisiert die Katendarstellung und zeigt auf der Karte die Zählspulen in Form spezieller Elemente an
Endzustand	Der Customer befindet sich in der Kartendarstellung und die Zählspulen werden mittels Symbol auf der Karte dargestellt
Referenzierte Anforderungen	F10, F11, F12, NF13

ID	AF20
Bezeichnung	Zu einer Zählspule per Klicken navigieren
Akteure	Customer (angemeldet)
Anfangszustand	Der Customer befindet sich auf der Startseite der Analyse und die Kartendarstellung ist aufgerufen
Ereignisfluss	1. Der Customer klickt auf eines der nummerierten Kreis-Symbole 2. Das System zoomt in die Kartendarstellung an dieser Stelle hinein (die Zählspulen-Symbole, verteilen sich entsprechend der neuen Skalierung) 3. Der Customer wiederholt den vorherigen Schritt solange, bis er auf der tiefst möglichen Ebene ist. Klickt er nun ein weiteres Mal gibt es zwei Möglichkeiten a) Nach dem Klick zeigt ihm das System direkt die Diagrammdarstellungen für die kurz- und langfristige Prognose oder... b) ....Nach einem Klick liefert das System dem Benutzer eine Auswahl der Zählspulen, die an der entsprechenden geografischen Position liegen. Klickt er diese an, so liefert ihm das System die Darstellungen der kurz- und langfristigen Prognose zu dieser Zählspule
Endzustand	Der Customer befindet sich in der Kartendarstellung und die Darstellungen der kurz- und langfristigen Prognose werden angezeigt
Referenzierte Anforderungen	F14, F15, NF6, NF7, NF13

ID	AF21
Bezeichnung	Abschnitte in der kurzfristigen Prognose markieren
Akteure	Customer (angemeldet)
Anfangszustand	Der Customer befindet sich auf der Startseite der Analyse, die Kartendarstellung ist angezeigt und es wird eine der Zählspulen mit ihren Darstellungen der kurz- und langfristigen Prognose angezeigt

Ereignisfluss	1. Der Customer führt mit einem Mouseover über die im Diagramm befindliche Line bis er sich direkt über einer der in der x-Achse befindlichen Uhrzeiten steht 2. Das System gibt dem Benutzer als Tooltip die Information über die Zeit und die Anzahl der Autos wieder
Endzustand	Der Customer befindet sich auf der Startseite der Analyse, die Kartendarstellung und die Diagramme werden angezeigt und im oberen Diagramm der langfristigen Prognose werden die oben genannten Informationen innerhalb eines Tooltips angezeigt
Referenzierte Anforderungen	F14, F15, NF6, NF7, NF13

ID	AF22
Bezeichnung	Abschnitte in der langfristigen Prognose markieren
Akteure	Customer (angemeldet)
Anfangszustand	Der Customer befindet sich auf der Startseite der Analyse, die Kartendarstellung wird angezeigt und es wird eine der Zählspulen mit ihren Darstellungen der kurz- und langfristigen Prognose angezeigt
Ereignisfluss	1. Der Customer führt mit einem Mouseover über einen der Punkte innerhalb der Scatterplot-Darstellung der langfristigen Prognose 2. Das System gibt dem Benutzer als Tooltip die Information über den Namen der Zählspule, der Uhrzeit und der Anzahl an Verkehrsmitteln, die voraussichtlich innerhalb der nächsten Stunde über diese Zählspule fahren

Endzustand	Der Customer befindet sich auf der Startseite der Analyse, die Kartendarstellung und die Diagramme werden angezeigt und im oberen Diagramm der langfristigen Prognose werden die oben genannten Informationen innerhalb eines Tooltips angezeigt
Referenzierte Anforderungen	F14, F15, NF6, NF7, NF13

ID	AF23
Bezeichnung	In die Kartendarstellung hineinzoomen
Akteure	Customer (angemeldet)
Anfangszustand	Der Customer befindet sich auf der Startseite der Analyse und die Kartendarstellung ist aufgerufen
Ereignisfluss	1. Der Customer zieht die Kartendarstellung mittels Doppelklick und Festhalten der Maus in die gewünschte Position 2. Anschließend klickt er auf das „Plus-Symbol“ am oberen Bildrand 3. Das System verändert die Kartendarstellung in der Form, als das in den Bereich den der Benutzer mittig auf der Karte platziert hat hineingezoomt wird
Endzustand	Der Customer befindet sich auf der Startseite der Analyse und der Vergrößerungsgrad wurde entsprechend seiner Einstellungen verändert
Referenzierte Anforderungen	F14, F15, NF6, NF7, NF13

ID	AF24
Bezeichnung	Aus der Kartendarstellung herauszoomen
Akteure	Customer (angemeldet)
Anfangszustand	Der Customer befindet sich auf der Startseite der Analyse und die Kartendarstellung wird angezeigt
Ereignisfluss	1. Der Customer zieht mittels Doppelklick und anschließend gedrückter Maustaste die Kartendarstellung in die gewünschte Position 2. Anschließend klickt der Customer auf das „Minus-Symbol“ am linken oberen Bildrand 3. Das System verändert die Kartendarstellung in der Form, als dass aus dem Bereich den der Benutzer mittig platziert hat herausgezoomt wird
Endzustand	Der Benutzer befindet sich in der Kartenansicht in einer tieferen Zoomstufe
Referenzierte Anforderungen	NF13, NF14,NF15, NF24

ID	AF25
Bezeichnung	Informationen zu einer Zählspule anzeigen
Akteure	Customer
Anfangszustand	Der Customer befindet sich in der Analyseansicht und hat die Karte angezeigt
Ereignisfluss	1. Der Customer klickt auf eine der in der Kartendarstellung befindlichen Zählspulensymbole 2. Das System aktualisiert das Infopanel auf der rechten Seite mit den folgenden Informationen zur angeklickten Zählspule: „Zählspule“, „Anzahl PKWs“, „Schadstoffbelastung“, „Ampelanlage“, „Bahnhof“ und „Event“

Endzustand	Der Customer befindet sich in der Kartendarstellung und die Informationen zu der von ihm angeklickten Zählspule werden im Infopanel angezeigt
Referenzierte Anforderungen	NF12, NF13, F15, NF24

ID	AF26
Bezeichnung	Prognosen für eine Zählspule anzeigen lassen
Akteure	Customer
Anfangszustand	Der Customer befindet sich auf der Startseite der Analyse und die Kartendarstellung ist angezeigt
Ereignisfluss	1. Der Customer zoomt in die Kartendarstellung hinein und navigiert auf diese Weise zu der Zählspule seiner Wahl 2. (Gegebenenfalls klickt der Benutzer auf eines der Felder mit einer Zahl. In diesem Fall blendet das System eine Auswahl der zur Verfügung stehenden Zählspuren in diesem Bereich ein) 3. Der Customer klickt auf das gewünschte Zählspulensymbol 4. Das System lädt die Kartendarstellung neu, markiert die ausgewählte Zählspule und blendet die Diagramme mit den Darstellungen der kurz- und langfristigen Prognosen zu dieser Zählspule ein

Endzustand	Der Customer befindet sich in der Analyseansicht und die Darstellungen für die kurz- und langfristige Prognose sind eingeblendet auf der linken oberen Seite des Bildschirms
Referenzierte Anforderungen	F15, NF10, NF12, NF13

ID	AF27
Bezeichnung	Feinstaubbelastung anzeigen (Kann)
Akteure	Customer (angemeldet)
Anfangszustand	Der Customer befindet sich in der Startansicht und die Kartendarstellung ist geöffnet. Zudem ist unter den Widgets die Schadstoffbelastung aktiviert
Ereignisfluss	1. Der Customer navigiert auf eine Zählspule, für die er die Schadstoffbelastung berechnen möchte und klickt auf diese 2. Das System berechnet auf Grundlage der Zählwerte der jeweiligen Zählspule, der Verteilung der unterschiedlichen Verkehrsmittel und allgemeinen Werten zur Feinstaubbelastung im Raum Oldenburg den entsprechenden Wert 3. Das System verändert die Anzeige des Infopanels und zeigt dort die berechneten Werte an
Endzustand	Der Customer befindet sich in der Startansicht und die Kartendarstellung ist geöffnet. Das Infopanel auf der rechten Seite enthält die vom System berechneten Feinstaubwerte, der jeweiligen Zählspule.
Referenzierte Anforderungen	NF10, NF12, NF13

ID	AF28
Bezeichnung	Unfallgefahr anzeigen lassen (Kann)
Akteure	Customer (angemeldet)
Anfangszustand	Der Customer befindet sich auf der Startseite des Portals und es ist die Kartendarstellung angezeigt
Ereignisfluss	1. Der Customer navigiert zu der Zählspule, für die er die Unfallgefahr berechnen lassen möchte und klickt auf diese 2. Das System berechnet die Unfallgefahr auf Grundlage der Zählwerte der jeweiligen Zählspule, der Verteilung der Verkehrsmittel, der Anzahl von Fahrzeugen, die durchschnittlich auf diesem Abschnitt fahren und der Länge der Strecke 3. Das System aktualisiert das Infopanel auf der rechten Seite mit dem berechneten Wert
Endzustand	Der Customer befindet sich auf der Startseite der Analyse, die Kartendarstellung ist aufgerufen und das Infopanel zeigt den vom System berechneten Feinstaubwert für die jeweilige Zählspule an
Referenzierte Anforderungen	NF10, NF12, NF13

9.3 Anwendungsszenario

Der Use-Case, der erarbeitet wurde, definiert den funktionellen Umfang des Web-Portals, welches von der Projektgruppe RAPID erstellt wurde, um ein konkretes Problem zu lösen. Im Hintergrund wird die SAP-HANA Technologie als Datenbank und Instrument zur Berechnung verwendet. Anhand dieses Use-Cases wird klar ersichtlich, welche Funktionalitäten vom Programm zu erwarten und wer potentielle Nutzer sind. Der Use-Case sieht vor, dass der aktuelle Zeitpunkt der 26.03.2015 ist und ein Experte-User, beispielsweise der Stadt Oldenburg oder der VWG, eine Prognose des Verkehrsflusses für ein spezifisches Gebiet erhalten möchte. Seine konkrete Anforderungen sind wie folgt:

- Eine Vorhersage des zu erwartenden Verkehrs für einen kurzen Zeitraum, in diesem Fall für 30 Minuten in der Zukunft.
- Eine Vorhersage des zu erwartenden Verkehrs für einen langen Zeitraum. In diesem Fall handelt es sich um eine Prognose für über 100 Tage in der Zukunft.

Diese genannten Zeiträume werden im folgenden detailliert beschrieben und sollen mit Hilfe des SAP-Hana Portals eruiert und analysiert werden, um die gewonnenen Informationen zur Optimierung des städtischen Verkehrs zu nutzen. Der erste Fall, die kurzfristige Prognose, soll der Überprüfung einer Ampelschaltung dienen, um diese aufgrund von unerwarteten Gegebenheit aktualisiert zu können. Hierbei steht die Stauprävention an erster Stelle. Zunächst kann dafür der Prognosealgorithmus für kurzfristige Zeiträume, in diesem Fall 30 Minuten, genutzt werden, welcher einen Annäherungswert liefert. Um die vorhergesagten Werte dann eventuell bestätigen zu können ist es ebenfalls möglich diesen mit vergangenen Tagen zu vergleichen indem Vergangenheitswerte aus der Datenbank ausgelesen werden. Im zweiten Fall möchte der Nutzer die Daten einer bzw. mehrerer Zählspule einzeln für die Planung einer durchzuführenden Baustelle nutzen und braucht dafür eine Prognose des Verkehrsflusses für Mitte Dezember. Hierzu verwendet er die langfristige Prognose um sich diese Daten darzustellen zu lassen um einen Plan zu erarbeiten, der den zu erwartenden Verkehr entsprechend umleitet. Dafür wurde als Beispiel die Alexanderstraße in Oldenburg gewählt. Das Portal visualisiert dem User nun sämtliche relevante Daten.

9.4 Backend

In diesem Abschnitt wird das Backend der Darstellung genauer erläutert, dabei wird auf Codeigniter und SAP HANA sowie auf das Rollenkonzept eingegangen.

9.4.1 Codeigniter und SAP HANA

Das Backend stellt die Daten- und Funktionssicht bereit. Daten und Prognosealgorithmen werden aus der SAP HANA Datenbank geladen, hierzu wird mittels ODBC eine Datenbankverbindung hergestellt. Die Programmlogik des Portals wird mit dem PHP Framework Codeigniter verwirklicht, welches sich auf das Model-View-Controller Konzept stützt und dadurch eine bessere Übersichtlichkeit und Grundstruktur garantiert. Das Framework stellt zudem vorgefertigte Methoden bereit, um den Programmieraufwand zu verringern. Die Installation geschieht durch hochladen der Codeigniter-Dateien auf den Webserver, im Ordner „/application/config“ befinden sich die Konfigurations-Dateien, etwa zur Einrichtung der Datenbankverbindung, dem automatischen Laden von gewünschten Bibliotheken oder der Auswahl der Default-View zur Anpassung der Startseite. Die Programmierung der Model, Controller und View Dateien zur Einbindung des Layouts und Contents geschieht im entsprechenden Unterordner „model“, „controller“ oder „view“ des Ordners „application“. Eine generelle Abwandlung des MVC-Konzepts wird vorgenommen, indem auf die Verwendung von Models verzichtet wird. Die Programmlogik sowie die Abfrage und Verarbeitung der Daten geschieht im Controller. Dieser Schritt wird gewählt, da es sich um eine übersichtliche Anzahl an benötigten Funktionen und Seiten handelt, sodass keine doppelte Verwendung eigener Funktionen eintritt, auf die Verwendung vorgefertigter Funktionen jedoch nicht verzichtet werden soll. Von dieser Strategie wird beim Aufbau der Datenbankverbindung abgewichen. Das Framework stellt keine direkte Schnittstelle zur HANA-Datenbank bereit, da in der Regel MySQL Datenbanken verwendet werden. Um eine Verbindung zur HANA Datenbank aufzubauen wird ein Model „Hana“ erstellt.

Im Controller wird ein Abfragestring an das Model geschickt, diese Abfrage wird vom Model aufgenommen und ausgeführt, nachdem die Verbindung zur HANA hergestellt wurde. Die Rückgabewerte aus der Datenbank werden ausgegeben, anschließend wird die Datenbankverbindung geschlossen. Im Controller können die Rückgabewerte verarbeitet werden. Aufgrund vieler benötigter Datenbankabfragen wird das Model zur Herstellung der Datenbankverbindung in der Konfigurationsdatei „autoload.php“ automatisch beim Aufruf des Controllers geladen.

Da die aktuellen Daten nicht vorliegen wird eine Echtzeit-Simulation eingebunden, sodass sich der Benutzer im März 2015 befindet. Dies wird durch Anlegen einer Datenbanktabelle in der HANA realisiert, die einen Zeitstempel enthält, welcher die simulierte Echtzeit darstellt. Per Cronjob wird diese Zeit jede Minute aktualisiert. Sämtliche Funktionen greifen auf den Zeitstempel zu, indem dieser im Controller abgefragt und den Variablen zugewiesen wird. Der hier verwendete Zeitstempel kann problemlos durch einen realen Zeitstempel ausgetauscht werden, wodurch die Umstellung auf den Echtzeitbetrieb vorgenommen wird. Eine sekündliche Aktualisierung der Serverzeit ist nicht zu empfehlen, da die Zählspulendaten nicht in diesem Rhythmus erhoben werden. Als Beispiel für den Prozess der Datenbankabfrage und der Verarbeitung der Rückgabewerte dient die Anzeige der Uhrzeit in der kurzfristigen Prognose. Das Model wird einmalig angelegt, die Variable $queryString enthält die im Controller angegebene SQL-Abfrage. PHP bietet Methoden zur Herstellung der Verbindung per ODBC, die zur Herstellung der Verbindung mit der Datenbank und zur Ausführung des Abfragebefehls genutzt werden. Die Variable $queryExec enthält die Rückgabewerte, zur Nutzung der Werte im Controller werden diese ausgegeben. Abschließend wird die Datenbankverbindung geschlossen.

```
function query_hana($queryString)
{
        //connect to db
        $conn = odbc_connect("hana","RAPID_OPERATOR","password",
           SQL_CUR_USE_ODBC);
        //insert
        $queryExec = odbc_exec($conn, $queryString);
        return $queryExec;
        //close connection
        odbc_close($conn);
}
```

In der Konfigurationsdatei „autoload.php“ wird das Model angegeben, da die Funktionalität häufig genutzt wird und ein Laden des Models im Controller auf diese Weise nicht vor jeder Datenbankabfrage vorgenommen werden muss.

```
$autoload['model'] = array('Hana');
```

Der Controller „Analysis“ beinhaltet die Programmlogik der Analyse-Ansicht. Zur Darstellung der Uhrzeit im Frontend wird ein SQL-String an das Model übermittelt. Die Rückgabe wird in einem Array gespeichert und verarbeitet. Um die Uhrzeit und das Datum nicht im vollständigen String anzeigen zu müssen wird der Zeitstempel über die explode() PHP-Funktion mehrfach anhand von Trennzeichen geteilt und in Variablen gespeichert.

```
$queryString = 'SELECT * FROM RAPID_OPERATOR."SERVERTIME_FRONTEND"'
    ;
$query = $this->Hana->query_hana($queryString);
$row = odbc_fetch_array($query);
        $datetime = explode(" ",$row['TIMESTAMP']);
                $date = explode("-",$datetime[0]);
                        $year = $date[0];
                        $month = $date[1];
                        $day = $date[2];
                $time = explode(":",$datetime[1]);
                        $hour = $time[0];
                        $minute = $time[1];
                        $second = $time[2];

$data['hour'] = $hour;
$data['minute'] = $minute;
```

Damit die definierten Variablen im View genutzt werden können werden sie im Array $data mit der gewünschten Bezeichnung gespeichert. Am Ende des Controllers wird das View geladen, hier wird das Array ebenfalls übergeben, sodass die darin gespeicherten Werte verwendet werden können.

```
$this->load->view('header');
$this->load->view('navigation');
$this->load->view('analysis_view',$data);
$this->load->view('footer');
```

Die Ausgabe der Werte im View geschieht anhand der Bezeichnung im Array. Das View enthält die JavaScript-Funktion zur Erstellung eines Diagramms per Google Charts, die Uhrzeit wird in der kurzfristigen Prognose für den Ausgangszeitpunkt benötigt.

```
function drawChart() {
        var data = google.visualization.arrayToDataTable([
                ['Zeitpunkt', 'Anzahl'],
                ['<?= $hour; ?>:<?= $minute; ?>',  <?= $pred_0;
                   ?>],
                ['15 min',  <?= $pred_15; ?>],
                ['30 min',  <?= $pred_30; ?>],
                ['60 min',  <?= $pred_60; ?>]
        ]);
}
```

Neben der Uhrzeit sind weitere Variablen enthalten, die im Controller definiert wurden. Für das Beispiel soll hierauf an dieser Stelle nicht weiter eingegangen werden. Im Portal wird die kurzfristige Prognose abgerufen, indem eine Zählspule ausgewählt wird. Für den Zeitpunkt 0 wird die Uhrzeit ausgegeben.

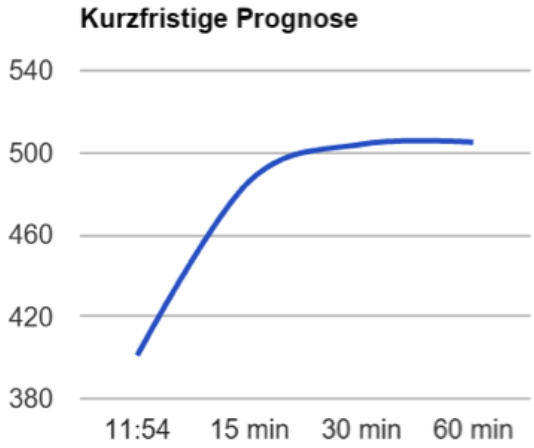


Abbildung 34: Diagramm zur Darstellung der kurzfristigen Prognose

Der Controller Analysis beinhaltet die gesamte Programmlogik und Datenverarbeitung der Analyse-Ansicht. Weitere Controller, die bei der Portalentwicklung erstellt wurden werden in der Folge aufgelistet:

- Admin Beinhaltet die Funktionen der Administrations-Oberfläche, hier werden Benutzer durch die Administratoren verwaltet.
- Cron Enthält die Funktionen, die von Cronjobs aufgerufen werden. Genutzte Funktionen sind die Aktualisierung der Serverzeit zur Echtzeit-Simulation sowie der Aufruf der Prognose-Prozeduren in der HANA.
- Import Der einmalige Import von Zählspulendaten, das Auffüllen dieser Daten sowie der ständige Import der Wetterdaten wird im Import-Controller definiert.
- User Die Funktionen zur Benutzerverwaltung sind hier zu finden. Neben der Login- und der Logout-Funktion sind die Passwort-vergessen- sowie die Registrierungs-Funktionalität enthalten.

9.4.2 Rollenkonzept

Die Festlegung von Verantwortlichkeiten für bestimmte Tätigkeiten innerhalb des Portals ist auf Grund der Sensibilität der zu verarbeiteten Daten essentiell wichtig. Für diesen Zweck wurden Rollen definiert, die bereits im Kapitel (Anforderungsdefinition) kurz erwähnt worden sind. Laut [SAP14] definiert eine Rolle „ eine Zusammenfassung von Privilegien, die z.B. Datenbankbenutzern, Benutzergruppen oder anderen Rollen zugewiesen werden können.“ Zudem erleichtert das Rollenkonzept dem Administrator die Administration und ermöglicht auch größeren Benutzergruppen eine gemeinschaftliche und effektive Arbeitsumgebung. Das Projekt RAPID definiert folgende Rollen:

- Rolle Customer

Die Rolle des Customers ist variabel gestaltet. Innerhalb des Projekts Rapid gibt es aktuell einen Hauptcustomer, die Stadt Oldenburg mit dem Status 1, sowie Projektgruppenmitglieder, die einen Vollzugriff auf die Analyseinhalte des Portals besitzen. Werden weitere Partner gewonnen, so können Bereiche individuell angepasst und nur die Bereiche sichtbar gemacht werden, die im Rahmen der Rechtevergabe vereinbart wurden.

- Rolle Administrator
  Der Administrator mit dem Status 2 besitzt eine gesonderte Stellung und kann jederzeit auf alle Inhalte des Portals sowohl im Analysebereich als auch im internen Administrationsbereich zugreifen. Dort wird es ihm zusätzlich ermöglicht, Customer zu aktivieren, zu deaktivieren oder zu löschen.

- Rolle Future Customer
  Ein Future Customer ist im Rahmen des Rollenkonzeptes ein Customer, der seine Registrierung bereits abgeschlossen hat und auf die Freigabe des Administrators wartet. Der Administrator muss eine eingehende Überprüfung durchführen, um die sensiblen Inhalte des Portals ausreichend zu schützen. Der Future Customer hat den Status 3.

Im Folgenden werden in Abbildung 35 allgemeine rollenspezifische Hauptfunktionalitäten des Portals anhand der dargestellten Rollen vorgestellt. Die Nutzung der organisationsabhängigen Portalfunktionalitäten hängt von der Freigabe des jeweiligen Customers ab.

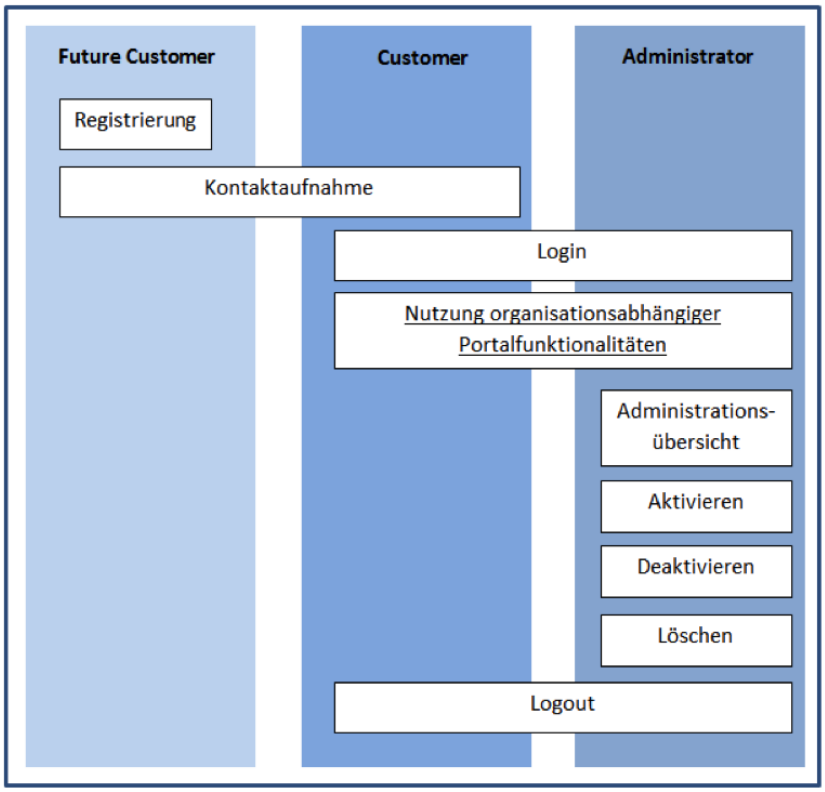


Abbildung 35: Rollenspezifische Hauptfunktionalitäten

Die zugrunde liegende Datenbankbasis für das Rollenkonzept umfasst 6 Columns und ist in der Definition der Tabelle USER_WEB aus Abbildung 36 zu erkennen. Neben der E-Mail Adresse, dem Passwort, dem Vornamen, dem Nachnamen und der Organisation, ist vor allem der Status entscheidend für die Rolle des Customers.

	Name	SQL Data Type	Di...	Column Store Data...	Key	Not Null	Default	Comment
1	EMAIL	VARCHAR	200	STRING	X(1)	X		
2	PASSWORD	VARCHAR	200	STRING				
3	FIRSTNAME	VARCHAR	100	STRING				
4	LASTNAME	VARCHAR	100	STRING				
5	ORGANIZATION	VARCHAR	100	STRING				
6	STATUS	INTEGER		INT				

Abbildung 36: Datenbank Tabellendefinition

Während der Registrierungsphase kann der Future Customer seine Daten im Formularfeld eingeben und absenden. Sobald der Administrator den Zugriff autorisiert hat, wird der Status anhand der Zugriffsrechte variabel gesetzt. Während des Logins wird in den Session-Informationen der interne Status aus der Datenbank geladen:

```
$queryString = 'SELECT STATUS FROM [DB] WHERE EMAIL = '$email' AND
    PASSWORD = '$password'';

$userdata = array('STATUS' => $row['STATUS'] [ ]);

$this -> session -> set_userdata($userdata);
```

Mittels Teilabfragen in den jeweiligen Portalbereichen können Inhalte für bestimmte Customer verborgen oder sichtbar gemacht werden:

```
if($this -> session -> userdata('STATUS') = 1)
```

De Abbildung 37 gibt einen Überblick über die aktuell registrierte Customer des Portals und deren Status.

EMAIL	PASSWORD	FIRSTNAME	LASTNAME	ORGANIZATION	STATUS
jannes@sp...		Jannes	Spekker	RAPID	2
ch.janssen...		Christian	JanÃen	RAPID Uni Oldenburg	2
alexander.s...		Alexander	Sandau	UniversitÃ¤t Oldenb...	1
kai.haenig...		Kai	Haenig	RAPID	2
kamiran.tiz...		Kamiran	Tizyani	RAPID	1
jannes_spek...		Jannes	Spekker	RAPID	1
olga.schwar...		Olga	Schwarz	Rapid	1
philipp.sch...		Philipp	Schumacher	CvO UniversitÃ¤t Old...	1
nils-steffen...		Nils	Worzyk	RAPID	1

Abbildung 37: Datenbank Tabelleneinträge

9.5 Frontend

In diesem Abschnitt wird auf die Darstellung des Frontends eingegangen, dazu gehören das Layout, die und die Visualisierung.

9.5.1 Layout

Ein visuell ansprechendes aber simples und gleichermaßen funktionales Layout wird mithilfe des CSS Frameworks Bootstrap geschaffen. Das Framework stellt vorgefertigte Klassen für die gängigsten HTML Elemente zur Verfügung und erleichtert dadurch die Gestaltung eines individualisierten Webdesigns. Das zu Projektbeginn erstellte Logo sowie die öffentliche Internetseite der Projektgruppe dienen als Vorlage für die genutzten Gestaltungselemente. Neben den Skript- und Darstellungssprachen HTML und CSS wird die JavaScript Bibliothek jQuery eingebunden, welche durch Bootstrap vorgegeben wird, allerdings auch Standard bei der Erstellung dynamischer Webseiten ist. Weitere APIs für die Visualisierung sind Leaflet zur Kartengenerierung und –manipulation und Google Charts zur Erstellung von Diagrammen. Die einzelnen Layoutkomponenten werden in einzelne logisch getrennte Abschnitte aufgeteilt und als View in Codeigniter angelegt. Die Abschnitte

werden grob in die HTML Bereiche Head, Body und Footer getrennt. Im Head Bereich werden die genutzten APIs und Stylesheet Dateien geladen, der Body Bereich beinhaltet die zur Darstellung verwendeten Elemente, hier werden außerdem die generierten Daten aus dem Controller geladen. Im Footer werden die geöffneten Tags des HTML-Konstrukts geschlossen.

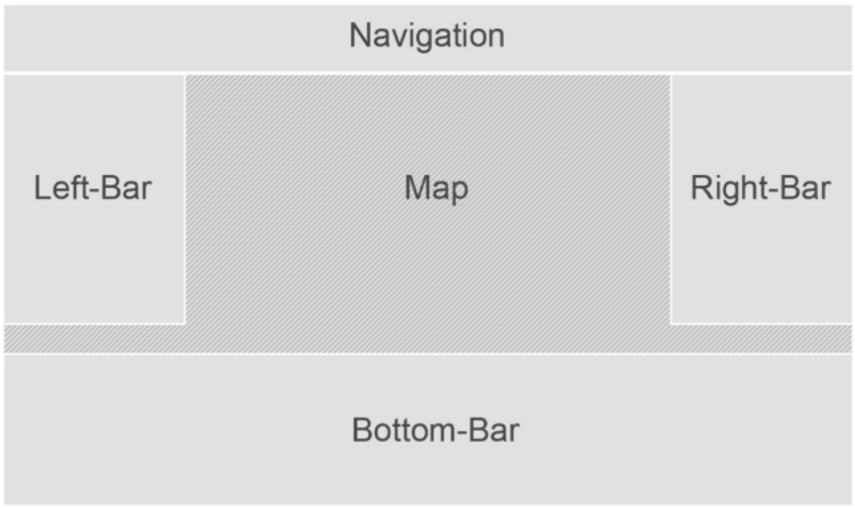


Abbildung 38: Anordnung der Portal-Elemente

Die einzelnen Komponenten für die Analyse werden als Widget eingebaut und können vom Benutzer entsprechend der benötigten Informationen ausgewählt werden. Die Benutzeroberfläche ist grob in fünf Bereiche eingeteilt. Die Karte ist zentraler Bestandteil und bleibt zu jeder Zeit sichtbar, hier werden die Objekte wie Zählspulen, Ampeln oder Buslinien eingeblendet. Im oberen Bereich ist die Navigationsleiste angeordnet, über die die Menüpunkte angesteuert werden. Auf der linken Seite werden Prognose-Diagramme eingeblendet, sobald eine Zählspule auf der Karte ausgewählt wird. Rechts wird die Uhrzeit und das Datum eingeblendet, sowie das Widget Symbol zur Auswahl der einzelnen Komponenten. Darunter wird eine kompakte Darstellung aller relevanten Daten zur Situationsanalyse und eine Legende zur Erläuterung der Symbole ausgegeben. Die weiteren Informationen sind in einer unten angeordneten Leiste zu finden, welche nur bei Auswahl entsprechender Widgets eingeblendet wird und über einen Button am unteren Bildrand ausgeblendet werden kann. Hier sind Daten zu Wetter, Events und Nahverkehr untergebracht.

9.5.2 User-Verwaltung

Das Kapitel User-Verwaltung behandelt die zentrale Interaktion des Users mit dem Portal. Im Folgenden werden die Funktionen „Registrieren“, „Login“, „Profilansicht“, „Logout“ und „Passwort vergessen“ beschrieben. Zudem wird ein Überblick über die Administrationsfunktion gegeben und Verwaltungsmöglichkeiten des Administrators dargestellt.

Registrierung

Bevor ein Customer die Funktionalität des Portals nutzen kann, muss eine Registrierung erfolgen. Über den Registrierungslink auf der Portalstartseite wird die Zuständige „Register_View“ geöffnet. Hier wird zunächst die E-Mail Adresse und das Passwort in zweifacher Ausfertigung eingegeben. Des Weiteren müssen der Vorname, Nachname und die betreffende Organisation, welche Zugriff auf das Portal wünscht, eingetragen werden. Auf Grund von Verschwiegenheitserklärungen und in Betracht auf der Einsicht von sensiblen Daten, sind diese Felder essentiell, da eine eingehende Prüfung nach absenden der Anfrage stattfindet. Der Customer muss zudem die Richtigkeit seiner Angaben bestätigen. In Abbildung 39 wird die Registrierungs-Übersicht dargestellt.

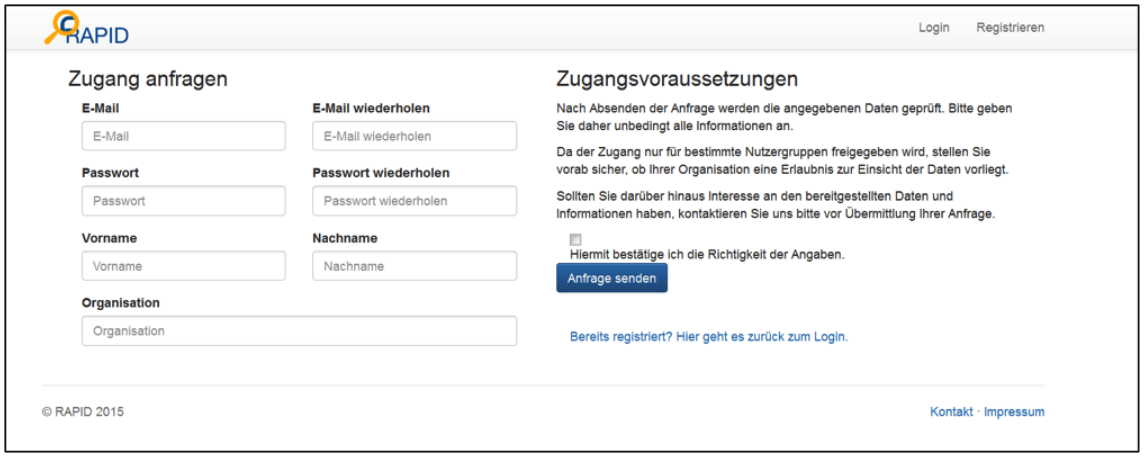


Abbildung 39: Registrierung

```
if(isset($_POST['check']))
```

Im Anschluss daran, werden die eingegebenen E-Mail Adressen und Passwörter auf Gleichheit überprüft. Ist dies nicht der Fall, wird im zuständigen Feld eine Fehlermeldung ausgegeben:

```
if($_POST['email']== $_POST['email2'])

   if($_POST['password']== $_POST['password2'])

      $error_msg;
```

Neben der Überprüfung der Benutzereingaben wird ein zusätzlicher Abgleich der E-Mail Adresse in der Datenbank vorgenommen. Ist ein Customer bereits registriert, so wird ihm dies mitgeteilt. Sind keine Eintragungen vorhanden, wird das Passwort aus Sicherheitsgründen mittels einer Hashmethode verschlüsselt und mit den übrigen Eingaben in die Datenbank eingetragen:

```
$password = do_hash($_POST['password']);
```

```
$queryString = 'INSERT INTO [DB] ($_POST['email'], $password,
   $_POST['firstname'],$_POST['lastname'], $_POST['organization'
   ])';
```

Zum Abschluss wird der nun registrierte Customer über die Register_Success_View auf die Portalstartseite geleitet.

Login

Das Login-Formular ist auf der Portalstartseite eingebettet und erbittet die im Registrierungsvorgang angegebene Adresse sowie das korrespondierende Passwort eines Customers. In Abbildung 40 ist der Login dargestellt.

Abbildung 40: Login

Die zuständige Funktion login() ist im Controller User implementiert. Im ersten Schritt überprüft die Funktion ob eine E-Mail Adresse und ein Passwort angegeben wurden:

```
if (isset($_POST['email'] and isset($_POST['password']))
```

Um die Eingaben mit den Eintragungen in der Datenbank zu vergleichen, wird zunächst eine SQL Abfrage mit der E-Mail Adresse sowie des Passworts generiert:

```
$queryString = 'SELECT * FROM [DB] WHERE EMAIL = '$email' AND
   PASSWORD = '$password''

$query = $this -> Hana -> query_hana($queryString);
```

In einer anschließenden WHILE Schleife muss zunächst der Account Status sowie die Länge des $queryString überprüft werden. Ist der Account Status nicht gleich 3 oder 0 so ist der Customer noch nicht autorisiert das Portal zu betreten. Ist die Länge des $queryString gleich 0, so ist der Customer noch nicht registriert. In beiden Fällen wird eine eindeutige Fehlermeldung angezeigt:

```
$num_rows = odbc_num_rows($query);

while ($rowCheck = odbc_fetch_array($queryCheck))

   if($rowCheck['STATUS'] !=3 and $rowCheck['STATUS']!=0)

      if($num_rows >0)
```

Wurde der Customer autorisiert, wird eine Session mit den eingetragenen Benutzerdaten angelegt:

```
$this -> session ->set_userdata($userdata);
```

Abschließend erfolgt eine Weiterleitung auf die Analyseseite des Portals.

Profilansicht

Die Profilansicht erlaubt dem Customer die Einsicht in seine bei der Registrierung hinterlegten Daten. Dabei wird der Vorname, Nachname, die Organisation und die E-Mail Adresse angezeigt. Zudem kann der Customer seine eingetragene Rolle unterhalb des Avatars einsehen. In Abbildung 41 wird die Profilansicht dargestellt. Aus Sicherheitsgründen wurde eine Bearbeitung der Daten deaktiviert.

Abbildung 41: Profil

Die zuständige Funktion profile() ist im Controller User implementiert. Zunächst werden die benötigten Informationen aus der Session des Customers geladen und an die User_Profile_View weitergegeben:

```
$firstname = $this -> session -> userdata('FIRSTNAME');

$lastname = $this -> session -> userdata('LASTNAME');

$email = $this -> session -> userdata('EMAIL');

$organization = $this -> session -> userdata('ORGANIZATION');

$condition = $this -> session -> userdata('STATUS');
```

Innerhalb der User-Profile-View werden die Informationen mit dem Profilformular verknüpft. Die Variable $condition lädt je nach Rolle des Customers ein anderes Avatarbild.

Logout

Der Logout regelt die Abmeldung des Customers vom Portal und ist im persönlichen Bereich des Customers in der Menüzeile eingebettet. Die Abbildung 42 wird die Logout-Möglichkeit aufgezeigt.

Abbildung 42: Logout

Die zuständige Funktion logout() ist im Controller User implementiert. Betätigt der Customer die Logoutfunktion, so wird im Wesentlichen die Session des Customers zerstört und der Customer zurück auf die Startseite des Portals geleitet:

```
this -> session -> sess_destroy();

redirect('/');
```

Passwort vergessen Funktion

Die „Passwort vergessen“ Funktion ist auf der Startseite des Portals eingebettet und bietet einem bereits registriertem Customer die Chance, ein vergessenes Passwort durch eine neues zu ersetzen. Dabei gibt der Customer seine registrierte E-Mail Adresse in der Reset_Password_View an und klickt auf den Link „Passwort zurücksetzen“. Die Abbildung 43 zeigt das Passwort zurücksetzen Formular.

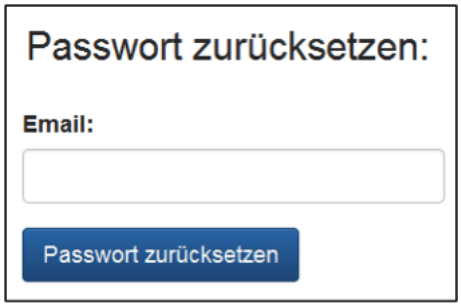


Abbildung 43: Passwort zurücksetzen

In einer automatisch generierten E-Mail findet der Customer einen einzigartigen Link, der aus Sicherheitsgründen die Verifizierung der E-Mail Adresse übernimmt. In Abbildung 44 wird das Grundgerüst der versendeten E-Mail dargestellt.

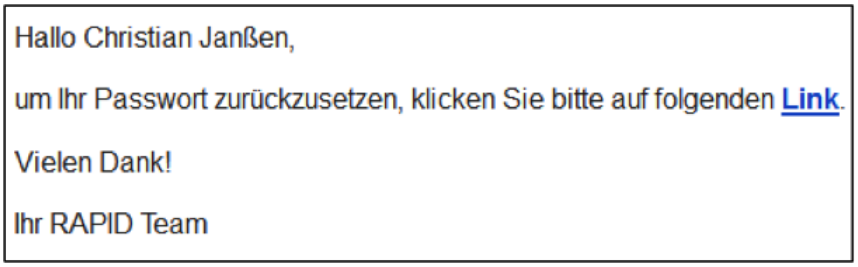


Abbildung 44: E-Mail mit einzigartigem Link

Betätigt der Customer den Link, so wird er auf die Update_Password_View geleitet. Hier ist die von ihm eingegebene E-Mail Adresse bereits hinterlegt. In den nachfolgenden Eingabefeldern wird nun das neue Passwort eingegeben. Durch betätigen des „Passwort erneuern“ Links wird das alte Passwort in der Datenbank ersetzt. Die Abbildung 45 gibt einen Überblick über das zuständige Update Formular.

Abbildung 45: Update Passwort

Die zuständige Funktion reset_password() ist im Controller User implementiert. Der Customer gibt zunächst seine E-Mail Adresse ein und betätigt den Link „Passwort zurücksetzten“. Die zuständige Funktion überprüft im ersten Schritt ob eine gültige E-Mail Adresse eingegeben wurde. Zudem darf das Eingabefeld nicht leer sein. Aus Validierungs- und Sicherheitstechnischen Gründen wird die eingegebene E-Mail verkürzt und auf SQL Einschüben überprüft:

```
if(isset($_POST['email']) && !empty($_POST['email']

    $email = set_rules('email','trim|valid_email|xss_clean');
```

Hat eine erfolgreiche Überprüfung der eingegebenen E-Mail Adresse stattgefunden, so wird die Hilfsfunktion email_exists() verwendet um festzustellen, ob die eingegebene E-Mail Adresse in der Datenbank hinterlegt ist:

```
$queryString = 'SELECT * FROM [DB] WHERE EMAIL = '$email'';

$resutl = $this -> Hana -> query_hana($queryString);

$num_rows = odbc_num_rows($result);

if($num_rows === 1)

   return TRUE;
```

Liefert die If-Abfrage ein positives Ergebnis, so wird die E-Mail Adresse an die Hilfsfunktion send_reset_password_email() weitergeletiet. Zunächst wird der Vor- und Nachname des Customers anhand der E-Mail Adresse aus der Datenbank gespeichert:

```
$queryStringName = 'SELECT FIRSTNAME, LASTNAME FROM [DB] WHERE
   EMAIL = '$email'';

$firstname = $row['FIRSTNAME']

$lastname = $row['LASTNAME'];
```

Um die E-Mail mit weiteren Informationen zu generieren, wird die interne CI Library EMAIL benötigt. Als erstes müssen E-Mail Informationen eingegeben werden:

```
$this -> load -> library('EMAIL');

$this -> email -> set_mailtype('html');

$this -> email -> from('RAPID');

$this -> email -> to($email);

$this -> email -> subject();
```

Die eigentliche Nachricht wird mit Hilfe von HTML Code generiert. Das Grundgerüst wurde bereits in Abbildung 44 dargestellt. Der einzigartige Link in der E-Mail besteht zum einen, aus der E-Mail Adresse des Customers und zum anderen aus einem zufällig generierten md5 kodierten Wert der an die Hilfsfunktion reset_password_form() übergeben wird. Innerhalb dieser Funktion wird die E-Mail Adresse und kodierte Wert mit dem in der Datenbank hinterlegten E-Mail Adresse des Customers verifiziert:

```
$verified= $this -> verify_reset_password_code($email, $email_code)
   ;

$queryString = 'SELECT FIRSTNAME, EMAIL FROM [DB] WHERE EMAIL = '
   $email' LIMIT 1';
```

```
$result = $this -> Hana -> query_hana($queryString);

$num_rows = odbc_num_rows($result);

if($num_rows === 1)

   return ($email_code == md5($this -> config -> item('salt') .
      $firstname)) ? true:false;
```

Bei einer erfolgreichen Verifizierung werden die Variablen $email und $email_code mittels des sha1 Hash Algorithmus aus Sicherheitsgründen verschlüsselt an die update_password_view übergeben:

```
$email_hash = sha1(($email . $email_code));
```

Das in Abbildung 45 vorgestellte Update Passwort Formular fordert den Customer auf, ein neues Passwort in zweifacher Ausfertigung anzugeben. Die Hilfsfunktion update_password() aus dem Controller User, validiert die Eingaben des Customers und führt zudem eine Sicherheitsüberprüfung hinsichtlich SQL Einschüben durch:

```
$securePass1 = set_rules('password', 'matches[password_conf]|
   xss_clean');

$securePass2 = set_rules('password_conf', 'xss_clean');
```

Zum Abschluss des Vorgangs muss das neue Passwort in der Datenbank gespeichert werden. Die integrierte Funktion update_passwort_db() verschlüsselt zunächst das vom Customer eingegebene Passwort mit dem bereits erwähnten sha1 Hash Algorithmus:

```
$password = sha1($this -> config -> item('salt') . $this -> input
   -> post('password'));
```

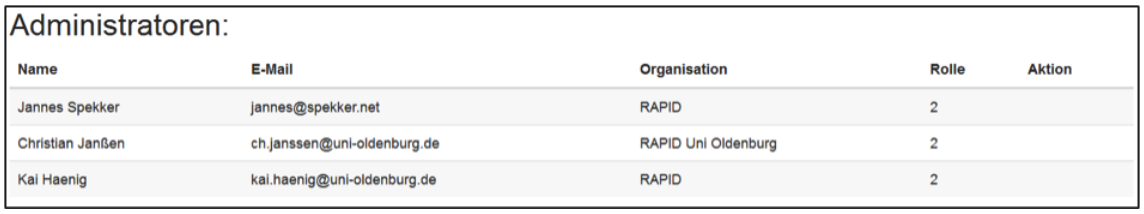
Das nun verschlüsselte Passwort wird in die Datenbank eingetragen und liefert bei Erfolg ein positives Ergebnis zurück:

```
$queryString = 'UPDATE [DB] SET PASSWORD = '$password' WHERE EMAIL
   = '$email'';

$query = $this -> Hana -> query_hana($queryString);

$num_rows = odbc_num_rows($query);

if($num_rows === 1)

   return TRUE;
```

Administration

Die Administrationsfunktion stellt im Wesentlichen die wichtigste Funktionalität innerhalb der User Verwaltung dar. Hier können Nutzergruppen organisationsspezifisch Rollen erhalten, die für bestimmte Bereiche des Portals und deren besonders geschützte Daten

wichtig erscheinen. Die Übersicht stellt zunächst die Administratoren (Rolle 2) des Portals in Abbildung 46 vor.

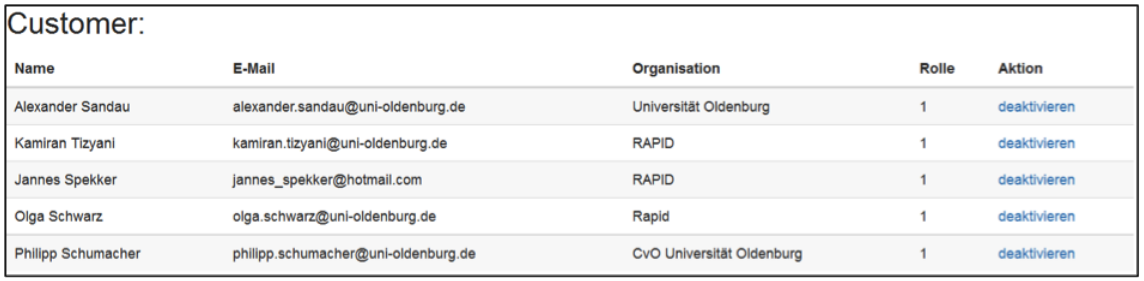


Administratoren:

Name	E-Mail	Organisation	Rolle	Aktion
Jannes Spekker	jannes@spekker.net	RAPID	2	
Christian Janßen	ch.janssen@uni-oldenburg.de	RAPID Uni Oldenburg	2	
Kai Haenig	kai.haenig@uni-oldenburg.de	RAPID	2	

Abbildung 46: Administration – Administratoren

Hier wurde bewusst auf Aktionen verzichtet, da die Administratoren untereinander die gleichen Rechte besitzen. Die nachfolgende Abbildung 47 zeigt Customer, die bereits einen Zugriff auf das Portal haben. Administratoren haben die Möglichkeit Customer zunächst zu deaktivieren, und ihnen den Zugriff auf das Portal zu verweigern. Weiterhin können Customer wieder aktiviert oder bei Ablauf der Partnerschaft gelöscht werden.



Customer:

Name	E-Mail	Organisation	Rolle	Aktion
Alexander Sandau	alexander.sandau@uni-oldenburg.de	Universität Oldenburg	1	deaktivieren
Kamiran Tizyani	kamiran.tizyani@uni-oldenburg.de	RAPID	1	deaktivieren
Jannes Spekker	jannes_spekker@hotmail.com	RAPID	1	deaktivieren
Olga Schwarz	olga.schwarz@uni-oldenburg.de	Rapid	1	deaktivieren
Philipp Schumacher	philipp.schumacher@uni-oldenburg.de	CvO Universität Oldenburg	1	deaktivieren

Abbildung 47: Administration – Customer

Die Future Customer Übersicht aus Abbildung 48 hilft dem Administrator festzustellen, welche neuen Nutzergruppen einen berechtigten, zukünftigen Zugang zum Portal erhalten oder nicht. Der Aktionsspielraum des Administrators beschränkt sich auf die Möglichkeit Future Customer freizuschalten oder bei einer nicht autorisierten Registrierung zu löschen. Eine weitere Kontaktaufnahme seitens des Administrators ist möglich.

Future Customer:

Name	E-Mail	Organisation	Rolle	Aktion
Nils Worzyk	nils-steffen@web.de	RAPID	3	freischalten löschen

Abbildung 48: Administration – Future Customer

Im Controller Admin sind die einzelnen Funktionen implementiert. Zunächst wird der Status überprüft und somit je nach Ergebnis die Administrationsübersicht in der Menüzeile des Portals angezeigt:

```
if($this -> session -> userdata('status') != 2)

   redirect(base_url() . 'login.php');
```

Die einzelnen Übersichten werden mit Hilfe von HTML Code generiert und durch SQL Abfragen mit Inhalt aus der Datenbank gefüllt:

```
$queryString = 'SELECT * FROM [DB] WHERE STATUS = 1';

$query = $this -> Hana -> query_hana($queryString);

while($row = odbc_fetch_array($query)

   $content .= $row['FIRSTNAME'], $row['LASTNAME'], $row['EMAIL']
      [..];
```

Mit Hilfe der Hilfsfunktionen activate(), deactivate() und delete(), wird der Handlungsspielraum des Administrators festgelegt.

9.5.3 Visualisierung

Wie im Abschnitt Backend erläutert dient der Controller „Analysis“ zur Entwicklung der Analyse-Funktionalität im Web-Portal, dieser enthält die Programmlogik und sorgt für die Verarbeitung der abgefragten Daten. Ebenfalls wird die Einbettung in HTML Elemente und die Belegung von JavaScript Variablen zur Darstellung von Objekten vorgenommen. Ein View bindet die nur für diesen Bereich genutzten JavaScript Dateien und APIs ein, außerdem werden die im Controller vorbereiteten HTML sowie JavaScript Variablen und Elemente eingefügt. Der Controller enthält die Funktionen „index“ und „widgets“, wobei die Widgets-Funktion lediglich der Verarbeitung eines HTML-Formulars dient. Die Index-Funktion bereitet die gesamte Darstellung des Navigationspunktes Analyse vor. Der Aufbau der Funktion wird im Folgenden erläutert, im Anschluss daran wird auf die einzelnen Funktionalitäten der Analyse-Seite eingegangen, indem zunächst die Benutzersicht und anschließend die Funktionssicht erläutert werden. Zunächst wird geprüft, ob die Session des Benutzers aktiv ist, er also im Portal eingeloggt ist. Ist dies nicht der Fall, wird der Benutzer auf die Login-Seite weitergeleitet, ansonsten wird der Controller geladen und die Seite entsprechend der ausgewählten Widgets aufgebaut. Um Datum und Uhrzeit in Datenbankabfragen und der Zeitanzeige korrekt und einheitlich zu verwenden wird der Zeitstempel abgefragt und durch den PHP explode-Befehl so zerlegt, dass die einzelnen Bestandteile in Variablen gespeichert werden. Daran anschließend wird abgefragt, welche Widgets vom Benutzer ausgewählt worden sind. Vorab werden Variablen für jedes Widget erstellt und mit dem Wert 0 belegt, beispielsweise $w_weather für die Wetteranzeige oder $w_spool für die Zählspulenanzeige. Eine Switch-Case Abfrage gleicht den Rückgabewert mit vorhandenen Widget-IDs ab, stimmen die Bedingungen überein wird die Variable auf den Wert 1 gesetzt. Liegt für ein Widget keine Auswahl vor, bleibt der Wert der Variable bei 0.

```
$w_weather = 0;
```

```
$w_event = 0;
$w_bus = 0;
$w_train = 0;
$w_pollutant = 0;
$w_traffic_signal = 0;
$w_spool = 0;
$queryString = 'SELECT * FROM RAPID_OPERATOR."USER_WIDGET" WHERE
   USER = \''.$this->session->userdata('EMAIL').'\'';
$query = $this->Hana->query_hana($queryString);
while($row = odbc_fetch_array($query) ) {
        switch($row['WIDGET']) {
                case $row['WIDGET'] == 1:
                        $w_weather = 1;
                        break;
                case $row['WIDGET'] == 2:
                        $w_event = 1;
                        break;
                case $row['WIDGET'] == 3:
                        $w_bus = 1;
                        break;
                case $row['WIDGET'] == 4:
                        $w_train = 1;
                        break;
                case $row['WIDGET'] == 5:
                        $w_pollutant = 1;
                        break;
                case $row['WIDGET'] == 6:
                        $w_traffic_signal = 1;
                        break;
                case $row['WIDGET'] == 7:
                        $w_spool = 1;
                        break;
        }
}
```

Vor dem Funktionsteil der jeweiligen Widgets im Controller wird eine If-Abfrage gemacht. Ist der Wert der Variable 1 wird der Inhalt der If-Abfrage abgehandelt und die Anzeige der Funktionalität vorbereitet. Dieses Vorgehen dient zum einen dem spezifischen Abruf der ausgewählten Seiteninhalte und verkürzt außerdem die Laufzeit des Skripts, indem dieses nur relevante Daten abfragt und vorbereitet. Auf die einzelnen auszuwählenden Bestandteile der Analyseansicht wird im Folgenden eingegangen.

Widgets

Zur spezifischeren Auswertung der vorliegenden Daten können die individuell gewünschten Parameter über die Widget Funktionalität aus- und abgewählt werden. Klickt der Benutzer im Portal auf das Widget Symbol wird ein Modal-Fenster in der Mitte des Browsers geöffnet, in einem Formular werden die in der Datenbank angelegten Widgets zur Auswahl per Checkbox ausgegeben. Es wird in einem weiteren Abruf geprüft, ob der Benutzer das

jeweilige Widget bereits ausgewählt hat. Ist dies der Fall, wird das Checkbox Element ausgewählt. Durch klicken des Bestätigen-Buttons wird das Formular abgesendet und die Seite entsprechend der gewählten Änderungen geladen. Die Aktualisierung der Datenbanktabelle geschieht in einem Zwischenschritt. Dieser und der Aufbau der Tabellen werden im nächsten Abschnitt dargestellt.

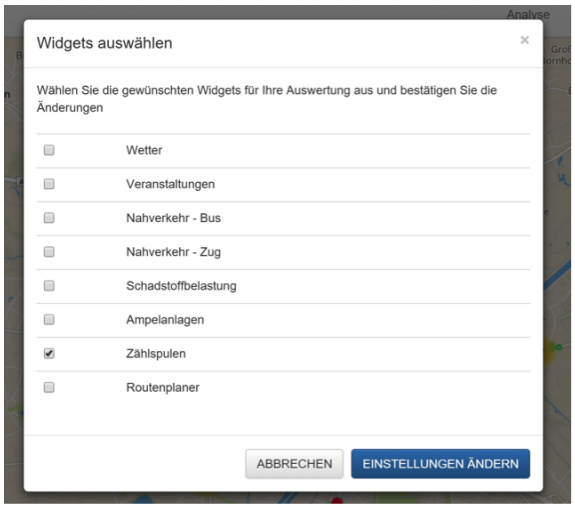


Abbildung 49: Modal Fenster zur Auswahl der Widgets

Die Datenbanktabelle WIDGETS enthält die verfügbaren Widgets, zu jedem Datensatz ist eine ID als Primärschlüsselattribut und der Name enthalten. Die Tabelle USER_WIDGETS ordnet die Widgets den Benutzern zu, indem die E-Mail des Benutzers und die ID des ausgewählten Widgets gespeichert werden. Die Aktualisierung geschieht, indem bei Absenden des Formulars die Funktion widgets im Analysis Controller aufgerufen wird. Die ausgewählten Checkbox-Elemente werden per GET Variable übergeben und die benötigte E-Mail-Adresse des Benutzers wird über gespeicherten Session Daten abgerufen. Die widgets Funktion löscht im ersten Schritt sämtliche vorhandenen Datenbankeinträge aus der Tabelle USER_WIDGET, die der E-Mail-Adresse zugeordnet sind. Im zweiten Schritt wird für jedes ausgewählte Checkbox-Element ein Insert Statement mit der Widget-ID aus der GET Variable und der E-Mail-Adresse des Benutzers ausgeführt. Damit der Benutzer zurück in die Analyse Ansicht gelangt wird im letzten Schritt eine Weiterleitung auf diese Seite eingerichtet.

```
public function widgets()
{
        if($this->session->userdata('logged_in') != TRUE) {
                redirect(base_url().'index.php/user/login');
        }
        else {
                $queryString = 'DELETE FROM "RAPID_OPERATOR".
                   USER_WIDGET WHERE USER = \''.$this->session->
                   userdata('EMAIL').'\'';
```

```
                $query = $this->Hana->query_hana($queryString);
                foreach($_GET as $wid)
                {
                        $queryString = 'INSERT INTO "RAPID_OPERATOR
                            ".USER_WIDGET (USER, WIDGET) VALUES (\''
                            .$this->session->userdata('EMAIL').'\','
                            .$wid.')';
                        $query = $this->Hana->query_hana(
                            $queryString);
                }
                redirect(base_url().'index.php/analysis');
        }
}
```

Map

Die Basis der Visualisierung bildet eine Karte mit dem Zentrum Oldenburgs als Ausgangspunkt der Betrachtung. Zwar liegen die zur Zeichnung einer Karte benötigten OSM Daten vollständig vor und sind in die HANA importiert worden, jedoch wird die Darstellung der Karte mithilfe der Leaflet API realisiert. Die API hat den Vorteil, dass bereits einige Bibliotheken zur Hervorhebung von Objekten, Zeichnung von Vektoren und weiteren grafischen Manipulationsoperationen vorhanden sind. Des Weiteren handelt es sich bei Leaflet, anders als beispielsweise Google Maps, um eine Open Source API, die auch kommerziell in vollem Umfang kostenlos genutzt werden kann. Die API wird durch die Mapbox API um weitere Funktionen erweitert. Die in der HANA vorliegenden erhobenen Datensätze werden mit Längen- und Breitengrad oder OSM IDs versehen, wodurch eine einfache Verknüpfung der Datenbasis mit den Kartendaten hergestellt werden kann und die Visualisierung ermöglicht wird. Die JavaScript und die Stylesheet Dateien von Leaflet werden im Head Bereich des Portals eingebunden, im Analysis Controller werden die angefragten Daten geladen und zugeordnet, abschließend werden im Analysis View die Informationen der Karte hinzugefügt. Die Karte wird nicht als Widget eingebunden, sie ist jederzeit sichtbar. Je nach gewähltem Widget werden jedoch unterschiedliche Objekte auf der Karte dargestellt. Der Benutzer kann in die Karte hereinzoomen, außerdem ist es per Maussteuerung möglich, sich in der Karte zu bewegen. Da sich die Ansicht auf das Stadtgebiet Oldenburg beschränken soll, sind für die Zoom-Funktionalität eine Minimal- und eine Maximalstufe festgelegt worden. Die Konfiguration der Karte geschieht per JavaScript im View. Hierbei wird ein HTML-DIV-Element benannt, in welchem die Karte dargestellt wird. Darüber hinaus werden das verwendete Kartentemplate, Copyrights und die API-Keys angegeben.

```
var map = L.map('map').setView([<?= $map_lat; ?>, <?= $map_long;
    ?>], <?= $map_zoom; ?>);
L.mapbox.accessToken = 'pk.eyJ1IjoiamFubmVzcmFwaWQiLCJhIjoiNTg5ODM
3MzQyNDkxNjA3Y2JkZDUyYzQ1N2EzMmQxMmUifQ.IRVr25dyzcKrppZDxr-61Q';
L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?
    access_token=
pk.eyJ1IjoiamFubmVzcmFwaWQiLCJhIjoiNTg5ODM3MzQyNDkxNjA3Y2Jk
```

```
ZDUyYzQ1N2EzMmQxMmUifQ.IRVr25dyzcKrppZDxr-61Q', {
        attribution: 'Map data &copy; <a href="http://openstreetmap
            .org">OpenStreetMap</a> contributors, ' +
                        '<a href="http://creativecommons.org/
                            licenses/by-sa/2.0/">CC-BY-SA</a>, ' +
                        'Imagery <a href="http://mapbox.com">Mapbox
                            </a>',
        id: 'mapbox.streets',
        minZoom: 13,
        maxZoom: 18
}).addTo(map);
```

Anstelle von statischen Koordinaten und einer Standard-Zoomstufe werden Variablen eingesetzt, welche das Zentrum der Karte beim Seitenaufruf definieren. Die angegebenen Variablen werden im Controller definiert. Dies hat den Vorteil, dass bei Auswahl von Zählspulen oder anderen Elementen per Mausklick das Kartenzentrum auf die ausgewählten Elemente geändert werden kann. Als Standardwerte für die Variablen sind die Koordinaten des Stadtzentrums angegeben, wird beispielsweise eine Zählspule ausgewählt werden die Koordinaten anhand von GET-Variablen über die URL übergeben anstelle der Standardwerte eingesetzt, auch die Zoomstufe wird in diesem Fall auf eine nähere Ansicht geändert.

```
if(!isset($_GET['lat']) or !isset($_GET['long']))
{
        $map_lat = '53.14553';
        $map_long = '8.20619';
        $map_zoom = '13';
}
else
{
        $map_lat = $_GET['lat'];
        $map_long = $_GET['long'];
        $map_zoom = '18';
}
$data['map_lat'] = $map_lat;
$data['map_long'] = $map_long;
$data['map_zoom'] = $map_zoom;
```

Zur Darstellung von Zählspulen, Ampeln, Buslinien oder anderen Objekten werden von Leaflet vordefinierte Elemente eingebunden. Zur Darstellung von Punkten werden Marker verwendet. Einzelne, statische Marker werden direkt per JavaScript im View definiert, andere werden im Controller mit Hilfe von Schleifen zu Variablen zugewiesen. Zur Einbindung der Marker wird in einer JavaScript Variable der Leaflet-Befehl L.marker() aufgerufen, als erster Parameter werden die Koordinaten angegeben, gefolgt von dem gewünschten Marker-Symbol. Der Zusatz „.addTo(map)“ gibt wiederum das Ziel an, zu welchem das Element hinzugefügt werden soll.

```
var icon_station = L.icon({
        iconUrl: '/html/web/ci/assets/images/icon_station.png',
```

```
        iconSize: [32, 32],
        iconAnchor: [32, 32],
        popupAnchor: [-3, -50]
});

var station = L.marker([53.1437084, 8.2225446], {icon: icon_station
    }).addTo(map);
```

Zur Zusammenfassung mehrerer Marker können Markercluster erstellt werden. Anstelle der Marker wird ein Kreis angezeigt, der die Anzahl der Marker angibt, die in einem Bereich vorhanden sind. Wenn in die Karte hineingezoomt wird oder auf ein Cluster geklickt wird, spalten sich diese in kleinere Cluster oder einzelne Marker auf. Hierzu wird im JavaScript zunächst ein Markercluster-Element erstellt, im Anschluss werden die einzelnen Marker dem definierten Cluster zugewiesen. Bei Mouseover über ein Markercluster werden die in dem Cluster zusammengefassten Marker von einem Polygon umrandet.

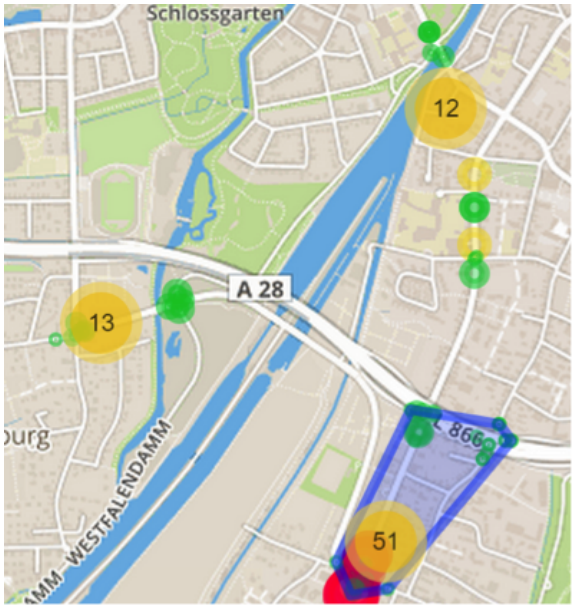


Abbildung 50: Markercluster zu Zählspulen in der Karte

Weitere im Projektverlauf verwendete Elemente sind Linien und Kreise. Die Erstellung wird synonym zum Marker-Element vorgenommen. Auf optionale Angaben zur Gestaltung wird in den Abschnitten eingegangen, in denen die Elemente verwendet werden.

Zählspulen (Verkehrsdaten)

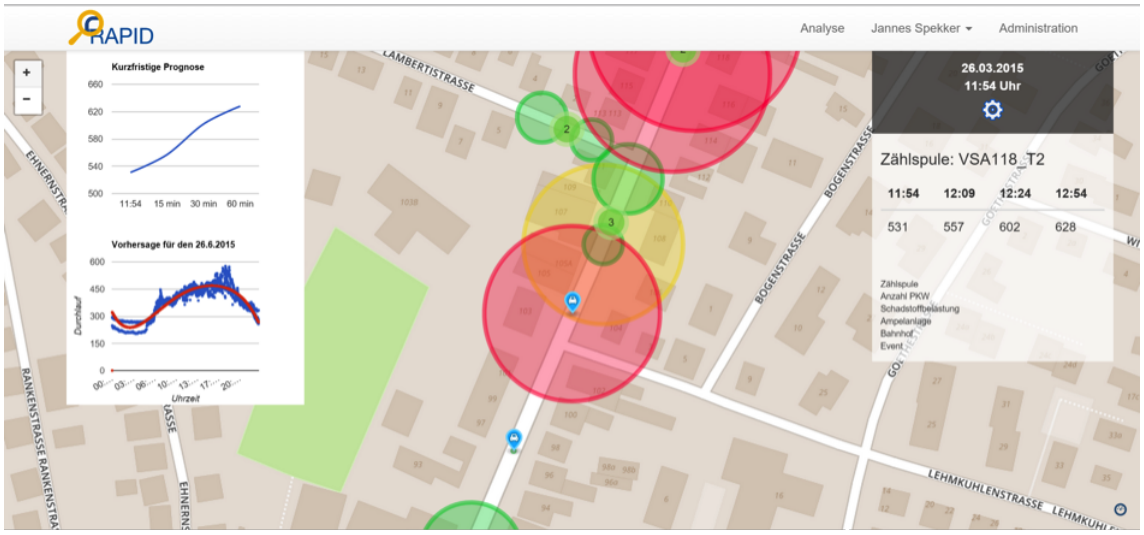


Abbildung 51: Heatmap

Dem Benutzer wird auf der Karte eine Heatmap bereitgestellt, welche die aktuelle Verkehrssituation übersichtlich darstellt. Durch heranzoomen in die Karte oder klicken auf ein Markercluster kann der zu analysierende Bereich ausgewählt werden. In der höchsten Zoomstufe können die Zählspulen über die Marker-Elemente ausgewählt werden. Die Seite wird neu geladen, im Zentrum der Karte befindet sich die ausgewählte Zählspule. An der linken Seite befinden sich die Diagramme zur kurzfristigen und langfristigen Prognose. Die Prognosewerte werden angezeigt, wenn der Benutzer den Mauszeiger über den Graphen bewegt. Auf der rechten Seite werden die Ergebnisse unterhalb der Zeitanzeige zusammengefasst. Hier werden der Name der Zählspule, relevante Prognosewerte und Informationen zu gewählten Widgets angezeigt. Im Controller werden nach Prüfung, ob das Widget ausgewählt wurde, die Angaben Name, Längen- und Breitengrad der Zählspule aus der Datenbanktabelle ZAEHLSPULEN_BASE abgerufen. Für jede Zählspule wird ein Marker-Element für die Karte erstellt, die jeweils dem Markercluster-Element „markers“ zugewiesen werden. Zur Erstellung der Heatmap werden im nächsten Schritt Zählspulenwerte für den aktuellen Zeitstempel aus der Datenbanktabelle ZAEHLSPULEN abgefragt. Die Heatmap ergibt sich aus Kreis-Elementen, die sich je nach Anzahl der erfassten Fahrzeuge in ihrem Radius unterscheiden. Der Controller weist der Variable zaehls_circle den JavaScript-String zur Erstellung des Elements zu, der Inhalt der Variable wird an das View übergeben und dort aufgerufen. Damit die Darstellung übersichtlich bleibt darf der Kreisradius nicht zu groß sein, daher wird eine Variable „circle_size“ mit dem Wert für die Zählspule belegt und durch einen Wert, es wird nach einigen Tests der Wert 15 gewählt, dividiert. Für die Erstellung der Kreis-Elemente wird die Leaflet-JavaScript-Funktion L.circle() genutzt, neben der Angabe der Koordinaten und des Kreisradius wird die Darstellung festgelegt, indem die Farbe und die relative Deckkraft des Elements definiert werden. Der Vorgang wird in eine Schleife eingebettet, sodass die Abfrage des Zählspulenwerts und die Erstellung des Kreis-Elements für jede Zählspule wiederholt

werden. Zur Erstellung der Diagramme wird Google Charts verwendet. Die API-Dateien werden im Header-View eingebunden. Die Werte für die kurzfristige Prognose liegen in der Datenbanktabelle PREDICTION in den Spalten 15_MINUTES, 30_MINUTES und 60_MINUTES vor. Nach Abfrage der Werte werden diese Variablen zugewiesen, die an das View übergeben werden. Im View werden die Variablen in die JavaScript Funktion zur Erstellung des Diagramms eingefügt.

```
<script type="text/javascript">
      google.setOnLoadCallback(drawChart);
      function drawChart() {
        var data = google.visualization.arrayToDataTable([
          ['Zeitpunkt', 'Anzahl'],
          ['<?= $hour; ?>:<?= $minute; ?>',  <?= $pred_0; ?>],
          ['15 min',  <?= $pred_15; ?>],
          ['30 min',  <?= $pred_30; ?>],
          ['60 min',  <?= $pred_60; ?>]
        ]);

        var options = {
                        title : 'Kurzfristige Prognose',
          curveType: 'function',
          legend: { position: 'none' }
        };

        var chart = new google.visualization.LineChart(document.
           getElementById('prog_chart'));

        chart.draw(data, options);
      }
</script>
```

Die Langfristige Prognose wird synonym zur kurzfristigen Prognose implementiert. Prognosewerte liegen in der Datenbanktabelle PREDVAL vor. Da bei der Berechnung die Wochentage in Gruppen unterteilt werden muss eine Switch-Case Abfrage im Controller ergänzt werden. Zunächst wird der aktuelle Wochentag ermittelt, im Anschluss wird je nach vorliegendem Tag die korrekte Datenbanktabelle gewählt.

Events

Das Widget Veranstaltungen ergänzt die Karte um Marker-Symbole an den Orten, an denen am aktuellen Tag Events stattfinden, außerdem wird eine Tabelle im Frontend eingeblendet, die Informationen zur Veranstaltung enthält. Das Widget greift auf die Event-Tabelle in der Datenbank zu, sie enthält Informationen zu sämtlichen Veranstaltungen, die im Stadtgebiet Oldenburg ausgerichtet werden. Im Controller wird die Abfrage gesendet, die den Zeitraum anhand der LIKE Bedingung eingrenzt. Der Variable event_marker wird der JavaScript-String zur Erstellung des Markers zugewiesen und an das View übermittelt. Das genutzte individuelle Kalender-Icon wird vorab auf den FTP-Server geladen und im View eingebunden.

```
if($w_event == 1)
{
        $queryString = 'SELECT * FROM "RAPID_OPERATOR"."Events"
           WHERE "Datum_Uhrzeit_von" LIKE \''.$day.'.'.$month.'%\''
           ;
        $query = $this->Hana->query_hana($queryString);
        $event_marker = '';
        $event_i = 0;
        while($row = odbc_fetch_array($query) ) {
                $event_marker .= 'event'.$event_i.' = L.marker(['.
                   $row['LATITUDE'].', '.$row['LONGITUDE'].'], {
                   icon: icon_event}).addTo(map),';
                $event_i++;
        }
        $event_marker=substr($event_marker, 0, -1);
        $data['event_marker'] = $event_marker;
}
```

Die Tabelle wird mit den vorgefertigten Bootstrap-CSS-Klassen erstellt, indem die Klassen table und table-condensed zugewiesen werden. Durch die Klasse table-condensed werden Abstände zwischen den Tabellenrändern zur besseren Darstellung komprimiert. Die Zeilen der Tabelle werden nacheinander mit den Rückgabewerten der Datenbankabfrage gefüllt.

Bus

Wählt der Benutzer das Widget „Nahverkehr-Bus“, wird der Liniennetzplan auf der Karte eingeblendet. Die einzelnen Linien werden dabei farblich unterschieden. Die aktuelle (SOLL-)Position der Busse wird durch Bus-Icons angezeigt.

Zählspulen (Verkehrsdaten)

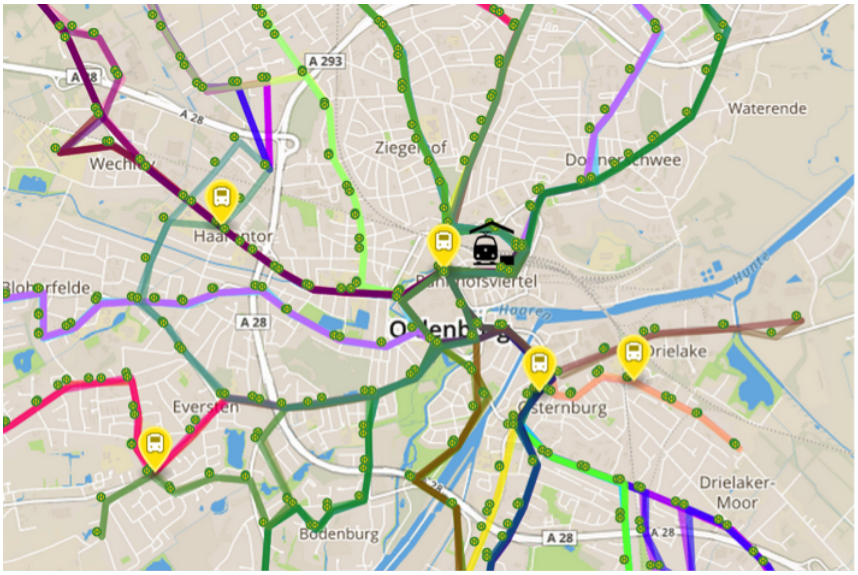


Abbildung 52: Darstellung des Liniennetzplans im Frontend

Wie bereits im Abschnitt Datenverständnis und Datenvorbereitung erläutert, beinhaltet die Datenbanktabelle BUS_BASE die Datensätze zur Definition des Liniennetzplans, die Tabelle BUS_TIMETAMBLE enthält Angaben zu Abfahrtzeiten. Im Controller wird die Tabelle BUS_BASE abgerufen, geordnet nach Linien-ID und Position, sodass eine korrekte Ordnung für die Darstellung vorliegt. Im ersten Schritt werden die Marker-Elemente aller Bushaltestellen erstellt. Der String wird der Variable bus_stops zugewiesen. Die Koordinaten der Bushaltestellen werden einer Variable der entsprechenden Buslinie zugewiesen. Die Leaflet-Funktion L.polyline() bedient sich dieser Variablen und verbindet die darin enthaltenen Koordinaten miteinander, sodass eine Linie entsteht. Die optionale „color“-Angabe sorgt für die farbliche Abgrenzung der Linien. Im letzten Schritt werden aus der Tabelle BUS_TIMETABLE diejenigen Einträge abgerufen, die für das Attribut TIME den Wert des jetzigen Zeitpunktes enthalten. Es werden für die Rückgabewerte Marker mit einem Bus-Icon erstellt.

Bahn

Der Bahnhof wird als Marker im JavaScript Bereich des Views definiert. Die eingehenden und ausgehenden Zugverbindungen werden in einer Tabelle dargestellt. Zur besseren Übersicht werden die nächsten vier Verbindungen ausgehend vom aktuellen Zeitpunkt abgerufen und dargestellt. Die verwendete Datenbanktabelle lautet ZUGVERBINDUNG und enthält die Abfahrtszeit, die Art des Zuges und die Fahrtrichtung.

Wetter

Dem Benutzer wird ein Diagramm angezeigt, welches die Entwicklung der Luft- und der Bodentemperatur im Tagesverlauf darstellt. Für die Wetterdaten-Ansicht werden die letzten bekannten Werte herangezogen, sodass diese nie länger als einen Tag zurück liegen.

Eine Prognose des Wetterverlaufs wird nicht vorgenommen, da hierfür keine verlässlichen Methoden zur Verfügung stehen. Der Einfluss von Wetterfaktoren auf den Straßenverkehr kann daher allein durch Beobachtung der Werte im Zeitverlauf analysiert werden. Die im Abschnitt Datenvorbereitung erstellte Datenbanktabelle dient als Datenbasis für die Abfrage. Die Erstellung des Wetterdaten-Diagramms gleicht der oben beschriebenen Generierung von Prognose-Diagrammen.

Schadstoffbelastung

Für die Schadstoffbelastung fehlen verlässliche Werte, da eine valide Berechnung anhand der vorliegenden Datenbasis nicht möglich ist. Im Portal wird die Darstellung der Schadstoffbelastung an den Zählspulen vorbereitet, indem eine farbliche Abstufung eingebaut wird, die sich an zu definierenden Grenzwerten orientiert. Die Stufen unterscheiden sich farblich in grüner, gelber und roter Darstellung, wobei grün für einen unkritischen Wert steht, der sich relativ weit unterhalb der Schadstoffgrenze befindet. Im kritischen Bereich, nahe der Schadstoffgrenze wird der Kreis gelb dargestellt. Sobald der Grenzwert überschritten wird färbt sich der Kreis rot. Die Grenzwerte sind im Controller ohne großen Arbeitsaufwand nachzupflegen.

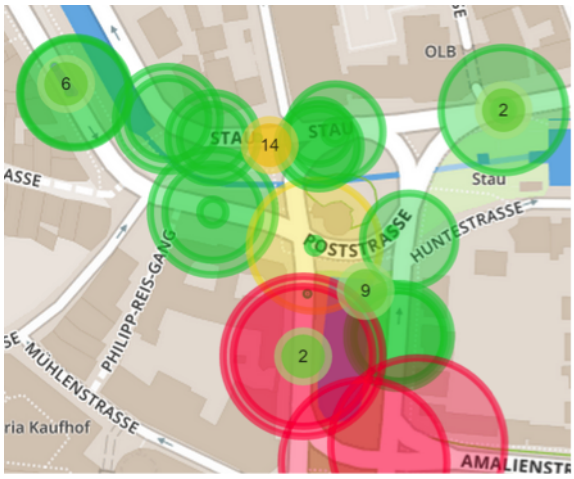


Abbildung 53: Farbliche Abgrenzung der Zählspulen zur Einordnung in Schadstoffklassen

10 Evaluation

Die Grundidee für die einjährige Projektgruppe „RAPID“ lag darin, eine intelligente Plattform für Mobilitätsdaten zu realisieren. Eine weitere Vorgabe war es, dass diese Plattform auf die In-Memory Technologie zurückgreift, um die Informationen dem Customer in Echtzeit zur Verfügung stellen zu können. Das verwendete In-Memory Database System für die Projektgruppe war ein SAP HANA System, welches durch die Abteilung VLBA bereitgestellt wurde, die sich wiederum an der Otto-von-Guericke Universität Magdeburg Serverkapazitäten angemietet haben.

Um einen Überblick über verschiedene Themen zu bekommen, die für die Umsetzung der Rahmenidee wichtig sein könnten, wurden bis zum 09.03.2015 Seminararbeiten von den, zu diesem Zeitpunkt noch 9 Teilnehmern, verfasst. Die meisten davon hatten technische Themen, wie beispielsweise

- *Entwicklungsumgebungen und Frameworks um SAP HANA*

oder

- *Visual Analytics: Konzepte, Methoden und Systeme*

die sich mit verschiedenen softwaretechnischen Aspekten für die Datenverarbeitung beschäftigten. Aber auch das Projektmanagement an sich wurde in der Seminarausarbeitung *Agiles Projektmanagement: Konzepte, Werkzeuge und Anwendung* behandelt.

Schon während der Seminararbeiten stellte sich heraus, dass die zugewiesenen Rechte durch die Serveradministration in Magdeburg nicht ausreichend sein würden. Diese Rechteproblematik zog sich durch die gesamte Durchführung der Projektgruppe, was teilweise zu starken Verzögerungen geführt hat. Eine weitere vorbereitende Maßnahme war die Durchführung einer Fallstudie innerhalb des SAP HANA Systems, um auch dort einen generellen Einblick zu bekommen. Während dieser Zeit trat der Projektgruppe kurzfristig ein 10. Mitglied bei, welches die Gruppe allerdings schnell wieder verlassen hat. Weiterhin verließ am Ende der Seminarphase ein weiteres Mitglied die Projektgruppe, sodass lediglich noch 8 Mitglieder verblieben.

Nachdem die Seminarphase abgeschlossen war, die einen groben Einblick in die Thematik verschaffen konnte, wurde anhand von Interviews zwischen den Projektgruppenteilnehmern und den Betreuern versucht, die Aufgabe der Projektgruppe näher zu definieren. Die Mitglieder der Projektgruppe wurden für die Interviews in drei Gruppen aufgeteilt, die zum einen jeweils generelle Fragen, aber auch spezifische Fragen zu den Themen *Use-Cases*, *Plattform* und *Analyse* erarbeiten und durch das Interview beantworten sollten.

Aus den Interviews ergaben sich schließlich einige Anforderungen, die allerdings im Laufe der Projektgruppe immer wieder angepasst werden mussten. Für die endgültigen Anforderungen soll an dieser Stelle auf Abschnitt **9.1** verwiesen werden. Als übergeordnetes Szenario hat sich eine Anwendung für Experten User ergeben, welche in Abschnitt **9.3** näher erläutert wird.

Kurz nach den Interviews verließ dann noch ein weiteres Mitglied die Projektgruppe, sodass, für den folgenden Verlauf und bis zum Ende 7 Mitglieder verblieben.

Neben dieser starken Personalfluktuation war bis zum 02.04.2015 nicht klar, in welchem Umfang und Format die Projektgruppe Verkehrsdaten für die Stadt Oldenburg bekommen würde, welche eine wesentliche Grundvoraussetzung für die Bearbeitung der Aufgabe darstellten.

Um die Wartezeit sinnvoll zu nutzen, wurden andere, möglicherweise relevante Daten beschafft, um diese zum einen nicht später akquirieren zu müssen und zum anderen das Hochladen von Daten in das SAP HANA System zu testen. Wie in Abschnitt **4.2.1** beschrieben, ist es möglich die Daten von einem FTP-Server sehr komfortabel in vorbereitete Tabellen zu laden, solange der FTP-Server mit dem SAP HANA System verbunden ist. Da ein solcher Server von Host-Seite abgelehnt wurde, war die Projektgruppe darauf angewiesen, die Daten auf eine andere Art und Weise in das SAP HANA System zu integrieren. Als passable Lösung hat sich der Transfer über eine ODBC-Schnittstelle erwiesen. Bei dem Import der Daten über die ODBC-Schnittstelle hat sich allerdings herausgestellt, dass es nicht möglich ist SQL Befehle auszuführen, die mehr als 1000 Zeilen beinhalteten. Dadurch hat der Import selber einige Zeit in Anspruch genommen.

Die Daten, die während dieser Zeit erhoben wurden waren

- Wetter Daten (vgl. Abschnitt **6.9**)
- Event Daten (vgl. Abschnitt **6.11**)
- OpenStreetMap Daten (vgl. Abschnitt **6.5**)
- ADAC Daten (vgl. Abschnitt **6.7**)

Als ab Mitte April der Projektgruppe die Daten der Verkehrsleitzentrale zur Verfügung standen, mussten diese zunächst vorbereitet werden (vgl. Abschnitt **7.10**). Parallel zu der Vorbereitung der Verkehrsdaten konnten auch die durch die Abteilung zur Verfügung gestellten ADAC Daten hinsichtlich des CO2 Ausstoß vorbereitet werden (vgl. Abschnitt **7.8**).

Anschließend ging es daran geeignete Prognoseverfahren zu finden um die gestellten Anforderungen an das System zu erfüllen. Bei der Recherche nach geeigneten Verfahren fand die Projektgruppe Bibliotheken, die innerhalb des SAP HANA Systems genutzt werden könnten um beispielsweise Regressionsberechnungen durchzuführen. Auch hier wurde die Anfrage nach einer Freigabe der Bibliotheken von Host-Seite aus abgewiesen. Aus diesem Grund musste sich die Projektgruppe überlegen, wie eine Prognose dennoch realisiert werden könnte.

Die Lösung für das Problem lag darin, die Algorithmen der Prognoseverfahren auf dem SAP HANA System manuell zu implementieren. Als Sprache stand der Projektgruppe dafür SQLScript zur Verfügung, welches direkt auf dem System ausgeführt wird. Da die Prognose für jede Zählschleife separat ausgeführt werden musste, galt es die benötigten SQL Befehle dynamisch für die einzelnen Zählspulen in einer Prozedur zu erstellen. Der Vorteil dieses Vorgehens lag darin, dass diese einfache angepasst werden kann, sollte sich das Prognoseverfahren verändern. Der Nachteil von dynamischen SQL Befehlen liegt allerdings darin, dass sie zur Laufzeit aufgebaut werden müssen, was viel Rechenzeit

beansprucht. Dennoch war es möglich die einzelnen Prognoseverfahren innerhalb eines angemessenen Zeitraums durchzuführen.

Parallel zu der Entwicklung der Prognoseverfahren wurde ein Webportal entwickelt, welches für die Darstellung genutzt wird. Innerhalb des Portals werden die aktuellen Werte der einzelnen Zählschleifen als Heatmap dargestellt und darüber hinaus besteht die Möglichkeit sich die Prognosewerte für einzelne Zählspulen anzeigen zu lassen. Weitere Features des Portals sind die Anzeige der Eisenbahnzüge, die innerhalb der nächsten Stunde zur aktuellen Zeit in Oldenburg ankommen, die Anzeige der Buslinien und der aktuellen Position der Busse, die Darstellung von Wetterdaten sowie von Veranstaltungen, die an einem angegebenen Tag ausgerichtet wurden und die Visualisierung der Ampelanlagen. Diese zusätzlichen Features können von Experten Usern genutzt werden um Zusammenhänge zwischen dem Verkehrsfluss und den entsprechenden Features für eine gegebene Fragestellung zu bestimmen.

11 Ausblick

Im folgenden Abschnitt soll dargestellt werden, welche Möglichkeiten bei der Verbesserung der Algorithmen und der damit verbundenen Darstellung bestehen. Darüber hinaus soll betrachtet werden, wie das System einerseits verbessert und mit neuen Features versehen werden- und andererseits einen höheren Automatisierungsgrad erreichen kann. Zu diesem Zweck sollen wichtige Themen, die ein großes Verbesserungspotential bergen im Folgenden einzeln betrachtet werden.

Algorithmen

Die Implementation der Algorithmen erfolgte auf Basis wissenschaftlicher Veröffentlichungen. Hierzu wurden spezifische Verfahren genutzt um eine möglichst präzise Vorhersage für Zeitpunkte von 15, 30 und 60 Minuten zu berechnen. Die 30- und 60-minütigen Zeitpunkte sind hierbei bereits sehr präzise. Problematisch wurde es bei der Berechnung des 15-minütigen Zeitpunktes, da das genutzte Verfahren wesentlich Rechenaufwändiger ist als die anderen Algorithmen, wodurch eine Zeitüberschreitung des 30-Sekunden Limits festgestellt wurde. Somit kann festgehalten werden, dass das SAP-Hana System nicht performant genug war, um den komplexeren jedoch auch präziseren Algorithmus innerhalb des 30 Sekunden Intervalls durchzuführen. Ergo führt eine Steigerung der Performance des Systems zu einer erhöhte Präzision bei der Berechnung der Algorithmen. Weiterhin, in der langfristigen Prognose, werden die vorhergesagten Zählspuren-Werte bislang lediglich auf Grundlage der Tageszeit getroffen. Durch eine größere Datenbasis wären allerdings auch andere Modelle zur langfristigen Prognose denkbar und das bisherige Regressionsmodell könnte z.B. durch ein neuronales Netz abgelöst werden. Auf diese Weise können dann auch andere Dimensionen wie das Wetter oder auch bestimmte Veranstaltungen in die Analysen mit einfließen und deren Einfluss hinsichtlich des aufkommenden Verkehrs bestimmt werden.

Automatisierung

Die Automatisierung des Gesamtsystems spielte während der gesamten Projektlaufzeit eine übergeordnete Rolle, sodass bei der eventuellen Bereitstellung eines Live-Datenfeeds das System umgehend auf einen automatisierten Betrieb umgestellt werden könnte. Der Import der Daten und somit die externe Verbindung zum Hana-System wurde stets über eine ODBC-Schnittstelle hergestellt. Der Nachteil war, dass der Datenaustausch sehr langsam von statten ging. Hierbei bietet das SAP Hana System eine Lösung mit einer seperaten Cloud, die wesentlich performanter arbeitet und auf der einerseits komplette Datensätze abgelegt werden können, und die andererseits auch als Schnittstelle für Live-Daten fungieren kann. Diese Cloud stand dem Projektteam leider nicht zur Verfügung und so musste auf die ODBC-Lösung ausgewichen werden. Ein weiterer Nachteil der Nutzung der ODBC-Schnittstelle war die externe Steuerung, und somit die Einrichtung eines externen Web-Servers. Einerseits bietet das SAP-Hana System eine interne Cron job Steuerung, die jedoch ebenfalls im System des Projektteams nicht zur Verfügung stand, sodass ein externer Webserver eingerichtet wurde, welcher die Funktion des Imports per Cron Job

komplett übernahm. Ein letztes Problem bei der Automatisierung waren die knappen Zeitressourcen der Kooperationspartner. Neben der Datenbasis vom März 2015 wurden ebenfalls weitere Daten angefragt um präzisere und aktuellere Werte durch die implementierten Algorithmen errechnen zu lassen. Dies konnte leider aufgrund der oben genannten zeitlichen Probleme nicht realisiert werden. Positiv hingegen ist, dass das System des Projektes RAPID in einer Art und Weise konzipiert und umgesetzt wurde, dass bei einer nachträglichen Übermittlung neuer Werte eine Aktualisierung bzw. die Umstellung auf einen Live-Betrieb mit minimalem Aufwand erreichbar ist.

Schadstoffbelastung

Die Schadstoffbelastung, insbesondere in Oldenburg, ist ein sehr brisantes Thema. Hierbei konnte im Rahmen des Projektes lediglich ein Mittelwert auf die Nutzung der Zählspulen hochgerechnet werden. Eine potentielle Erweiterung dieses Systems wäre es zwischen PKW, LKW und Bussen zu differenzieren, sodass individuelle Heatmaps zur visualisierung der Emissionen erstellt werden können. Ebenfalls würde dies einen wesentlich präziseren Rahmen liefern wodurch keine zu starke Verallgemeinerung von nöten wäre. Darüber hinaus sind die Schwellenwerte, bei denen von hoher, mittlerer und niedriger Schadstoffbelastung gesprochen werden kann, äußerst vage und schwer zu eruieren. Eine Orientierung liefern EU Richtlinien, jedoch existieren keine fixen Werte. Solche Werte im Rahmen eines wissenschaftlichen Projektes zu eruieren würde dem erstellten Portal eine wesentlich höhere Genauigkeit der eigentlichen Schadstoffbelastung ermöglichen.

Visuelle Features

Bei den Visuellen Features, die in das Front-End des Portal integriert werden können wäre zu Beginn ein SQL-Abfrageeditor zu nennen. Dies wurde den Vorteil bringen, dass ein Experten-User, der mit der Datenbankstruktur des SAP-Hana Systems vertraut ist problemlos individuelle Reports erstellen kann und sämtliche hinterlegten Daten auslesen kann, ohne einen Zugang zur eigentlichen Datenbank zu haben und somit Gefahr zu laufen, die Datenbasis absichtlich oder unabsichtlich zu manipulieren. Das Portal könnte neben Experten-Usern ebenfalls Privatpersonen zur Verfügung gestellt werden und ein Wegweise-Algorithmus könnte erstellt werden, welcher Privatpersonen per App durch den Oldenburger Stadtverkehr führt und vielbefahrene Straßen umgeht. Hierbei würden Eventdaten und Zugdaten berücksichtigt, sodass geschlossene Bahnschranken und Großveranstaltungen umfahren werden können. Kritisch gesehen müsste überprüft werden, ob es in eine Stadt wie Oldenburg sinnhaft ist ein solches System zu installieren, jedoch könnte diese Variante für Großstädte eine große Entlastung einzelner Straßenabschnitte bedeuten und somit eine Verbesserung des Verkehrsflusses. Darüber hinaus könnten Problemstellen, die durch einen hohen Schadstoffwert auffallen ebenfalls entlastet werden, da Staus vermindert- und alternative Routen durch Autofahrer genutzt werden würden. Eine Differenzierung nach LKW und PKW wäre ebenfalls möglich, sodass der vermehrte Ausstoß von Schadstoffen, insbesondere durch LKWs, auf alternative Routen verteilt wird.

Literatur

[AB09] A. Bauer, H. G. ; GmbH dpunkt.verlag (Hrsg.): *Titel fehlt, Aufl. 3.* 2009

[AK11] A. Kemper, A. E. ; Muenchen, Oldenbourg V. (Hrsg.): *Datenbanksysteme-Eine Einfuehrung, Aufl. 8.* 2011

[Bal08] Balzert, Helmut ; Verlag, Spektrum A. (Hrsg.): *Lehrbuch der Softwaretechnik.* Bd. 2. 2008

[Bel07] Belikan, Oliver: *Anforderungsanalyse, Arbeitspapier zu methodischen Anforderungsanalyse.* Bd. 1. 2007

[Chr05] Chrobok, Roland: *Theory and application of advanced traffic forecast methods.* http://duepublico.uni-duisburg-essen.de/servlets/DerivateServlet/Derivate-5656/Chrobokdiss.pdf. Version: 2005

[Cla15] Claassen, Margret: *Intelligente Datenbanken: ODBC/JDBC.* http://www.iai.uni-bonn.de/III/lehre/AG/IntelligenteDatenbanken/Projektgruppe/SS03/Seminar/MargretClaassen.pdf. Version: 2015

[Cle14] Cleve, Jürgen ; München, Oldenbourg Wissenschaftsverlag G. (Hrsg.): *Data Mining.* Bd. 1. 2014

[DKS01] Dräber, Rolf ; Koschek, Holger ; Sabling, Carsten ; KG, O Reilly Verlag GmbH & C. (Hrsg.): *Scrum kurz & gut.* Bd. 1. 1. Aufl. 201

[Eas00] Easter, Mert ; Berlin, Springer (Hrsg.): *Knowledge Discovery in Databases. Techniken und Anwendung.* Bd. 1. 2000

[Erl] Erlemann, Kai: *Straßenverkehr.* http://www-brs.ub.ruhr-uni-bochum.de/netahtml/HSS/Diss/ErlemannKai/diss.pdf,Zugriffam:03.10.2015

[fie] *fietsberaad OL-Verkehrsmittelumfrage- Oldenburg.* http://www.fietsberaad.nl/library/repository/bestanden/OL-Verkehrsmittelumfrage-Oldenburg.pdfS.5

[Fou15] Foundation, R: *What is R?* https://www.r-project.org/about.html. Version: 2015

[Gab09] Gabriel, Roland ; Witten, W3L G. (Hrsg.): *Data Warehouse und Data Mining.* Bd. 1. 2009

[Gil15] Gilsdorf, Frank: *ODBC/JDBC.* http://www.syssoft.uni-trier.de/systemsoftware/Download/Seminare/Middleware/middleware.9.book.html. Version: 2015

[Gmb15] GmbH, Experian: *Was ist Datenbereinigung?* http://www.experian.de/glossar/datenbereinigung.html. Version: 2015

[Hil15] Hildebrand, Knut ; Verlag, Springer (Hrsg.): *Daten- und Informationsqualität*. Bd. 3. 2015

[IBM15] IBM: *SPSS Modeler CRISP-DM-Handbuch*. ftp://public.dhe.ibm.com/software/analytics/spss/documentation/modeler/15.0/de/CRISP-DM.pdf. Version: 2015

[Ill12] Illapani, Prasad: *Third Party ETL Tool Certification Program for SAP HANA*. https://blogs.saphana.com/2012/12/12/third-party-etl-tool-certification-program-for-sap-hana/. Version: 2012

[int] *Intel.de unstructured-data-analytics*. (http://www.intel.de/content/dam/www/public/emea/de/de/pdf/unstructured-data-analytics-%20paper.pdf,Zugriffam:04.03.2015

[Jun12] Jung, Thomas: *SAP HANA Extended Application Services*. http://scn.sap.com/community/developer-center/hana/blog/2012/11/29/sap-hana-extended-application-services. Version: 2012

[Kei15] Keim, Daniel: *Datenvisualisierung und Data Mining. Grundlagen der praktischen Information und Dokumentation*. http://kops.uni-konstanz.de/handle/123456789/5623. Version: 2015

[Kel13] Kelly, Jeff: *Primer on SAP HANA*. http://wikibon.org/wiki/v/Primer_on_SAP_HANA. Version: 2013

[Kof11] Kofler, Michael: *Ubuntu Server*. https://books.google.de/books?id=dYESFmtciKgC&pg=PA13&lpg=PA13&dq=ubuntu+server&source=bl&ots=K3naed_0Xc&sig=BPP1BdVuaIBAtLOl19c1flzUBqw&hl=de&sa=X&ved=0CEoQ6AEwBjgKahUKEwjG6-Om5KHHAhWKDywKHVeHDLo#v=onepage&q=ubuntu%20server&f=false. Version: 2011

[Lem13] Lemmer, Karsten: *Verkehrsmanagement - DLR*. http://www.dlr.de/fs/Portaldata/16/Resources/verkehrsmanagement/DLR-TS_Verkehrsmanagement_2014_dt_online.pdf. Version: 2013

[reg] *oth regensburg SAP HANA AKWI*. https://www.oth-regensburg.de/fileadmin/media/fakultaeten/im/forschung-projekte/ccse/pdf/SAP_HANA_AKWI_2014_v6.pdf

[SAP14] SAP: *SAP HANA Developer GUIDE*. http://help.sap.com/hana/SAP_HANA_Developer_Guide_en.pdf. Version: 2014

[SAP15] SAP: *SAP HANA R Integration Guide*. http://help.sap.com/hana/sap_hana_r_integration_guide_en.pdf. Version: 2015

[sta09] *Umweltbundesamt.* https://www.umweltbundesamt.de/sites/default/files/medien/publikation/long/3565.pdf. Version: 2009

[Sta15] STATISTA: *Definition Regressionsanalyse.* https://www.fbi.h-da.de/fileadmin/personal/j.wietzke/mein_ordner/Programmierwettbewerb_V/OpenStreetMap_Auszug.pdf. Version: 2015

[WFH11] WITTEN, Ian H. ; FRANK, Eibe ; HALL, Mark A.: *Data mining: Practical machine learning tools and techniques.* 3. ed. Amsterdam : Elsevier/Morgan Kaufmann, 2011 (Morgan Kaufmann series in data management systems). http://reference-tree.com/book/data-mining-practical-machine-learning-tools-and-techniques?utm_source=gbv&utm_medium=referral&utm_campaign=collaboration. – ISBN 9780123748560

[Wie15] WIETZKE, Joachim: *OpenStreetMap- Auszug.* https://www.fbi.h-da.de/fileadmin/personal/j.wietzke/mein_ordner/Programmierwettbewerb_V/OpenStreetMap_Auszug.pdf. Version: 2015

Zusammenfassung der Ergebnisse aus den Interviews

Allgemeines:

Es soll eine Plattform zur Analyse von Mobilität erstellt werden, Mobilität ist hier als generischer Begriff zu sehen. Die Plattform soll Kommunikation mit der SAP HANA ermöglichen, konfigurierbar und flexibel sein. Es soll ein Mehrwert im Zusammenhang mit Verkehrsdaten entstehen: Sie sollte mobilitätsspezifische Werkzeuge zur Verfügung haben und geospezifische Aktionen (Karte berechnen, Straßen entlangfahren, Visualisierung) drauf haben. Wünschenswert ist wenn dieses nicht „READONLY" sondern Interaktiv funktioniert. Ein spezieller Use Case könnte dabei beispielsweise ein Städteplaner sein, der „Worst-Case Szenarien" durchspielt und dabei neue Bushaltestellen sowie Straßen plant. Ein weiterer Use Case ist die Errichtung einer Car Sharing Station an strategisch günstigen Stellen.

Dabei sollten verschiedenste Quellen in der Plattform zusammengeführt werden aus dem verschiedene Wissensaspekte gezogen werden können. Beispielsweise könnten die Prognosen der VWG im verbessert werden. Hier können aktuelle Verkehrsinfos einfließen und die Ankunftszeit der Busse genauer „in Echtzeit" dargestellt werden (siehe Beispiel Busoptimierung).

Das Wichtigste ist der Mittelbau der Plattform. Dabei können die bereits genannten Use Cases umgesetzt werden.

Beispiel Busoptimierung:

- Wann kommen die Busse im Durchschnitt an?
- Halten die Busse die vorgegebenen Zeiten ein?
- Benötigen die Busse mehr Zeit? Wenn ja, wie viel?
- Müssen die Ankunftsangaben auf den Anzeige-Tafeln angepasst werden?

Lasten-/Pflichtenheft:

- Soll von der PG selbst erstellt und bereitgestellt werden

Grundlagensysteme, auf denen diese PG aufbaut:

- Cuberunner (Smart Windfall Control):
    - SAP HANA wurde verwendet, dazu R-Funktionen
    - Anhand der Daten sollten Prognosen für die Wartungszyklen der Windkraftanlagen abgeleitet werden können.
- Eventuell ALISE (nur Simulation wurde durchgeführt)
- Masterarbeiten und Dokumentationen werden zur Verfügung gestellt

Normen:

- Das Thema kann vernachlässigt werden, zu beachten sind
    - Datenschutzgesetze
    - Telekommunikationsgesetze

Mögliche Ergebnis-Objekte (Kennzahlen/Return-Werte, Operationen, Dokumente):

- Kennzahlen
  (je nachdem, was die Analysten, Entscheider, etc. benötigen. Die Analyse-Gruppe übernimmt im Zuge des Projektes die Rolle des Endanwenders und soll selbstständig ermitteln, welche Kennzahlen für die Entscheider relevant sind)
- Funktionen (siehe oben)
- Konfigurationen für den nötigen ETL-Prozess in Bezug auf die Mobilitätsdaten

<u>Stakeholder:</u>

Allgemeine Stakeholder:

- Mobilitätsdienstleister
  - Für neue Märkte – Car Sharing (vergangenheitsbezogen)
  - Für Prognosen – VWG (Echtzeitrelevant)
  - Stadtplanung (Bau neuer Bushaltestellen)

Nutzer:

- Hauptsächlich Analysten
- Es soll kein Webfrontend entwickelt werden in dem ein Analyst arbeitet. Es sollen Techniker die Plattform später verwenden.

Systemzugriff des Nutzers:

- Entscheidung der PG:
  - Umfang beachten
  - Ggf. generische Schnittstellen
  - Reporting für Endkunden nicht im Vordergrund
  - Datenaspekt hat Priorität!

Anforderungen des Nutzers an das System:

- Jähnig
  - Datenlieferant – noch in Verhandlung
- VWG
  - Keine Anforderungen
- Betreuer
  - PG als Experten für Mobilität (Potentiale erkennen etc.)
  - Haben klare Idee an PG, bei Unklarheit nachfragen

Informationen für Akteur:

- Verkehrsflüsse (Ankunftszeiten, Durchkommen)
- Standortbestimmung
- Bottlenecks erkennen
- Umwege vorschlagen
- Gründe für Verkehrsaufkommen erkennen und ausgeben

Typischer Flow eines Nutzers, Fragestellung der Akteure:

- Daten sollen in das System eingespielt und verarbeitet werden
- Regenvorhersage etc. mit einspielen

Daten:

Daten Allgemein:

- Datenverarbeitung ist zentraler Punkt des Projekts:
    - Aufbereitung und Analyse
    - Erkennen bestehender Defizite
- Datensicht am wichtigsten
- Form der Daten ist noch nicht klar, nach Möglichkeit Echtzeitdaten
- Daten aus der Region Oldenburg
- Eingehende Daten sollen über SAP HANA verarbeitet werden
- Sicht auf die Daten nicht nur mit BI-Self-Services
    - Data Mining
    - Wissen aus Daten generieren
    - Neue Erkenntnisse
- Periodischer Report

Art der Daten:

- Verkehrsdaten
- Daten sind abhängig von der Quelle, Beispiele:
    - Jähnig (gewünschtes Unternehmen, Verhandlungen laufen noch)
        - Daten stammen aus Zählanlagen und Ampelsensoren, die den Verkehrsfluss innerhalb der Region Oldenburg überwachen
        - Braunschweig: AIM-Projekt (ähnliche Zielsetzung mit hohem Budget, Daten evt. ebenfalls von Jähnig bereitgestellt)
    - Verkehrsleitzentrale
    - Schrankendaten
    - Ampelsensoren (Fahrzeug-Typ etc.)
    - Datenanreicherung über externe Quellen
- Beispiele: Straßen, Karten, Planungssysteme, Wetter, Beschwerden bei Bus-Verspätungen
- Die Gruppe kann sich zusätzlich Gedanken über den Einsatz von weiteren Datenpaketen machen.

Format und Größe der Daten:

- Eine genaue Größe der Daten kann zum jetzigen Zeitpunkt nicht bestimmt werden
- Generell wird mit einer großen Menge an Daten gerechnet
- Der Speicherplatz der SAP HANA „In Memory“ wird mit 256 GB angegeben. Der Storage im Hintergrund ist um ein 4faches höher.
- Das Größenverhältnis der Daten hängt auch von der Anzahl der Daten ab, die wir analysieren wollen. Hier ist die Arbeit in der Gruppe gefragt.

- Generell gesagt sollen wir uns nicht auf Größen/Formate etc. festlegen, sondern ein Modell entwickeln, was auch genutzt werden kann
- Es gibt nicht „das“ Format mit dem wir später arbeiten. Wir müssen die Plattform universell gestalten, sodass eine Vielzahl von Formaten unterstützt wird. Hier ist auch wichtig, dass wir die Plattform nicht in Abhängigkeit eines Formates erstellen sondern uns auf Use Cases beziehen. Die Datenformate müssen somit von der Gruppe „gematched“ werden.
- Hierbei ist jedoch wieder abzuwarten, bis die Daten der Jähnig Gruppe vorliegen. Die Idee einer generischen Plattform wurde vorgebracht und diese können sowohl für Busse, Autos, LKWs und Fahrräder gelten. Somit sollte die Plattform nicht abhängig von den Daten sein. Das zu entwickelnde Modell sollte viele Facetten abdecken. Das Modell „in der Mitte“ ist entscheidend.
- Die Verwendung eines Bottom up Verfahrens wurde hier vorgebracht, da der Erhalt eines Ergebnisses das zentrale Ergebnis dieser Plattform darstellt. Falls die Daten der Jähnig Gruppe noch nicht vorliegen, können für den Anfang auch Daten der Regierung „Regierungsoem“ aber auch andere Grunddaten verwendet werden.
- Werden nur sensorische Daten oder auch andere Formate verwendet? [2]
    - Wie bereits angemerkt werden Daten aus Zählschleifen oder aus ampelgesteuerten Vorgängen verwendet. Dabei ist jedoch zu beachten, das vorhandene BI Lösungen nie eine integrierte Lösung darstellen sondern aus einer Vielzahl von Tools bestehen. Diese Stellen nur ein Frontend dar.

Datenübertragung:

- Keine Vorgabe, möglichst flexibel/generisch

Datenverarbeitung:

- Hier ist eine Unterscheidung zwischen der Speicherung von Daten „In Memory“ und einer Speicherung im Storage vorzunehmen. Wie genau die Einpflege von Daten aussieht ist zum jetzigen Zeitpunkt unbekannt und obliegt später uns.
- Segmentierung (Auto, Bus, Bahn oder nur „viel Verkehr“ und „wenig Verkehr“) ist abhängig davon, welche Daten wir erhalten: Wird erst in der Projektphase ersichtlich
- Daten sollen auf Karten abgebildet werden können
- Mit welchen Daten wir arbeiten und wie wir diese in welchen Formaten verarbeiten, bleibt der PG überlassen.

Echtzeit:

- Hierbei geht es um den „Datenload“ in die Datenbank. Wichtig ist es, so viele Aufgaben wie möglich an die HANA auslagern und spezifische Analysen ergänzen. Die Umgehung des Flaschenhalses ist dabei der wichtigste Aspekt der von der Gruppe beachtet werden muss.
- Als Beispiel hierfür könnten folgende Komponenten entwickelt bzw. eingebunden werden:
    - Eine Komponente für den Datenimport.
    - Eine Komponente für die Analyse
    - Eine Komponente für die Informationsbereitstellung
- Signifikanz des Echtzeitaspekts ist vom jeweiligen Use Case abhängig. Für Busse wäre der Echtzeitaspekt in Minuten interessant. Wenn die Stadt Reports braucht, dann könnte der

Echtzeitaspekt in Monaten interessant sein. Zudem ist die Abhängigkeit der Daten und der jeweiligen Fragestellung wichtig. Echtzeitdarstellung wäre gewünscht aber nur wenn mit Daten möglich.
- Ein Datenimport ist für die Echtzeitanforderung nicht wichtig.
- Echtzeit ist wenn möglich wünschenswert, Prognosen sollen erstellt werden können (Straßenausbau, Verspätungen, Stau, etc.)

Systemanforderungen:

- Schnittstellen möglichst offen halten
- Plattform generisch/flexibel halten

Schnittstellen:

Innerhalb des Datenmanagements müssen wir uns alle Schnittstellen überlegen. Hierbei ist die Kommunikation mit HANA eine der wichtigsten Schnittstellen. Diese stellt eine Primäraufgabe für die zu entwickelnde Plattform dar. Zudem müssen wir uns in Abhängigkeit unserer Use Cases mit weiteren Adaptern beschäftigen. Hierbei sind Schnittstellen zur Wetterdatenbank des Deutschen Wetterdienstes mit anzubinden.

Technologien der vorigen Gruppen/ besonders empfehlenswerte Software:

Die Alise Gruppe hat eine Architektur im Webfrontend Format mit einer Kartenstellung entwickelt. Hier wurden Mobilitätsflüsse nachempfunden. Beispielsweise konnte zu Stoßzeiten die Auslastung einer Straße auf Simulationsbasis modelliert werden. Die Gruppe hat hauptsächlich Java Technologien sowie Apache aus dem Bereich Open Source verwendet.

Unsere Gruppe soll dabei noch einen Schritt weiter gehen und die Use Cases auf Echtzeitbasis nachempfinden. Beispielsweise können wir auf die Verkehrssituation von vor einem Jahr zurückgreifen und für den heutigen Verkehr Prognosen (Was passiert in einer Stunde bei Starkregen) erstellen.

Die Gruppe Cuberunner ist für unser Projekt schon eher interessant, da hier mit SAP Hana gearbeitet wurde und R Routinen mit eingebunden wurden. Ebenfalls wurde deren Projekt mit der SAP Hana Datenbank verbunden.

Aktuelle PG Dokumentationen können im StudIP geladen werden, es finden sich weitere Informationen in den Seminarthemen und den PG Dokus.

ABAP/J2EE-Schnittstelle:

Die Verwendung von ABAP- bzw. J2EE Schnittstellen ist der Gruppe Überlassen. Hier herrscht freie Entscheidungsbasis die nur von der Lizenzfrage gestoppt werden kann. Also beispielsweise steht eine Netviewer Lizenz als Plattform in der Abteilung momentan nicht zur Verfügung. Use Case abhängig.

SAP-Komponenten:

Es sollten so viele Komponenten wie möglich aus der SAP HANA und deren korrespondierenden Anwendungen verwendet werden. Welche bleibt uns überlassen. Seminarthemen interessant.

Die Einbindung von SAP HANA ist von uns zu entwickeln. PG Cuberunner als Vorlage/Anregung

Analyse

Datenanalyse:

- Auswertungsmöglichkeiten:
    - Heat-Maps
    - Auslastungen
    - Busverspätungen
    - Stau
    - Car-Sharing
    - Stadtplanung
    - Infrastrukturplanung

Gewünschte Analyseverfahren:

- SAP Hana bietet einfache/allgemeine Analyseverfahren die natürlich keinen speziellen Bezug zu dem Kontext Verkehr haben. Aufgabe der Gruppe ist es dann, selbst herauszufinden, welche Analyseverfahren und Software genutzt werden kann (R-Funktionen). Diese soll dann verwendet werden.
- Die PG ALISE hat Software entwickelt mit der der Verkehrsfluss für Städte (mit fiktiven Daten) simuliert werden kann. Ein wesentlicher Unterschied soll sein, dass echte Daten verwendet werden sollen.

Werkzeuge zur Datenanalyse

- SAP Hana bringt auch Analysefunktionen, aber ohne konkreten Bezug auf Mobilitätsdaten. Deshalb sollen spezielle Funktionen eigenständig erstellt werden (z. B. mit R-Language).
- Tools:
    - Import: BO Data Service
    - Datenbank: SAP HANA
    - Browser-gestützte Anwendungssoftware: Business Objects Explorer

Ergebnisaufbereitung:

- Wie müssen die Ergebnisse der Funktionen aufbereitet werden, um sinnvoll anderen Programmen (Excel, SPSS,…) bereitgestellt werden zu können?
    - Offen! Muss selbst von der Analysegruppe der PG „nachempfunden“ werden.

Darstellung:

- Darstellungen sind am Ende abhängig vom Kunden.
- Wünschenswert wäre es, wenn man interaktiv mit Daten arbeiten könnte (Operationen auf den Darstellungen möglich? [Drill Down, Roll Up, …])

Use-Cases:

- Connected Car
- Car Sharing Stationen an strategisch günstigen Stellen errichten
- Studien von Google und Nokia
- Regionale Ebene steht im Vordergrund
- Planung neuer Bushaltestellen und Straßen auf Basis von Worst-Case-Szenarien

Programmierrichtlinien RAPID

Inhalt

1. Einleitung

Programmierrichtlinien sind wichtig, um Vorgaben bezüglich einer Struktur und den Aufbau eines Quellcodes einheitlich zu behandeln. Die Verwendung in Projekten ist daher nicht unüblich um einen einheitlichen und leicht verwertbaren Programmcode zu erzeugen. Die Vermeidung von Spagetti-Code-Tendenzen wird angestrebt und stellt sicher, dass der erzeugte Code später für andere Programmierer leichter lesbar ist um eine schnellere Einarbeitung zu garantieren. Zudem reduziert sich die Wartungszeit während einer Wartungsphase.

Im Folgenden werden Programmierrichtlinien für JAVA, Javascript, HTML und PHP festgelegt die im Wesentlichen mit dem von Oracle Sun herausgegebenen Java Code Conventions[1] übereinstimmen. Zudem werden unterstützend, die Java-Programmierrichtlinien von Prof. Dr. Pape [2] von der HS Karlsruhe hinzugezogen. Im Bereich Javascript dient eine verkürzte Form der JQuery Guidelines von Rüdiger Marwein [3] als Referenz. Die Programmierrichtlinien für HTML Seiten entstammen einer Sammlung von Regelungen von Rüdiger Borrmann[4] und die PHP Richtlinien basieren auf dem von uns verwendeten Framework CodeIgniter Version 2.2.0[5].

Für das Projektteam RAPID wird festgelegt, die hier erläuterten Standards für die Erzeugung von JAVA-, Javascript- oder HTML- Code einzuhalten.

[1] http://www.oracle.com/technetwork/java/codeconvtoc-136057.html
[2] http://www.home.hs-karlsruhe.de/~pach0003/informatik_1/java_richtlinien/einleitung.html#uebersicht
[3] http://keinerweiss.de/wp-content/JavaScript-Coding-Guidelines.pdf
[4] Htmlbasic.wikispaces.com/InfoProgrammierRichtlinien
[5] https://ellislab.com/codeigniter/user-guide/general/styleguide.html#file_format

2. JAVA

2.1.Java Namenskonventionen

2.1.1. Bezeichner

2.1.1.1. Aussagekräftiger Name

§ Für Bezeichner müssen immer aussagekräftige und selbsterklärende Namen gewählt werden.
§ Es dürfen keine Abkürzungen für Bezeichner verwendet werden

Durch die Verwendung eines aussagekräftigen Namens wird der Quellcodeund die Anforderung an die Lesbarkeit erfüllt. Zudem wird der Quellcodeverständlicher.

Beispiel:

Statt: *KFZ, DB*
Besser: *Kraftfahrzeug, DeutscheBahn*

2.1.1.2. Keine verschiedenen Sprachen verwenden

§ Es dürfen keine verschiedenen Sprachen (z.B. Englisch und Deutsch) miteinander in einer Klasse vermischt werden

Zum einen wird der Quellcode dadurch verständlicher und lesbarer und zum anderen muss die Sprache im Vorfeld festgelegt werden. Im Rahmen der Projektgruppe RAPID wird der Quellcode auf **Englisch** programmiert.

2.1.1.3. Alphanumerische Zeichen verwenden

§ Es sollten nur die alphanumerischen Zeichen A-Z, a-z, 0-9 und den Unterstrich _für Bezeichner verwendet werden.

Durch die Verwendung von alphanumerischen Zeichen wird der Quellcode international portabel. Gerade in Verbindung mit verschiedenen Texteditoren und unterschiedlichen Sprachen ist dies unerlässlich.

Beispiel:

Person, zugFahren, MAXIMALE_ANZAHL, ueberpruefen

2.1.1.4. Verwendung des Upper Case Camel Styles

§ Bei Bezeichnern, die aus mehreren Teilwörtern bestehen, muss der erste Buchstabe jedes Teilworts großgeschrieben werden.

Dies wird verwendet, um längere Bezeichner übersichtlicher und lesbarer zu gestalten.

Beispiel:

zugFahren, UniversitätOldenburg

2.1.2. Variablen

2.1.2.1. Anfangsbuchstaben kleinschreiben

§ Variablen-Namen werden klein geschrieben. Bei mehreren Teilwörtern wird das folgende Teilwort groß geschrieben.

Bezeichner werden dadurch im Quellcodeals Variable identifizierbar.

Beispiel:

anzahlPersonen, quersumme

2.1.2.2. Konstanten mit Großbuchstaben

§ Konstanten werden mit Großbuchstaben geschrieben und Teilwörter werden mit Unterstrichen_ getrennt.

Bezeichner werden dadurch im Quellcodeals Konstante identifizierbar.

Beispiel:

LICHTGESCHWINDIGKEIT

2.1.2.3. Substantive verwenden

§ Mindestens ein Substantiv als Variable verwenden.

Dadurch wird die Bedeutung der Daten ersichtlich

Beispiel:

Person student; Person[] studenten;

2.1.2.4. Zählvariablen

§ Als Zählvariablen in Schleifen sollten Buchstaben wie i,j,k,l verwendet werden.

Die Verwendung von Buchstaben wie i und j hat historische Gründe.

Beispiel:

For(int i = 0; i< studenten.length(); i++){}

2.1.2.5. Verwendung von this vor Objektattributen

§ this referenziert direkt das Objektattribut und dient daher der Unterscheidung zwischen lokalen Attributen und Objektattributen.

Durch die Verwendung von this werden Fehler im Programmcode vermieden und die Lesbarkeit erhöht.

Beispiel:

this.name = name;

2.1.3. Methoden

2.1.3.1. Verben verwenden

§ Es sollte mindestens ein Verb in Präsensform verwendet werden.
§ Das Verb sollte eine möglichst genaue Beschreibung der Handlung einer Methode geben.

Durch die Verwendung eines genau beschreibenden Verbes wird der Quellcodelesbarer.

Beispiel:

personSuchen();, loeschen();

2.1.3.2. Getter/Setter Methoden

- § Möglichst getter oder setter Methoden verwenden wenn Werte mit dem nachfolgendem Substantiv als Ergebnis zurückgegeben werden oder Werte neu gesetzt werden.
- § Bei der Verwendung von booleschen Operatoren wird is verwendet.

Wird verwendet um das Geheimnisprinzip der Programmierung zu gewährleisten.

Beispiel:

String getVorname(); void setVorname(String vorname),boolean isSchaltjahr();

2.1.4. Static

2.1.4.1. Referenzieren mit Klassennamen

- § Statische Variablen oder -Methoden innerhalb der erstellten Klasse sollten immer mit vorangestelltem Klassennamen referenziert werden.

Dadurch können statische Variablen oder –Methoden klarer von lokalen Variablen, Objekten oder Methoden unterschieden werden.

Beispiel:

Math.abs(-16); Color.red;

2.1.5. Packages

2.1.5.1. Kleinbuchstaben verwenden

- § Ausschließlich Kleinbuchstaben und Zahlen sowie Substantive für Paketnamen verwenden.

Wird verwendet um Portabilität von Javaprogrammen zu verbessern.

2.1.5.2. java/javax als Paketnamen verboten

- § Paketnamen java/javax sind für Java- Erweiterungen vorgesehen.

Damit wird sichergestellt, dass die zukünftige Erweiterbarkeit von Java konform ist und nicht mit anderen internen Paketnamen in Konflikt gerät.

2.2.Java Quelltextformatierungen

2.2.1. Allgemein

2.2.1.1. Programmierstile

§ Das Mischen von verschiedenen Programmierstilen sollte vermieden werden

Der Quellcodewird dadurch lesbarer

Beispiel:

```
If( a > 0 )
    {
     a = a + 1; // Einrückungen von Klammer zu Klammer
    }
```

2.2.1.2. Zeilenlänge

§ Zeilen sollten nie mehr als 80 Zeichen enthalten.

Der Quellcodewird dadurch lesbarer.

2.2.2. Anweisungen

2.2.2.1. Einrücken

§ Bei Kontrollstrukturen wie if, else, while sollte immer eine geschweifte Klammer verwendet werden
§ Anweisungen sollten innerhalb des Klammerausdrucks um 2-4 Zeichen nach rechts eingerückt werden. Wichtig Konstanz!

Dadurch ist eine bessere Lesbarkeit gegeben und Programmierfehler werden vermieden.

Beispiel:

```
If  (    ) {
      If (    ) {
         ....
      }
} else {
      ....
}
```

2.2.2.2. Klammerungs-Stil

§ Entweder sollte die geschweifte öffnende Klammer in der nächsten Zeile geschrieben werden oder hinter der Kontrollanweisung ohne weitere Ausführung hinter der schließenden Klammer.

Der Quellcodewird dadurch kürzer und lesbarer.

Beispiel

```
If ( a > b )
{
     ....
}
    Oder:
If ( a > b ) {
     ....
}
```

2.2.2.3. Anweisungen pro Zeile

§ Jede einzelne Anweisung muss in eine separate Zeile geschrieben werden.

Der Quellcodewird dadurch lesbarer.

2.2.2.4. Continue und Break vermeiden

§ Nach Möglichkeit das Unterbrechen und Fortführen von Schleifen vermeiden. Ausnahme ist break bei case Anweisungen.

Der Quellcodewird dadurch lesbarer.

2.2.2.5. Verschachtelungstiefen von Kontrollstrukturen

§ Zu tiefe Verschachtelungen sollten vermieden werden. Maximale Vertiefung ca. 3 Kontrollanweisungen.

Der Quellcodewird dadurch lesbarer.

2.2.3. Ausdrücke

2.2.3.1. Leerzeichen bei Operatoren.

§ Vor und nach jedem binären Operator sollte ein Leerzeichen gesetzt werden.

Die Ausdrücke bleiben bei längeren Bezeichnern lesbarer.

Beispiel:

flaecheninhalt = kreisradius * kreisradius * 3.14159265

2.2.3.2. Umbrechen von überlangen Ausdrücken

§ Bei arithmetischen Operatoren sollte eine Brechung der Zeile vor einem Operator mit der schwächsten Bindung erfolgen.
§ Bei Methodenaufrufen nach einem Komma.
§ Der umgebrochene Teile sollte um 2-4 Zeichen eingerückt werden und unter dem zugehörigen linken Teilausdrucks des Operators stehen.

Der Quellcode wird dadurch lesbarer.

Beispiel:

```
a * a * a + 3 * a * a * b
          + 3 * a * b * b
```

2.2.3.3. Vergleich mit Booleschen Werten

§ Der Vergleich mit == auf die Boolschen Werte ist nicht zulässig. Besser bei true einen boolschen Ausdruck selbst und bei false mit desse Negation (!).
§ Vermischung mit Kurzschlussoperatoren wie (&&,||) und Booleschen Operatoren wie (&,^,|) vermeiden.

Der Quelltest wird lesbarer.

Beispiel:

Falsch: schaltjahr == true && volljaehrig == false

Richtig: schaltjahr && ! volljaehrig

2.2.3.4. Operatorenreihenfolge

§ Reihenfolge muss eingehalten werden. Besonders bei mathematischen Vergleichen (<,>)

Der Quellcode wird dadurch verständlicher.

Beispiel:

Für 0<i<j<n:

```
If ( 0 < i < j < n )  {

}
```

2.2.3.5. Vermeiden von Seiteneffekten

§ Ausdrücke oder Funktionen sollten keine Seiteneffekte hervorrufen und keine Zustandsänderung herbeiführen.

Programmierfehler werden dadurch vermieden und der Quellcode wird verständlicher.

Beispiel:

Welchen Wert hat a, nachdem die letzte Anweisung ausgeführt wurde.

```
a = ( a = 2 ) + ( a += a) * (a = 1 + a);
```

2.2.4. Methoden

2.2.4.1. Methodenrumpf

§ Methode sollte komplett auf einem Bildschirm passen (ca. 20-30 Zeilen)
§ Überlange Methoden sollten in weitere Methoden ausgelagert werden.

Der Quellcode wird dadurch verständlicher.

2.2.4.2. Deklaration lokaler Variablen

§ Lokale Variablen müssen zu Anfang einer Methode deklariert werden. Im Anschluss folgt eine Leerzeile.

Die Methodenimplementierung wird dadurch verständlicher.

2.3.Javadoc

2.3.1. Allgemeines

2.3.1.1. Beschreibung

§ Beschreiben was eine Methode macht, nicht wie sie es macht.
§ Beschreiben was die Werte in einer Variable bedeuten, nicht wie die Variablen verwendet werden.

2.3.1.2. Kommentarform

§ Kommentar muss so kurz wie möglich und so spezifisch wie nötig sein.

2.3.2. Klassen

2.3.2.1. Entwurfsentscheidung dokumentieren

§ Im Klassenkommentar sollte in einem Satz beginnend mit dem Klassennamen alle wesentlichen Entwurfsentscheidungen ausgedrückt werden.
§ Vermeide die konkrete Nennung von Datentypen.

Das Verständnis einer Klasse wird verbessert.

Beispiel:

```
/**
   Eine Hochschule mit Studenten, Dozenten und Studiengängen.
*/
public class Hochschule {
}
```

2.3.2.2. Klassenbeschreibung

§ Eine Beschreibung wo und von welchen anderen Klassen die Klasse verwendet wird sollte vermieden werden.

Überflüssige Zusatzinformation die das Verständnis einer Klasse nicht verbessert.

2.3.3. Attribute

2.3.3.1. Bedeutung von Werten

§ Es sollte eine Beschreibung des Attributwertes und seiner Bedeutung vorgenommen werden.

Der Quellcode kann dadurch besser verstanden werden.

Beispiel:

```
/**
   Der Verkaufspreis in Euro. Er darf nicht negativ sein.
*/
private double verkaufspreis;
```

2.3.4. Methoden

2.3.4.1. Methodenbeschreibung

§ Beschreibe was eine Methode macht und nicht wie sie implementiert ist.
§ Kommentar sollte mit dem Verb des Methodennamens in Präsensform begonnen werden.
§ Alle Parameter müssen mit dem @param-Tag aufgeführt werden
§ Rückgabewerte analog zu Parameter mit @return-Tag

Beispiel:

```
\**
    Immatrikuliert den Student an der Hochschule

    @param student der zu immatrikulierende Student: darf nicht null sein
*\
public immatrikulieren (Student student);
```

2.4.Sonstiges

2.4.1. Primitive Datentypen

§ Möglichst int statt byte, short oder long verwenden
§ Möglichst double statt float verwenden
§ Keine Vermischung von unterschiedlichen Zahlentypen in einem Ausdruck
§ Wissenschaftliche Notationen nur bei sehr kleinen oder sehr großen Ausdrücken verwenden.

Dadurch werden Fehler im Programmcode vermieden.

2.4.2. Lokale Variablen

§ Lokale Variablen vermeiden, deren Wert nur einmal verwendet wird.

Java-Programme werden dadurch kürzer und lesbarer.

Beispiel:

```
public void getFahrenheit(){
    Return 1.8 * celsius + 32,0;
}
```

2.4.3. Felder

§ Java Deklarationsstil für die Deklaration von Feldern verwenden.

Beispiel:

```
Int [][] matrix;
```

3. Javascript

3.1.Abstände

- Zum Einrücken der Abstände immer Tabs verwenden.
- Leerzeichen zur besseren Lesbarkeit zwischen Ausdrücken.
- Auf einer öffnenden Klammer muss ein Leerzeichen folgen.
- Leere Zeilen dürfen keine Leerzeichen enthalten.
- Es dürfen keine Leerzeichen am Ende einer Zeile stehen.

Beispiel:

```
If ( blah === “foo“ ) {
   Foo ( “bar“,“baz“, { zoo: 1} ) ;
}
```

3.2.Kommentare

- Bei langen Kommentaren Javadoc ähnliche Form (Beispiel a).
- Einzeilige Kommentare enthalten je eine eigene Zeile und stehen über der Zeile, die sie referenzieren. Über der Kommentarzeile muss sich eine Leerzeile befinden (Beispiel b).

Beispiel a:

```
/*
…
*/
```

Beispiel b:

```
Var some = “stuff“;
// We’re going to loop here
For ( var i = 0; i < 10; i++){}
```

3.3.Vergleiche

§ Strikte Prüfung (===) sollte gegenüber (==) bevorzugt eingesetzt werden.

3.4.Blöcke

§ Blöcke müssen immer geschweifte Klammern haben.
§ Ausdrücke dürfen nicht in der selben Zeile wie Bedingungen stehen.
§ Else/else if/catch sitzen mit der öffnenden geschweiften Klammer in einer Zeile.
§ Else if ist zuslässig, ein else{} ist nicht notwendig.
§ Ternäre Operatoren sind nicht anstelle von if/else zu verwenden.

Beispiel:

```
Var a = “nicht null“;

If ( b == 0 )  {

   a = “null“;

}
```

3.5.Funktionsaufrufe

§ Leerzeichen rund um Funktionsparameter verwenden. (Beispiel a)
§ Ausnahme foo (true). Hier sind keine Leerzeichen erlaubt wenn es sich innerhalb eines anderen Funktionsaufrufes befindet. (Beispiel b)
§ Ausnahme Funktions- Objekt- und Array-Literale werden direkt an die Klammern angelegt sofern sie das einzige Argument sind. (Beispiel c)
§ Ausnahme mehrzeilige Funktions- Objekt- und Array Literale werden direkt an die schließende Klammer gelegt. (Beispiel d)
§ Hinter Kommas und Semikolons muss immer ein Leerzeichen stehen, außer am Ende einer Zeile.

```
Beispiel a:
Foo ( true ) ;
Foo ( “blah” ) ;
Beispiel b:
Foo ( bar(true) ) ;
Beispiel c:
Foo (function() { }) ;
Foo ([  ]) ;
Beispiel d:
Foo ( true, { blah:  “baz“ }) ;
```

3.6. Arrays und Objekte

§ Leere Objekte und Arrays benötigen keine extra Leerzeichen.

Beispiel:

[], {}

3.7. Deklaration und Wertzuweisung

§ Wertzuweisungen müssen immer mit einem Semikolon abgeschlossen werden.
§ Nach Semikolons muss immer ein Zeilenumbruch erfolgen.
§ Wertezuweisungen in einer Deklaration müssen immer in einer eigenen Zeile stehen
§ Deklarationen ohne Wertzuweisungen müssen am Anfang der Deklaration stehen.

Beispiel:

```
Var a, b, c,
   test = true,
   test = false;
```

3.8. Zeichenketten

§ Zeichenketten müssen immer mit Gänsefüßchen angegeben werden, nicht mit Hochkommata.

4. HTML

4.1.Aussehen einer HTML Seite

- § Nach Möglichkeit die wichtigsten Infos im sichtbaren Bereich des Bildschirms ohne Scrollen darstellen.
- § Horizontales Scrollen ist zu vermeiden.
- § Zusammenhängende Seiten haben ein einheitliches Design.
- § Angenehme Farben und Kontraste verwenden, um die Lesbarkeit der HTML Seite zu gewährleisten. Farbe und Schrift dürfen nicht identisch sein.
- § Vergrößerung des Darstellungsinhaltes darf nicht zu einer Verunstaltung der Seite führen.
- § Bilder dürfen nur in benötigter Auflösung verwendet werden. Zu große Bilder verzögern den Ladevorgang.
- § Bilder dürfen nicht skaliert werden.

4.2. Quellcode einer HTML Seite

- § Öffnende und schließende Tags bei langen Zeilen und komplexen Elementen stets untereinander schreiben.
- § Einrückungen für eine bessere Übersichtlichkeit des Quellcodes verwenden.
- § Kommentare zum besseren Verständnis verwenden.
- § Leerzeilen für eine bessere Struktur einfügen.
- § Frames nur noch als Gestaltungsmittel verwenden.
- § Layoutformatierungen finden ausschließlich mittels CSS statt. Diese werden im head Bereich der HTML Seite eingebunden.
- § IMG Tag mit korrekten Werten für width und height verwenden um Skalierungsfehler zu vermeiden.

4.3. Konventionen für Dateinamen einer HTML Seite

- § Verständliche Namen wählen. Z.b. impressum.html, hauptseite.html
- § Nicht zu lange Dateinamen (maximal 32 Zeichen verwenden)
- § Keine Sonderzeichen verwenden z.B. ä,ü,ß etc.
- § Der Punkt sollte nur genau einmal vor der Dateinamenserweiterung stehen z.B. dateiname.html
- § Grundsätzlich ist die Endung .html zu bevorzugen.

5. PHP

5.1. PHP Closing Tags

- Start mit <?php Tag und beenden mit einem Kommentarfeld sowie der Location des Files
- ?> End Tags sind optional und werden nicht vom PHP Parser benötigt

Beispiel:

```
<?php

Echo “Mein Code“;

/*Ende des PHP Files test.php*/

/*Location: ./system/controler/test.php*/
```

5.2. Klassen und Methoden Namen

- Klassen Namen sollten immer mit einem Großbuchstaben beginnen.
- Multiple Namen sollten mit einem Unterstrich (_) getrennt werden
- Alle anderen Methoden Namen sollten mit Kleinbuchstaben beginnen und klar identifizierbar sein. Wenn möglich Verben verwenden
- Vermeiden von langen und komplizierten Namen.

Beispiel:

```
Class Super_class{

    function_construct()

    {

    }

}
```

5.3. Variablen Namen

- § Variablen Namen sollten analog wie Methoden Namen immer mit Kleinbuchstaben beginnen.
- § Multiple Namen werden mit einem Unterstrich (_) getrennt und werden stets mit Kleinbuchstaben fortgeführt
- § Eindeutige Namen zur Identifizierung verwenden
- § Vermeiden von langen und komplizierten Namen

Beispiel:

```
for($j = 0; $j < 10; $j++)

$str

$ buffer

$group_id

$last_city
```

5.4. Kommentare

- § DocBlock Kommentare sollen für Klassen und Methoden verwendet werden um Ihre Grundlegende Funktionsweise sowie verwendete Pakete, Kategorien, Autoren oder Links darzustellen.
- § Einfache Kommentare sollten mit dem Tag // in einzelne Zeilen eingebettet werden

5.5. Konstanten

- § Konstanten sollten ausschließlich aus Großbuchstaben bestehen

Beispiel:

```
MY_CONSTANT, NEWLINE, SUPER_CLASS_VERSION
```

5.6. TRUE, FALSE und NULL

§ Keywords TRUE, FALSE und NULL sollten stets aus Großbuchstaben bestehen

5.7. Logische Operatoren

§ Von der Nutzung des Operators II wird abgeraten, da hier die Klarheit auf verschiedenen Medien nicht gegeben ist. Stattdessen OR verwenden
§ && wird vor AND bevorzugt.
§ Ein Leerzeichen sollte zwingend einem ! folgen.

5.8. Rückgabewerte und Typecasting

§ Rückgabewerte intelligent wählen, da einige PHP Funktionen neben FALSE auch “” oder 0 bei einem Fehler zurück liefern.
§ Variablen Typ explizit wählen um beim Rückgabewert einen eindeutigen und erwarteten Output zu generieren.
§ Wenn möglich sollten === und !=== verwendet werden.

Beispiel:

```
If(strpos($str,‘foo‘)===FALSE

function ($str===“”)

{

    If($str===””)

    {

    }

}
```

5.9. Klassen und Programm Namen

§ Um Kollisionen mit anderen Klassen oder Programmen zu vermeiden, müssen Namen eindeutig und einzigartig gewählt werden. Sie dürfen nicht wiederholt vorkommen.

5.10. Private Methoden und Variablen

§ Für private Methoden sollte immer ein (_) verwendet werden.

Beispiel:

convert_text() // public method

_convert_text() // private method

Seminararbeiten

Einsteigend in die Thematik des Projektes wurde von jedem Teilnehmer eine Seminararbeit angefertigt. Die Themen der Seminararbeiten wurden zu Beginn durch die Betreuer vorgestellt, anschließend war eine freie Themenwahl für dir Teilnehmer möglich. Mit dem Beginn der Seminarphase hatten die Teilnehmer des Projektes drei Monate Zeit sich intensiv mit dem gewählten Thema auseinanderzusetzen.

VERY LARGE
BUSINESS APPLICATIONS
Carl von Ossietzky Universität Oldenburg

Visual Analytics

Ausarbeitung
im Rahmen der Projektgruppe RAPID

Themensteller:	Prof. Dr.-Ing. Jorge Marx Gómez
Betreuer:	M. Sc. Alexander Sandau
Vorgelegt von:	Philipp Schumacher 11. Semester (M. Sc. Wirtschaftsinformatik) 26129 Oldenburg 0160/90995606 philipp.schumacher@uni-oldenburg.de
Abgabetermin:	09. März 2015

Inhaltsverzeichnis

Abkürzungsverzeichnis

infoviz Information Visualization

KDD Knowledge Discovery in Databases

OLAP Online Analytical Processing

DWH Data Warehouse

DWHS Data-Warehouse-System

KNN Künstliches Neuronales Netz

Abbildungsverzeichnis

1 Einleitung

Der Fortschritt der Technik hat dazu geführt, dass es möglich ist immer größere Mengen an Daten elektronisch zu speichern [Eng09, vgl. S.3]. Gerade mittlere und größere Unternehmen, die auf ihren Daten wichtige Analysen (OLAP: Online Analytical Processing) ausführen, müssen auf dieses Problem reagieren. Bewährte Prozesse, insbesondere der des Knowledge Discovery in Databases (KDD), müssen sich verändern und die neuen Umstände angemessen zu berücksichtigen [UF96b, S. 1 ff.]. Zu bedenken ist, dass das computergestützte Ausführen von Methoden aus dem Data Mining allein längst nicht mehr angemessen ist, um die immer größer werdenden Datenberge zu beherrschen. In Zukunft gilt es immer mehr auch den Menschen, als entscheidende und steuernde Komponente, besser in den Prozess der Wissensgenerierung einzubinden.

Ein weiterer bedeutsamer Trend im analytischen Kontext ist der des In-Memory Computings. Dieser hebt in vielen Anwendungen die Grenzen zwischen transaktionalen und analytischen Systemen weitestgehend auf und beseitigt somit zunehmend auch den ETL-Prozess als Flaschenhals im Data-Warehouse-Kontext. Allerdings rentiert sich dieses Konzept im Moment lediglich bei sehr großen Unternehmen. Im Zuge des aus der zunehmenden Datenspeicherung und -nutzung resultierenden Information Overload bestehen bei den Daten laut Keim 3 Hauptprobleme [Kei08, S. 1 f.]:

- Die Daten sind irrelevant zur Aufgabenstellung
- Die Daten wurden in unangemessenen Verfahren ermittelt
- Die Daten werden in einer unangemessenen Art und Weise präsentiert

Das macht die Wichtigkeit der Anwendung der richtigen **Methoden** und **Modelle** auf den richtigen **Daten** deutlich. Das Problem welches besteht ist eher das oft fehlende Verständnis der eigenen Analyse („analysing the analysis“)[EB10, S. 18]. Gerade bei voll automatisierten Methoden, die schon seit langer Zeit im Unternehmen verwendet werden, fehlt häufig das Verständnis über deren innere Prozesse. Dieses bleibt oft nur denen vorbehalten, die die Methoden programmiert haben und über ein ausreichendes Verständnis in Bereichen wie Statistik, Programmierung und Data Mining verfügen. Das verdeutlicht wie wichtig die Dokumentation dieser Komponenten ist.

Unter Aufgreifen des klassischen KDD-Prozesses und durch Einbeziehung menschlicher Fähigkeiten, wie Intuition, Interpretationsvermögen und das Verstehen grafischer Darstellungen bietet Visual Analytics ein interessantes Konzept, um sich insbesondere dem oben beschriebenen Problem des Information Overloads (insbesondere der drei oben genannten Punkte) zu stellen. Der erste Abschnitt der Arbeit behandelt zunächst grundlegende

Konzepte und stellt einleitend den klassischen Prozess des KDDs vor. Anschließend wird Visual Analytics genauestens definiert. Darüber hinaus wird erläutert inwiefern sich dieses Konzept vom KDD unterscheidet. Danach wird der Ablauf des Prozesses an einem generischen Modell dargestellt. Dann folgt dessen Umsetzung in einem speziellen Modell von Pirolli und Card. Der darauf folgende Abschnitt zeigt auf wie Visual Analytics im Verkehrswesen eingesetzt werden kann. Danach wird ein Szenario dargestellt, welches die Vor- und Nachteile von Visual Analytics behandelt.

2 Grundlagen

2.1 Definition Online Analytical Processing (OLAP)

Laut Bauer und Günzel steht OLAP für eine Gattung von Anfragen, die nicht nur einen einzelnen Zugriff auf einen Wert, sondern einen dynamischen, flexiblen und interaktiven Zugriff auf eine Vielzahl von Einträgen erfordern. Fälschlicherweise wird OLAP oft sofort mit Data-Warehouse-Systemen assoziiert, was allerdings falsch ist, da Data-Warehouse-Systeme (DWHS) durchaus auch in einem transaktionalen Kontext stehen können. [AB09, S. 105]

2.2 In-Memory Computing

In-Memory Computing bezeichnet die vollständige Verarbeitung von zu analysierenden Daten im Hauptspeicher des Computers. Durch die Möglichkeit sehr große Datenmengen vollständig im Hauptspeicher des Computers zu speichern, kann der Zugriff auf diese ca. 100.000 Mal schneller erfolgen [BB14, S. 27]. Auf diese Weise lässt In-Memory Computing die Grenzen zwischen der transaktionalen und der analytischen Datenspeicherung verschwinden. Die Daten in den Datenbanken der Unternehmen müssen nicht erst durch den sehr zeitintensiven ETL-Prozess in eine Struktur gebracht werden, um sie in einem Data Warehouse (DWH) zu speichern. Stattdessen können die Auswertungen direkt erfolgen und deren Ergebnisse den entsprechenden Entscheidern umgehend bereitgestellt werden [BB14, S. 26].

2.3 Knowledge Discovery in Databases

Der Prozess des Knowledge Discovery in Databases (KDD) fokussiert den Prozess der Wissensgenerierung aus großen Datenbeständen. Dabei bezieht sich der Prozess insbesondere auf die Speicherung und Abfrage der Daten (häufig aus heterogenen Quellen), die Algorithmen des Data Minings zum Durchlaufen und Analysieren der Daten, die Interpretation und Darstellung der Ergebnisse des Analyse-Prozesses und darüber hinaus auf die Mensch-Maschine-Kommunikation (HCI: Human-Computer-Interaction) zur durchgehenden Unterstützung des Prozesses.[UF96a, S. 39 f.]

Abbildung 1 veranschaulicht den Prozess des KDDs mit all seinen Phasen. Im Folgenden werden diese näher beschrieben.

- **Selektion**: Dieser Phase beschreibt das Auslesen der zur Analyse notwendigen Daten aus den Quelldaten. In diesem Schritt werden die für die Analyse relevanten Daten in eine Datenbank geladen. [UF96a, S. 42]

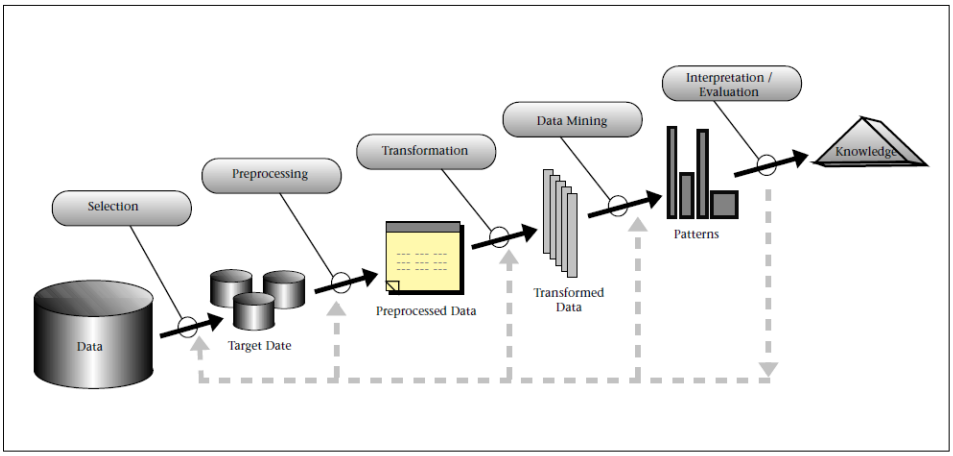


Abbildung 1: KDD-Prozess: [UF96a, S. 41]

- **Preprocessing**: Beim Preprocessing werden die Daten, die für die Analyse gedacht sind durchlaufen, reduziert, vereinheitlicht und bereinigt (Data Cleansing, Data Scrubbing, etc.). Auf diese Weise werden die Daten in eine für die Analyse geeignete Form gebracht. Oft werden in diesem Schritt Ausreißer beseitigt, Wertebereiche geglättet oder fehlende Werte ergänzt. Aber auch das Bilden von Klassen (Intervallen) zur besseren Übersicht gehört in diesen Prozessschritt [UF96a, S. 42]. Für einen eingehenderen Blick in den statistischen Hintergrund dieses Prozessschrittes empfiehlt sich für den interessierten Leser ein Blick in die folgende Literatur: [JH12, S. 83 ff.] .

- **Transformation**: Diese Phase beschreibt die Wahl und Aufbereitung der Daten für spezielle Analysezwecke. In der Regel werden die Parameter, welche für die Analyse tatsächlich nötig sind ausgewählt und überflüssige Parameter ausgelassen. Dieser Schritt ist insbesondere dafür notwendig, um ein Datenmodell zu erzeugen, welches die Anwendung bestimmter Methoden aus dem Data Mining (dem anschließenden Prozessschritt) zulässt. Modelle für die Analyse lassen sich auf diese Weise einfacher generieren und analytische Anwendungen effizienter abwickeln.

- **Data Mining**: Beim Data Mining geht es um das Aufspüren von (neuen) logischen Zusammenhängen (Mustern) in großen Datenbeständen. Beim Data Mining kommen Methoden aus den Bereichen Klassifizierung, Assoziation und Clustering zum Einsatz.[AK11, S. 555]

- **Evaluation und Interpretation**: In dieser Phase werden die Ergebnisse des Data Minings interpretiert. Auf diese Weise soll ein Lernprozess stattfinden, so dass die Möglichkeit besteht die Qualität der einzelnen Teilschritte in den nächsten Durchlaufen zu verbessern. [UF96a, S. 42]

2.3.1 Bewertung des KDD-Prozesses

Der Nachteil dieses Prozesses ist allerdings, dass er sehr linear ist und dass das Feedback erst nach dem letzten Schritt des Prozesses (Interpretation und Evaluation) möglich ist. Im KDD soll dieses Feedback zur Verbesserung der anderen Einzelschritte beitragen und somit bewirken, dass der gesamte Prozess effizienter wird. Dadurch, dass diese Rückkopplung allerdings erst im letzten Schritt erfolgt, kommt das tatsächliche Hinterfragen von **Daten**, **Methoden** und **Modellen** zu kurz. Wichtiger wäre eine durchgängige Untersuchung dieser drei Komponenten. Fällt bspw. erst in der letzten Phase auf, dass ein schlechtes Analyseverfahren zur Ermittlung der Ergebnisse verwendet wurde, hat dies in den allermeisten Fällen auch Auswirkungen auf die Auswahl der zu analysierenden Daten und des Modells welches für die Analyse verwendet wird. Fayyad selbst macht in seinem Artikel „*KDD for Science Data Analysis*“darauf aufmerksam, dass weiterhin auch eine bessere Einbindung des Menschen selbst notwendig ist:

> „*There is an urgent need to create an intermediate level at which scientists can operate effectively; isolating them from the massive sizes and harnessing human analysis capabilities to focus on tasks in which machines do not even remotely approach humans - namly, creative data analysis, theory and hypothesis formation and drawing insights into underlying phenomena.*“[UF96b, S. 1]

2.4 Definition Visual Analytics

In Anlehnung an die Definition von Keim ist Visual Analytics ein Konzept, welches die automatisierte Analyse mit interaktiven Visualisierungsmethoden vereint, um einerseits ein effektives Verständnis komplexer Datenbestände zu gewinnen und darüber hinaus die Daten, Methoden und Modelle der Analyse durchgehend zu evaluieren, um deren Wirkung innerhalb des Prozesses zu maximieren.[Kei08, S. 157]

Dabei greift Visual Analytics den klassischen KDD-Prozess auf und unterstützt mittels visueller Darstellung das Eingreifen des Menschen in den Prozess. Das soll einen besseren Fokus auf die durchgehende Evaluation der Daten, Methoden und Modelle ermöglichen und somit das strenge Vorgehen des klassischen KDDs durch ein flexibleres und

dynamischeres Modell ablösen. Somit versteht sich Visual Analytics als ein Dreiklang der Konzepte Visualisierung (information visualization), (explorativer) Datenanalyse und Interaktionsmethoden. Die unten stehende Grafik stellt Visual Analytics zudem als einen stark interdisziplinären Begriff dar. [DK06, S. 1 f.]

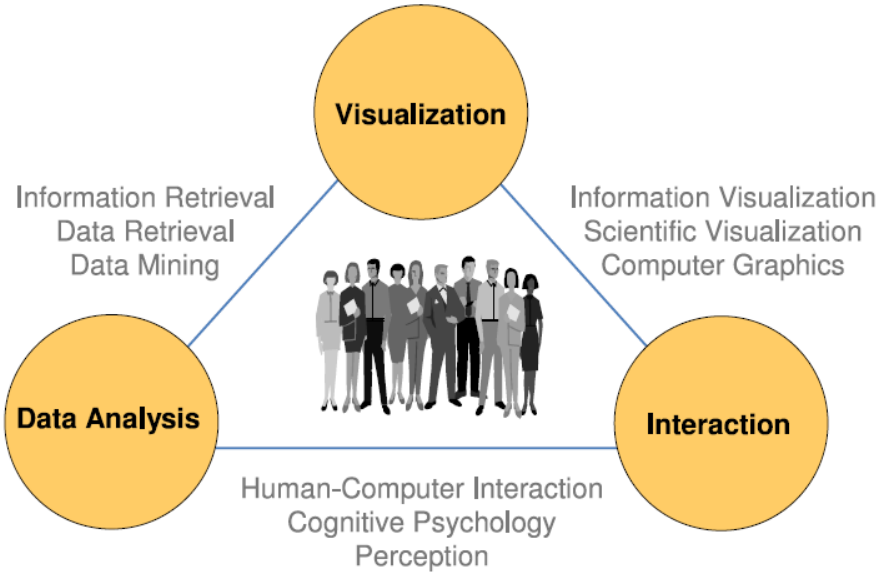


Abbildung 2: Visual Analytics - Komponenten: [DK06, S.2]

2.4.1 Explorative Datenanalyse:

Als Teilgebiet der Statistik setzt sich die explorative Datenanalyse mit der Analyse von Daten auseinander deren Zusammenhänge unklar sind. Bei dieser Form der Analyse sind die Teile der Datenbasis zu durchlaufen, in denen vielversprechendes Material aufzuspüren ist. Infolge dessen wird eine Vielzahl von Methoden aus dem Data Mining angewandt. Als spezielles Merkmal der Explorativen Datenanalyse schlug John W. Turkey das Bilden von Hypothesen über den betrachteten Daten vor [Tuk77, S. 1 ff.]. Auch in Bezug auf Visual Analytics ist das Bilden und Testen von Hypothesen von grundlegender Bedeutung, da auf diese Weise die Datenbasis nach bestimmten Vorgaben durchsucht wird.

2.4.2 Information Visualization (infoviz):

Information visualization (infoviz) setzt sich mit der Darstellung von großen Datenbeständen auseinander und hat das Ziel nützliche Informationen über visuelle Darstellungsformen komprimiert, anschaulich und verständlich darzustellen und auf diese Weise die Analyse der Daten zu vereinfachen [S. 1][EB10]. Laut Piriolli und Card nehmen diese Darstellungen in Bezug auf die Analyse eine besondere Rolle ein, da sie es Experten ermöglichen ihre besonderen Kenntnisse und Fähigkeiten bestmöglich auszunutzen:

> „*Experts don't just automatically extract patterns and retrieve their response directly from memory. Instead, they select the relevant information and encode it in special representations that allow planning, evaluation and reasoning about alternative courses of actions. [PP01, S. 2]* "

2.4.3 Interaktive Methoden:

Interaktive Methoden in Bezug auf Visual Analytics dienen dazu, dem Benutzer die Möglichkeit zu geben Darstellung interaktiv zu durchlaufen, um neues Wissen in ihnen zu suchen. Durch vordefinierte Operationen auf Darstellungen, kann der Nutzer sie weiter erforschen. [EB10, S. 1]

2.4.4 Prozess von Visual Analytics

Nachdem der vorherige Unterabschnitt sich mit dem Begriff von Visual Analytics befasst hat, widmet sich dieser Abschnitt der Verwirklichung des Konzeptes als Prozess. Im Folgenden soll deshalb ein sehr generisches Prozessmodell vorgestellt werden, welches für weitere Anpassungen offen ist. Die Abbildung 3 zeigt den Hintergrundgedanken von Visual Analytics auf. Die zu analysierenden Daten werden aus den Quelldaten ausgelesen und gelangen in die Sense Making Loop. Dort werden sie für die Entscheider entsprechend in bestimmten Darstellungen aufbereitet, so dass diese sich ein Gesamtbild von der Situation machen können. Nachdem die Entscheider durch eine adäquate Visualisierung neues Wissen gewonnen haben, kann dieses einerseits gespeichert werden und andererseits einen veränderten Eindruck in ihrer Wahrnehmung hinterlassen. Durch das Wissen werden infolge dessen konkrete Hypothesen über den aktuellen Zustand aufgestellt. Diese werden dann im Zuge der Explorativen Datenanalyse weiter untersucht, so dass eine Spezifikation zu einer weiteren Sicht der Visualisierung führt und die Sense Making Loop von neuem angestoßen wird [Kei08, S. 164 f.].

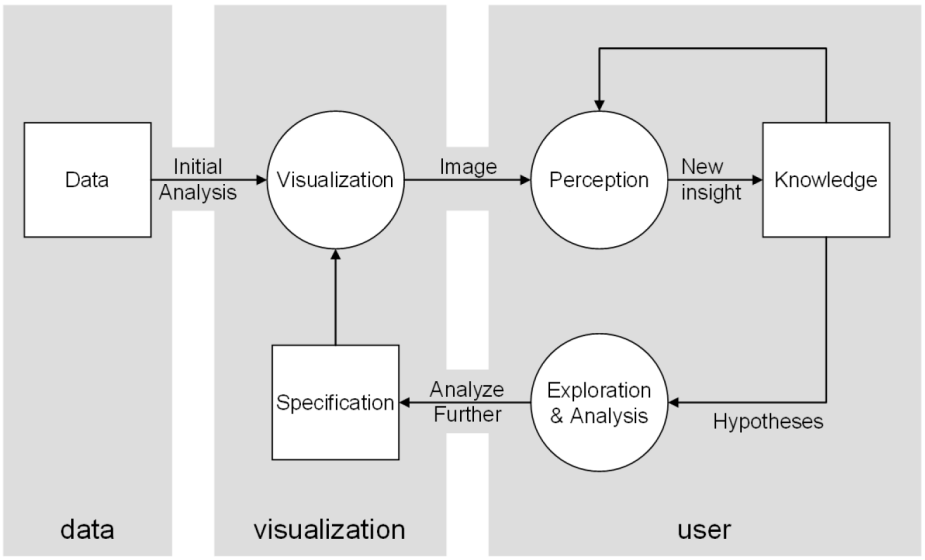


Abbildung 3: Visual Analytics-Prozessmodell: [Kei08, S. 165]

2.4.5 Spezielleres Modell von Card

Dieses Modell von Pirolli und Card beschreibt ein beispielhaftes Vorgehen zur intelligenten Analyse (intelligent analysis). Auch wenn dieses Modell nicht explizit für Visual Analytics vorgesehen ist, lassen sich einige Parallelen ziehen. Die für die Analyse relevanten Daten werden von den Quellen in die **shoebox** geladen. Von dort aus werden verschiedene Ausschnitte der Daten in die **evidence-file**-Komponente geladen. Dieser Prozess findet in der sog. textbfforaging loop iterativ mit Vorwärts- und Rückwärtskopplungen statt. Im nächsten Schritt werden die Daten in **Schemata** geladen, welche es den Experten möglich machen Hypothesen zu bilden. Diese Hypothesen werden auf konkreten Präsentationsmodellen nach außen kommuniziert. Der Prozess vom schema bis zur presentation findet in der **sense making loop** statt. Der Aufwand dieses Verfahrens und die Struktur der Daten nimmt nach jeder Prozessstufe zu. [PP01, S. 2 ff.]

Bewertung des Modells von Pirolli und Card: Auch wenn dieses Modell nicht speziell für den Visual-Analytics-Prozess vorgesehen ist, lassen sich jedoch viele Parallelen zum zuvor dargestellten Modell erkennen. Anderes jedoch als beim Modell von Wjik schlägt dieses Modell ein wesentlich flexibleres Vorgehen bei der Auswahl der Daten für die Analyse vor. Dies ist gerade im Hinblick auf die Wichtigkeit der Datenbasis für die weitere Analyse ein

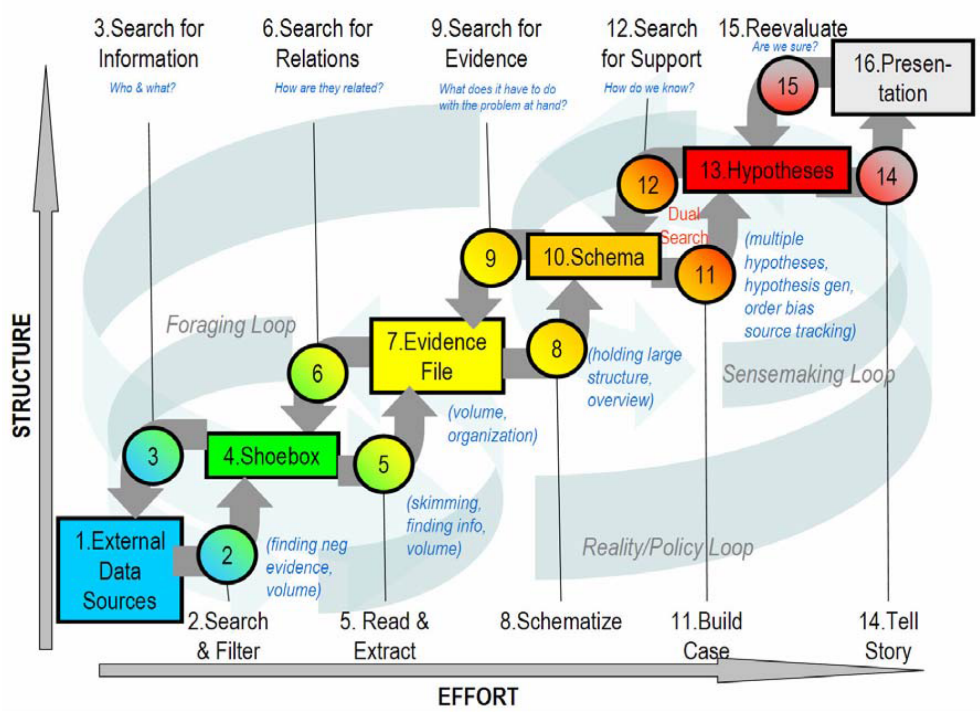


Abbildung 4: Prozess der intelligence analysis: [PP01, S.3]

bedeutender Unterschied.

3 Visual Analytics im Verkehrswesen

In diesem Abschnitt wird die Anwendung von Visual Analytics im Verkehrswesen illustriert. Dazu wird insbesondere auf die eingangs behandelten Komponenten **Daten**, **Methoden** und **Modelle** eingegangen. Natürlich kann diese Arbeit nicht die gesamt Vielfalt der Verkehrsanalyse aufzeigen. Dazu sei der interessierte Leser verwiesen auf folgende Quelle: [Eng09, S. 1].

3.1 Wichtigkeit der Analyse des Verkehrswesens

Der Straßenverkehr ist ein hochkomplexer und durch stochastische Zusammenhänge beeinflusster Prozess ([Eng09, S. 1] in Anlehnung an [Hel13, S. 7]).

Dies wird deutlich unter Betrachtung der Tatsache, dass besonders im urbanen (städtischen Verkehr) viele Verkehrsteilnehmer mit eigenen individuellen Verhaltensweisen im Verkehrsgeschehen agieren. Unter Einflussnahme bestimmter ausgestellter Verkehrsregeln, induziert durch Verkehrsschilder, Ampelschaltungen o.ä., sind die Geschehnisse des Straßenverkehrs in vielen Fällen nur schwer bis gar nicht vorherzubestimmen. Das Verkehrsmanagement hat die Aufgabe den Verkehrsfluss zu optimieren und Störungen zu beseitigen. [DKB05, S. 1]

Der Analyse von Verkehrsdaten kommt somit eine besondere Bedeutung zu. Sie hat die Aufgabe, die Geschehnisse im Straßenverkehr zu erfassen und mittels geeigneter Methoden Besonderheiten wie Gefährdungen, Staus, etc. vorherzusagen und den Entscheidungsträgern eine geeignete Grundlage zur Einleitung von Entscheidungen zu geben.

3.2 Daten im Verkehrswesen

Für die Analyse relevante Daten im Verkehrswesen gibt es sehr zahlreich und in verschiedener Form. Welche Daten für die Analysen eine Rolle spielen, steht in Abhängigkeit von dem Zweck für den sie benötigt werden. Verkehrsdaten können in relationalen Datenbanken vorliegen. Noch häufiger kommen die Daten allerdings in Form unstrukturierter Textfiles vor [Eng09, S. 5 f.]. Oft werden Daten von den GPS-Systemen der Verkehrsteilnehmer oder Signaldaten über Ampelschaltungen analysiert. Des weiteren stammen Daten ebenfalls von Infrarot-, Video- und Induktivschleifendetektoren ([Eng09, S. 41 f.] in Anlehnung an [Tec08, S. 20]). Durch geeignete Methoden, die im Folgenden angesprochen werden, sollen diese Daten bereinigt, aufbereitet und relevanten Entscheidungsträgern zur Verfügung gestellt werden können.

3.3 Methoden im Verkehrswesen

Gerade für die Vorhersage von Ereignissen im Verkehr ist es notwendig verschiedene Dimensionen zu betrachten (Zeit, Fahrzeugaufkommen, Wetter, Straßenverhältnisse, etc.). Diese Vielfalt erfordert in der Regel effiziente Methoden aus dem Data Mining, die in der Lage sind eine große Anzahl von Datensätzen zu durchlaufen. Im paper „*traffic accident analysis using decision trees and neural networks* "werden Entscheidungsbäume und Künstliche Neuronale Netze (KNNs) zur Klassifikation von Autounfällen verwendet [Bui06]. Regressionsanalysen können bspw. dafür dienen einen Zusammenhang zwischen der Tageszeit und dem Verkehrsaufkommen an bestimmten Straßen zu ermitteln. Diese Form der Analyse wird auch weiter unten noch einmal erwähnt. Insbesondere die Clusteranalyse wird in der Arbeit von Engelmann behandelt. Sie bietet u.a. eine Möglichkeit Straßengruppen zu erfassen, auf denen ähnliche Verkehrsausprägungen (Geschwindigkeitsprofil, Verkehrsqualität etc.) bestehen [Eng09, S. 40].

3.4 Modelle im Verkehrswesen

Modelle im Verkehrswesen behandeln verschiedene Objekte, die für die Analysen in diesem Kontext relevant sind. Dies können Ampelschaltungen aber auch Haltezeiten von Fahrzeugen an bestimmten Orten sein. Eine Übersicht über diese und weitere interessante Darstellungen im Bereich von Ampeldaten bietet das folgende Dokument: [AG12]. Gerade im Verkehrswesen sind Modelle notwendig, die eine Vielzahl von Daten zusammenfassen und den Entscheidern eine geeignete Basis bieten gute Entscheidungen zu treffen, die sich positiv auf den Verkehrsfluss, die Sicherheit der Verkehrsteilnehmer etc. auswirken.

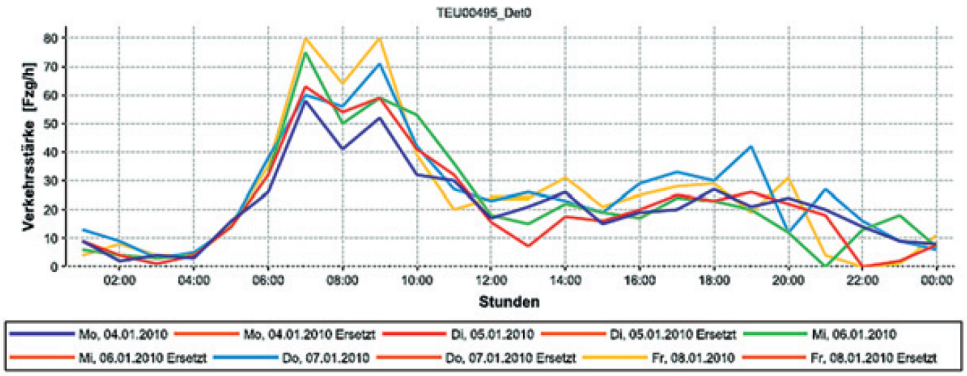


Abbildung 5: Diagramm zur Darstellung der Verkehrsdichte: [AG12, S. 16]

Abbildung 5 zeigt ein Diagramm, welches die stündliche Uhrzeit und die Verkehrsstärke als Fahrzeug pro Stunde anzeigt. Mit Hilfe verschiedenfarbiger Linien kann der Abgleich aller Wochentage stattfinden. Dabei lässt sich die zeitliche Dimension unterschiedlich aggregieren (also auch auf Minuten, Stunden, etc.). Auf diese Weise sind interaktive Operationen wie Roll-Up und Drill-Down möglich und erlauben deshalb eine angenehmere und flexiblere Navigation.[AG12, S. 16]

Eine wesentlich speziellere Darstellungsform, zeigt Abbildung 6. Auch in diesem Diagramm wird die Verkehrsdichte in Abhängigkeit zur Zeit dargestellt. Des Weiteren findet aber auch eine Abgrenzung darüber statt welche der Fahrzeuge PKWs und welche LKWs sind. Darüber hinaus wird die stündliche Fahrzeuggeschwindigkeit beider Typen modelliert und der Verlauf als Übergangslinie dargestellt [AG12, S. 16].

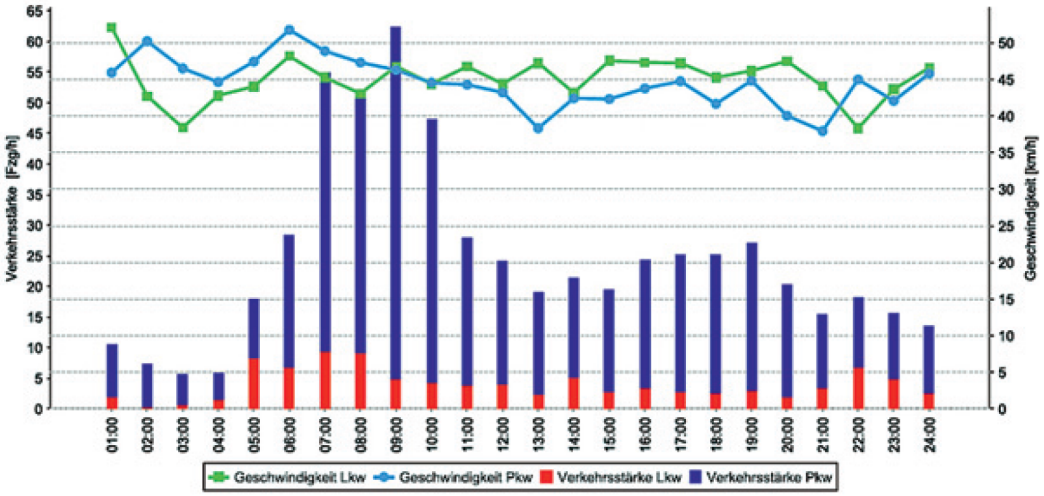


Abbildung 6: Verkehrsdichte (andere Darstellungsform): [AG12, S. 16]

Eine weitere häufig genutzte Darstellungsform zur Visualisierung ist die Heatmap. Mit dieser Darstellungsvariante lassen sich unterschiedlich stark verwendete Verkehrsregionen aufspüren. Abbildung 7 zeigt die Verläufe von Fahrradrouten in einer Heatmap an. Realisiert wurde das, indem eine große Anzahl von Fahrradfahrern mit GPS-Trackern ausgestattet wurden [END14].

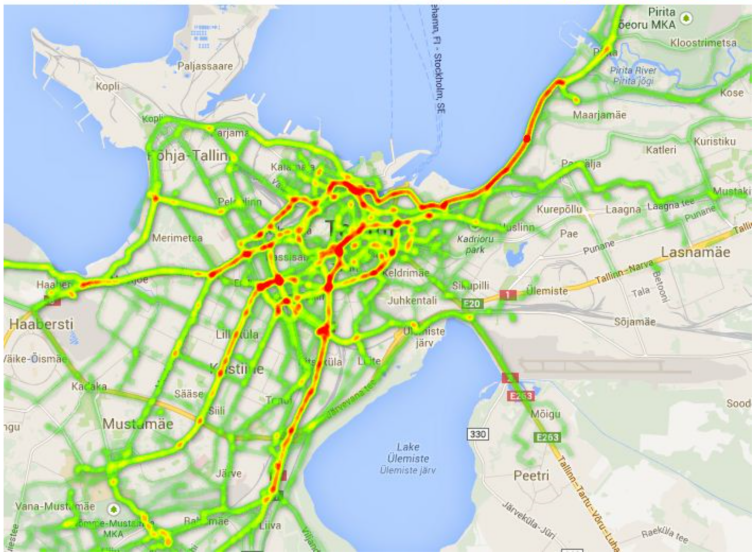


Abbildung 7: Verkehrrouten als Heatmap: [END14]

3.5 Szenario zur Illustration

Die Bürger des Ortes O sind unzufrieden mit dem Verkehrsfluss auf der Hauptstraße H. Dieser scheint in letzter Zeit wesentlich zäher als sonst zu verlaufen. Im Zuge einer Initiative wurden gemeinschaftlich Unterschriften gesammelt, um sich auf diese Weise für den Ausbau der Straße einzusetzen. Dieses Engagement bewirkte letzten Endes tatsächlich, dass sich auch die oberste Ebene mit diesem Thema befasste. Da der Ausbau der Hauptstraße mit erheblichen Kosten verbunden ist, wurde zunächst beschlossen im Zuge einer Verkehrsflussanalyse den Verkehr mittels geeigneter Methoden zu untersuchen. Mit Hilfe einer Visual-Analytics-Anwendung werden dazu Daten das dem Verkehrsmanagement untersucht. Diese Daten stammen u.a. von Sensoren, die sich an verschiedenen Stellen der Hauptstraße befinden und dort rund um die Uhr das Verkehrsaufkommen in regelmäßigen Abständen messen. Für die Analysen werden die Sensordaten im Zeitraum 2014 geladen. Auf der Benutzeroberfläche werden dann die Zeitdimension und die Sensordaten-Dimension ausgewählt. Zunächst wird nur der Zeitraum von Januar bis März betrachtet in dem es zu Beschwerden kam. Das System generiert automatisch ein Verkehrsflussdiagramm (x-Achse: Zeit in Tagen, y-Achse: Verkehrsdichte in Fahrzeugen pro Zeiteinheit). Zu erkennen ist, dass die Werte definitiv höher liegen als damals.

In einem weiteren Durchlauf wird entschieden die Werte über den gesamten Zeitraum 2014 abzubilden. Zur besseren Übersicht zeigt das System diesmal nur monatliche Mit-

telwerte an. Auch diese Darstellung offenbart keine neuen Erkenntnisse, denn auch hier erscheint der Verkehrsfluss hoch.

Die Analyse führt dazu, dass es zu einem Ausbau der Hauptstraße kommt. Wie kalkuliert zieht dies beträchtliche Kosten mit sich. Bereits wenige Monate nach dem Ausbau stellt sich allerdings heraus, dass der damals schlechte Verkehrsfluss durch eine Straßensperrung auf einer anliegenden Strecke herbeigeführt wurde und durch eine zeitweise Geschwindigkeitsbegrenzung auf einer weiteren anliegenden Straße. Besser als der Ausbau der Hauptstraße wäre es sicherlich gewesen, den Verkehr zu diesen Zeiten umzuleiten bzw. einfach den Zeitraum in denen diese Einschränkungen stattgefunden haben abzuwarten.

3.5.1 Bewertung des Szenarios

An diesem Szenario lässt sich relativ leicht die Wichtigkeit einer angemessenen Datenbasis erkennen. Im obigen Fall fehlte eindeutig eine vorherige Analyse. Weiterhin unangemessen war, dass keine Informationen über die Sperrung und die Geschwindigkeitseinschränkung verfügbar waren. Faktoren wie diese hätten aber zuvor unbedingt berücksichtigt werden sollen. Das Analyseverfahren lässt sich deshalb rückblickend als ein Verfahren betrachten, bei dem der Mensch einzig und allein mit Informationen des Analyseprogramms versorgt wurde. Informationen von anderen Verkehrsinstitutionen waren nicht vorliegend und hätten auch nachträglich nicht in die Analyse miteinbezogen werden können. Darüber hinaus wäre es besser gewesen, die Daten von anderen Straßen miteinzubeziehen. So hätten bspw. auch Sensordaten von an den Ort anliegenden Landstraßen entscheidende Hinweise liefern können.

Auch die Methoden, die zur Analyse der Daten verwendet wurden, waren nicht gut gewählt. Eine Regressionsanalyse hätte hilfreiche Aufschlüsse darüber gegeben, dass das Verkehrsaufkommen der Hauptstraße stark mit der Verkehrsdichte der anderen beiden Verkehrsbereiche zusammenhängt. In diesem Fall wäre es sogar möglich gewesen auf ein sinnvolles Ergebnis zu kommen, ohne von der Sperrung und der Geschwindigkeitseinschränkungen jemals was mitbekommen zu haben.

Dies hätte sich auch in punkto Darstellungsformen als hilfreich erwiesen. Das Erforschen der Daten mit der Roll-Up-Methode auf Monatswerte erwies sich als unangemessen. Ein Drill-Down auf eine tiefere Ebene (Tageswerte oder gar Stundenwerte) hätte u.U. auch Aufschluss darüber geben können, dass es an bestimmten Tagen, zu bestimmten Zeiten einen wesentlich besseren Verkehrsfluss gab. Auf diese Weise wäre es möglich gewesen zu merken, dass es an Tagen an denen die Baustelle nicht in Betrieb war, zu einem besseren Verkehrsfluss kam.

4 Zusammenfassung

Die vorliegende Arbeit befasst sich mit Visual Analytics. Der vordere Grundlagenteil beschäftigte sich mit der Klärung notwendige Begriffe wie Visual Analytics selbst aber auch weitere für das Verständnis notwendige Begriffe und Konzepte, wie der KDD-Prozess, Data Mining oder In-Memory Computing wurden erläutert. Außerdem wurde eine Abgrenzung zweier Vorgehensmodelle (generisches Modell nach Wijk und das Modell zur intelligence analysis von Pirolli und Card) durchgeführt. Anschließend wurden auf konkrete Daten, Methoden und Modelle eingegangen, die im Zuge der Verkehrsdatenanalyse eine wichtige Rolle spielen. Danach wurde deren Wichtigkeit in einem Szenario illustriert.

5 Fazit

Abschließend lässt sich feststellen das Visual Analyitics ein fälliges Konzept ist. Gerade die Flexibilität der in dieser Arbeit immer wieder angesprochenen Komponenten Daten, Methoden und Modelle ist wichtig. Diese wird zunehmend dadurch gefördert, dass immer mehr der Mensch als entscheidende und wertende Instanz eingebunden wird. Dieser soll nicht nur die Instanz darstellen, die Analysen per Knopfdruck ausführt, sondern auch komplexe Analyseprozesse mitverantwortlich steuert. Die Fähigkeiten von Maschinen werden also zunehmend mit denen des Menschen angereichert, um auf diese Weise einen immer stärker werdenden Zweiklang zwischen Mensch und Technik zu schaffen. Gerade die Betrachtung des generischen Prozesses nach Wijk und des Szenarios zeigte, dass die beste Visual-Analytics-Anwendung jedoch ohne die richtige Datenbasis wertlos ist. Mehr Flexibilität bei den Daten, sowie sie bei dem Modell von Pirolli und Card zumindest angedeutet wird, kann nur erfolgen wenn der Prozess der Datenbeschaffung, die Dokumentation der Daten und der verwendeten Methoden und die Organisationsstrukturen rund um die Analyse selbst angemessen berücksichtigt werden. Visual Analytics als Konzept schlägt mit dem stärkeren Einbeziehen des Menschen eindeutig die richtige Richtung ein. Allerdings sollte die Berücksichtigung des Menschen nicht nur auf Grundlage rein technischer Gegebenheiten (verbesserte Intelligenz der Programme, schnellere Laufzeit bei der Analyse oder bessere Bedienbarkeit über die grafische Oberfläche) stattfinden.

Schwer wird es in dieser Hinsicht Programme nach ihrer Tauglichkeit für Visual Analytics zu bewerten, denn es müsste auch darauf eingegangen werden, inwieweit die Anwendung in der Lage ist die Organisationsstruktur des Unternehmens aufzugreifen oder inwiefern es die Dokumentation und Bewertung der verwendeten Daten, Methoden und Modelle berücksichtigt. Sicherlich besteht an dieser Stelle erheblicher Forschungsbedarf. Gerade deshalb bleibt abzuwarten, wie sich Visual Analytics als Konzept weiterentwickeln wird und welchen Anklang es in den nächsten Jahren in Unternehmen findet.

Literatur

[AB09] Andread Bauer, Holger G. ; 3 (Hrsg.): *Data Warehouse Systeme - Architektur, Entwicklung, Anwendung.* dpunkt.verlag, 2009

[AG12] AG, Siemens: *Verkehrsdatenanalyse in Sitraffic Scala/Concert - Das Expertensystem für Visualisierung, Qualitätsmanagement und Statistik.* http://www.mobility.siemens.com/mobility/global/SiteCollectionDocuments/de/road-solutions/urban/traffic-control-center/verkehrsdatenanalyse.pdf. Version: 2012, Abruf: 02.03.2015

[AK11] Alfons Kemper, André E. ; 8 (Hrsg.): *Datenbanksysteme - Eine Einführung.* Oldenbourg Verlag, 2011

[BB14] Bjarne Berg, Penny S. ; 3 (Hrsg.): *SAP HANA - An introduction.* Gallieo Press, 2014

[Bui06] Bui, A.: *Innovative Internet Community Systems: 5th International Workshop, IICS 2005, Paris, France, June 20-22, 2005. Revised Papers.* Springer, 2006 (Lecture Notes in Computer Science / Information Systems and Applications, incl. Internet/Web, and HCI). https://books.google.de/books?id=dQ2uhVzxBh0C. – ISBN 9783540339731

[DK06] Daniel Keim, Andreas Stoffel Hartmut Z. Florian Mansmann M. Florian Mansmann: Visual Analytics. (2006). http://bib.dbvis.de/uploadedFiles/36.pdf, Abruf: 20.02.2015

[DKB05] Dr. Klaus Bogenberger, Steven B. Prof. Dr. Robert Bertini B. Prof. Dr. Robert Bertini: Analytische Methoden zur Interpretation von Verkehrsdaten. (2005). http://www.bmwgroup.com/publikationen/d/2005/pdf/analytischen_methoden_2005.pdf, Abruf: 05.03.2015

[EB10] Enrico Bertini, Denis L.: Investigating and Reflecting on the Integration of Automatic Data Analysis and VisualiVisual in Knowledge Discovery. (2010), Januar

[END14] ENDURANCE: *Partition 2.0.* http://www.epomm.eu/newsletter/v2/eupdate.php?nl=0414_2&lan=de. Version: April 2014, Abruf: 03.03.2015

[Eng09] Engelmann, Stefan: *Data Mining zur Analyse von Verkehrsdaten*. `http://uni.stoffelito.de/Diplomarbeit.pdf`. Version: Februar 2009, Abruf: 01.03.2015

[Hel13] Helbing, D.: *Verkehrsdynamik: Neue physikalische Modellierungskonzepte*. Springer Berlin Heidelberg, 2013 `https://books.google.de/books?id=H3geBgAAQBAJ`. – ISBN 9783642590634

[JH12] Jiawei Han, Jian P. Mecheline Kamber K. Mecheline Kamber ; 3 (Hrsg.): *Data Mining - Concepts and Techniques*. Morgan Kaufmann, 2012

[Kei08] Keim, Daniel: Visual Analytics: Definition, Process and Challenges. (2008), April

[PP01] Peter Pirolli, Stuart C.: The Sensemaking Process and Leverage Points for Anaylst Technology as Identified Through Cognitive Task Analysis. (2001), S. 6

[Tec08] Technologie, Deutschland B.: *Verkehrsmanagement und Verkehrstechnologien: Mobile Zukunft mit intelligenten Verkehrssystemen*. PRpetuum GmbH, 2008 (Innovationspolitik, Informationsgesellschaft, Telekommunikation). `https://books.google.de/books?id=8n-1YgEACAAJ`

[Tuk77] Tukey, J.W.: *Exploratory Data Analysis*. Addison-Wesley, 1977 (Addison-Wesley series in behavioral science: quantitative methods). `https://books.google.de/books?id=XaZztQAACAAJ`

[UF96a] Usama Fayyad, Padhraic S. Gregory Piatetsky-Shapiro: From Data Mining to Knowledge Discovery in Databases. In: *American Association for Artificial Intelligence* (1996)

[UF96b] Usama Fayyad, Paul S. David Haussler H. David Haussler: KDD for Science Data Analysis: IIssue and Examples. In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining* (1996). `https://genomics.soe.ucsc.edu/sites/default/files/fayyad96kdd.pdf`, Abruf: 07.03.2015

Abschließende Erklärung

Ich versichere hiermit, dass ich diese Ausarbeitung mit dem Titel Visual Analytics selbständig und ohne fremde Hilfe angefertigt habe, und dass ich alle von anderen Autoren wörtlich übernommenen Stellen wie auch die sich an die Gedankengänge anderer Autoren eng anlegenden Ausführungen meiner Arbeit besonders gekennzeichnet und die Quellen zitiert habe.

Oldenburg, den 8. März 2015 Philipp Schumacher

Die Bedeutung von In-Memory für Datenbankparadigmen OLTP und OLAP

Seminararbeit
im Rahmen des Projektes
Regional Analysis and prediction
Platform by In-Memory Data (RAPID)

Abteilung Wirtschaftsinformatik I:
Very Large Business Applications

Themensteller:	Prof. Dr.-Ing. Jorge Marx Gómez
Betreuer:	Dipl. Oec. Benjamin Wagner vom Berg
Vorgelegt von:	Janine Haase Weidenstraße 18 26389 Wilhelmshaven 0176 / 98 37 66 54 janine.haase@uni-oldenburg.de
Abgabetermin:	09. März 2015

Inhaltsverzeichnis

Abkürzungen

AKID Atomarität, Konsistenz, Isoliertheit, Dauerhaftigkeit

BI Business Intelligence

CO2 Kohlenstoffdioxid

DML Datenmanipulationssprache

DW Data Warehouse

FASMI Fast Analysis of Shared Multidimensional Information

HANA High Performance Analytic Appliance

HOLAP Hybrides Online Analytical Processing

MDX Multidimensional Expressions (Datenbanksprache für OLAP

MOLAP Multidimensionales Online Analytical Processing

MPP Massive Parallel Processing

OLAP Online Analytical Processing

OLTP Online Transaction Processing

RAM Random Access Memory

RAPID Regional Analysis and prediction Platform by In-memory Data

ROLAP Relationales Online Analytical Processing

SAP Systeme, Anwendungen, Produkte (Unternehmen SAP AG)

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einleitung

Wachsende Volumina von Daten und Einzelinformationen, steigende Anzahl von Informationsquellen und zunehmendes Verlangen nach größtmöglicher Aktualität von Datenanalysen und Datenverarbeitung sind Begriffe, die bereits seit langem Anforderungen an Data Warehouses und damit verbundene Technologien prägen. Die Anforderungen führten bislang zu einer strikten Trennung von sogenannten operativen transaktionalen und analytischen Systemen, welche lange Zeit unantastbar blieb.

Spätestens seit jedoch SAP mit SAP HANA diesem Markt beigetreten ist, ist das Thema In-Memory-Computing in aller Munde. Es wird ein Paradigmenwechsel – weg von OLAP- und OLTP-Systemen – vorhergesagt.

Nach deren strikter Differenzierung eröffnet In-Memory-Computing neue Möglichkeiten, OLTP und OLAP „zumindest ein gehöriges Stück näher zusammenzubringen, wenn nicht gar in naher Zukunft miteinander zu verschmelzen.“ ([Sim12], S. 3) Können in einer gemeinsamen Speichertechnologie Transaktions- und Analysedaten abgelegt werden, werden Datenanalyen auf Basis aktueller operativer Daten in Echtzeit durchführbar, was bislang durch die Trennung nur schwer möglich war. (Vgl. [Sim12], S. 3f)

Die folgende Arbeit wird sich mit diesem Themenbereich befassen. So werden erst die Datenbankparadigmen OLAP und OLTP mit ihren Regelwerken erklärt und anschließend gegenübergestellt. Es wird herausgestellt, warum die beiden Systeme physisch getrennt voneinander arbeiten müssen. Anschließend wird die In-Memory-Technik beschrieben, in deren Zusammenhang die spalten- und zeilenorientierte Datenspeicherung dargestellt und deren Vor- und Nachteile kurz aufgeführt werden. Die Bedeutung von In-Memory für OLAP und OLTP soll verdeutlicht werden, abschließend wird die im Projekt „Regional Analysis and prediciton Platform by In-memory Data“ – kurz RAPID – zu nutzende In-Memory-Technologie in Zusammenhang zum Projekt gebracht und einige Anwendungsfallideen vorgestellt.

Ziel dieser Arbeit ist es, dem Leser einen Überblick über OLAP, OLTP und In-Memory zu geben und einen Bogen zum Projekt RAPID zu schlagen, um dieses in seiner weiteren Ausarbeitung mit Hintergrundwissen und Anwendungsfallideen zu unterstützen.

2 Grundlagen für OLAP und OLTP

In diesem Abschnitt werden die OLAP- und OLTP-Technologien sowie ihnen zugrunde liegende Regelwerke kurz beschrieben. Abschließend findet ein Vergleich der beiden Systeme statt, der ebenfalls erklärt, aus welchem Grund die beiden Systeme nicht auf der gleichen physischen Datenbank sondern getrennt ausgeführt werden sollten.

2.1 OLAP

Die Abkürzung OLAP steht für Online Analytical Processing (Vgl. [Bri13], S. 132) und ist ein „Konzept für die im Dialogbetrieb realisierte Verdichtung und Darstellung von managementrelevanten Daten aus einem Data Warehouse[1].“ ([Bri13], S. 133) *Online* kennzeichnet hierbei den direkten Zugriff auf den Datenbestand einer zentralen Datenablage zur Datenbetrachtung und Daten(-sichten-)manipulation. *Analytical* steht für die Bereitstellung unterschiedlicher Sichten auf die vorhandenen Daten, die dann beispielsweise zur Entscheidungsfindung in einem Unternehmen genutzt werden können. Processing stellt das Konzept dar, welches schnelle Berechnungen und Datenmanipulationen in den Vordergrund stellt und nicht die Methoden von Datenhaltung oder -speicherung, z.B. eines Data Warehouses.

Ein meist nach dem Starschema aufgebauter Datenwürfel (englisch: cube) mit Faktentabelle und zugehörigen Dimensionstabellen als Satellitentabellen bildet die OLAP-Struktur. (Vgl. [SV08], S. 19) Die Faktentabellen enthalten dabei kalkulierbare Daten, Dimensionstabellen beschreibende Daten. „Sie definieren Suchkriterien und legen entlang von Hierarchien Verdichtungsstufen fest. Diese Multidimensionalität wird als zentrales Charakteristikum von OLAP angesehen.“([SV08], S. 20)

Zentrale Forderung an OLAP-Systeme liegt dazu in der schnellen Beantwortung komplexer, lesender Anfragen, um eine Datengrundlage für analytische Auswertungen bieten zu können. (Vgl. [SV08], S. 24)

Im Vordergrund stehen bei OLAP daher „dynamische und multidimensionale Analysen auf historischen, konsolidierten Datenbeständen.“ ([SV08], S. 19) Es wird als eine hypothesengestützte Methode bezeichnet: Vor einer eigentlichen Untersuchung müssen die

[1] Data Warehouse: Datenbank ausschließlich mit Lesezugriff, die zwar von operativen Datenverarbeitungssystemen separiert ist, aber regelmäßig mit deren Daten aktualisiert wird. Die Daten werden im Data Warehouse „zusammengetragen, vereinheitlicht, nach Nutzungszusammenhängen geordnet, verdichtet und dauerhaft“ ([Bri13], S. 35) in dessen Datenbasis archiviert. Das Ziel ist die Unterstützung strategischer Unternehmensentscheidungen durch eine umfangreiche unternehmensinterne Informationsversorgung. ([Bri13], S. 35)

Anfragen an das System bekannt sein, durch das entstehende Analyseergebnis wird dann die vorher aufgestellte Hypothese bestätigt oder negiert. (Vgl. [SV08], S. 19)

OLAP-Anwendungen verarbeiten sehr große Datenmengen. Dabei greifen sie auf historische Daten zu, die oft in unterschiedlichen Datenquellen liegen (Vgl. [Rög10], S. 28f): Analytische Daten werden dabei aus operierenden Daten und externen Quellen extrahiert und anschließend durch Transformation, Aggregation, Gruppierung und Speicherung unter anderem mithilfe von OLAP-Operatoren[2] abgeleitet und in einem Data Warehouse (DW) gespeichert. (Vgl. [Mül13], S. 4) Durch die Überführung ins Data Warehouse werden diese Daten verdichtet und transformiert, wodurch OLAP-Analysen dann in für den Nutzer adäquater Zeit durchgeführt werden können. (Vgl. [Rög10], S. 28f) Das Data Warehouse Konzept wird in Abbildung 1 dargestellt.

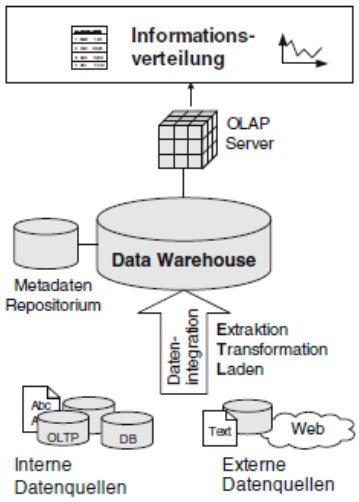


Abbildung 1: Das Data Warehouse-Konzept [Mül13], S. 15

Die lesenden Zugriffe der OLAP-Analysen ändern die Daten nicht, es wird daher kein Transaktionskonzept benötigt, die Daten in den Systemen werden jedoch „in regelmäßigen Abständen aus meist relationalen Datenbanksystemen aktualisiert.“ ([Rög10], S. 28f)

Bei OLAP wird zwischen drei Speicherarten unterschieden:

- Relationales OLAP (ROLAP)

[2]OLAP-Operatoren dienen unter anderem zur Selektierung, Weiterverarbeitung, Projektion oder Aggregation von Daten innerhalb eines multidimensionalen Datenraumes. Beispiele für Grundlegende OLAP-Operatoren sind Slicing zur Selektion, Dicing zur Projektion oder Roll-up zur Aggregation von Daten. (Vgl. [Mül13], S. 53 f)

- Multidimensionales OLAP (MOLAP)
- Hybrides OLAP (HOLAP)

ROLAP benutzt zur Zellenspeicherung eines Datenwürfels relationale Datenbanken, Haupteigenschaften sind Skalierbarkeit, eine geringe Performanz und keine Redundanzen. Im *MOLAP* „wird der OLAP-Datenwürfel in einem speziellen proprietären Format gespeichert." ([Mül13], S. 58) Aggregate einzelner Dimensionen und Teilwürfel werden in dem Format vorberechnet gespeichert, was kurze Antwortzeiten ermöglicht. Nachteil hiervon ist eine redundante Datenhaltung, damit verbunden auch ein hoher Speicherbedarf und hohe Kosten für entsprechende proprietäre Technologien. Bei *HOLAP* werden nur aggregierte Daten in MOLAP gespeichert, einzelne Fakten in relationale Tabellen geschrieben. Damit stellt *HOLAP* einen Kompromiss zwischen ROLAP und MOLAP dar. (Vgl. [Mül13], S. 58)

OLAP-Systeme stellen die Datenbasis für anspruchsvolle Analysen und sind in allen betrieblichen Funktionen und Bereichen einsetzbar. Sie dienen der Informationsversorgung von Fach- und Führungskräften und können überall da eingesetzt werden, wo dispositive bzw. analytische Aufgaben in Organisationen zu lösen sind. (Vgl. [Cha06], S. 151f)

2.1.1 Codds Rules

Codd's Rules aus dem Jahr 1993 stellen ein Leitbild mit zwölf Evaluationsregeln dar, die, wenn sie erfüllt werden, die OLAP-Fähigkeit von Informationssystemen garantieren sollen. Die Regeln wurden von Chamoni und Gluchowski ausführlich beschrieben, die folgende Reihenfolge lehnt sich an die benannten Autoren und soll lediglich einen kurzen Überblick verschaffen.

1. *Mehrdimensionale konzeptionelle Perspektiven*, logische Sichten auf für Entscheidungen relevante Zahlengrößen sollten multidimensional sein
2. *Transparenz* für leichte Anwendbarkeit von OLAP-Werkzeugen und durch einheitliche optische Gestaltungskriterien von Informationen
3. *Zugriffsmöglichkeiten* durch eine offene Architektur und damit die Unterstützung von Datenzugriffen auf möglichst viele interne und externe Datenformate
4. *Stabile Antwortzeiten bei der Berichterstattung,* auch bei Erhöhung von Dimensionen und/oder Datenvolumen

5. *Client-/Server-Architektur* zur Verarbeitung großer Datenmengen

6. *Grundprinzipien der gleichgestellten Dimensionen* mit einheitlichem „Befehlsumfang zum Aufbauen, Strukturieren, Bearbeiten, Pflegen und Auswerten der Dimensionen“ ([Cha06], S. 148)

7. *Dynamische Verwaltung „dünnbesetzter“ Matrizen* für eine optimale Datenspeicherung ohne Beeinträchtigung mehrdimensionaler Datenmanipulation durch eine Kombination unterschiedlicher Datenorganisationen.

8. *Mehrbenutzerfähigkeit* für Berechtigungen und gleichzeitige Nutzung der Systeme (für das Projekt RAPID nicht relevant)

9. *Unbeschränkte kreuzdimensionale Operationen über Dimensionen hinweg* durch eine vollständige und integrierte Datenmanipulationssprache (DML) und ein in Bezug auf Abfragemöglichkeiten offenes System. Es müssen sowohl Elementaggregation innerhalb einer Dimension als auch Datenelemente verknüpfende Verfahren zwischen mehreren Würfeln zur Verfügung stehen.

10. *Intuitive Datenmanipulation* durch ergonomische und einfache Benutzerführung und -oberfläche

11. *Flexibles Berichtswesen* durch vorformulierte Standardauswertungen und dynamisch erzeugbare Auswertungen

12. *Unbegrenzte Dimensions- und Aggregationsstufen* als Maximalziel für ein OLAP-System, dazu sollen Verdichtungsebenenanzahl und -art nicht eingeschränkt sein.

Kritik an den genannten Regeln besteht darin, dass die fachlich-konzeptionellen Anforderungen und technische Realisierungsaspekte nicht eindeutig voneinander getrennt sind. Es ist unklar, „ob die konzeptionellen mehrdimensionalen Datensichten auch eine zwingende Nutzung spezieller Speicher- und Datenverwaltungstechniken impliziert oder ob die verbreiteten relationalen Datenbanksysteme auch hier zum Einsatz gelangen können.“ ([Cha06], S. 148) Aus diesem Grund wurden weitere Anforderungskonzepte unterschiedlicher Anbieter mit abweichenden Anzahl und auch Ergänzungen und Vereinfachungen veröffentlicht. Dazu hat auch Codd selbst im Jahr 1995 eine Restrukturierung und Erweiterung um weitere sechs Regeln vorgenommen. (Vgl. [Far11], S. 24)

2.1.2 FASMI

Ein häufig genutztes vereinfachtes System von Evaluierungsregeln zur Beschreibung des OLAP-Konzeptes stellt FASMI dar. FASMI steht für Fast Analysis of Shared Multidimensional Information, ist ebenfalls von Chamoni und Gluchowski ausführlich beschrieben und soll hier im Folgenden kurz dargestellt werden:

1. *Fast:* Es wird für eine Antwortzeit von ein bis zwei, für komplexe Anfragen eine von maximal 20 Sekunden erwartet.

2. *Analysis:* Die Analysefunktionalität soll im spezifischen Anwendungsfall entsprechende Anforderungen erfüllen und intuitiv bedienbar sein.

3. *Shared:* Es muss Mehrnutzerbetrieb möglich sein, dazu ein „Sperrverfahren bei konkurrierenden Schreibvorgängen." ([Cha06], S. 151)

4. *Multidimensional:* Analytische Informationssysteme bestehen essentiell aus einer multidimensionalen konzeptionellen Sicht auf Informationsobjekte, die Anwender müssen freien Zugriff auf die Datenbestände und „multiple Berichtshierarchien über die Dimensionen legen" können. ([Cha06], S. 151)

5. *Information:* Kritischer Faktor bei der Beurteilung von OLAP-Werkzeugen ist die Datenmenge, nicht die Ressourcenbelastung, so dass ein System nur dann einen hohen Nutzen aufweist, wenn bei stabiler Antwortzeit mehr Datenelemente verarbeitet werden können.

2.2 OLTP

Die Abkürzung OLTP steht für Online Transaction Processing. (Vgl. [Bri13], S. 132) OLTP-Systeme werden auch als transaktionale Systeme bezeichnet und haben beinahe ausschließlich Gegenwartsbezug (Vgl. [Mül13], S. 3). Zwar sind in diesen Systemen auch Auswertungen historischer Daten möglich, jedoch werden hierfür OLAP-Systeme bevorzugt, da dort Analysen unabhängig vom operativen Betrieb stattfinden und diesen daher nicht beeinträchtigen.

Unter OLTP wird in operativen Anwendungssystemen die Verarbeitung vorgenommen. Über die Zusammenfassung einzelner Aktionen, die Datenobjekte manipulieren, werden Datenbanktransaktionen durchgeführt. Es steht die Konsistenzerhaltung und Koordination konkurrierender Zugriffe im Vordergrund. (Vgl. [Cha06], S. 175) Mehrere parallel ablaufende Dialoganwendungen, die Datenmanipulation durch Transaktionen vornehmen, kenn-

zeichnen OLTP-Systeme, dazu werden Hintergrundprozesse unterstützt. Da die Hauptanforderung an operative Systeme die Verfügbarkeit in Echtzeit ist, spricht man hierbei auch von Transaktionsverarbeitung. Weiterhin sind operative Systeme durch viele Nutzer gekennzeichnet, die viele kurze Transaktionen durchführen und dabei viele Änderungen im Datenbestand vornehmen. (Vgl. [SV08], S. 41) Die Datenspeicherung erfolgt zeilenbasiert (Vgl. [Mül13], S. 26). Klassische Anwendungsfälle liegen im operationalen Tagesgeschäft und in Geschäftsprozessen in Unternehmen. (Vgl. SchmidtV, S. 41). Für gewöhnlich wird für OLTP-Systeme eine Client-/Server-Architektur genutzt, bei der leistungsfähige Server Dienstleistungen anbieten, die bei Bedarf von Clients angefordert werden. Die Kommunikation basiert auf von Clients generierten und vom Server bearbeiteten Transaktionen. (Vgl. [SV08], S.44f)

Als besondere Stärke von OLTP benennt Schmidt-Volkmar die transaktionsorientierte Bearbeitung weniger Tabellen. Durch das OLTP-System unterstützte Aufgaben sind stark strukturiert und nur gering veränderlich. Schwerpunkt des OLTP-Systems liegt auf einer „Transaktionssicherheit bei parallelen Anfragen, Minimierung der Antwortzeit von Anfragen sowie einem möglichst hohen Datendurchsatz“ ([SV08], S. 46) pro Zeiteinheit. Da die Inhalte der OLTP-Datenbanken durch den transaktionalen Einsatz in ständigem Wandel stehen und sehr viele Detaildatensätze umfassen, sind relationale Datenbank-Managementsysteme hierfür besonders geeignet. Um dort Datenredundanzen zu vermeiden, werden die entsprechenden Datenbank-Tabellen stark normalisiert und erzeugen damit komplexe Strukturen mit vielen kleinen aufeinander bezogenen Tabellen. Dies erschwert die Nutzung dieser Systeme als Entscheidungssysteme (Vgl. [Lan01], S. 796):

Da Daten in OLTP-Systemen sehr komplex und unzusammenhängend aufgebaut sind, können sie nicht unmittelbar als Datengrundlage für analytische Auswertungen genutzt werden. (Vgl. [SV08], S. 46ff) Eine Datenaggregation wird nur mangelhaft unterstützt, wohingegen diese in OLAP-Systemen als wichtig eingestuft wird. Einen weiteren Nachteil stellt eine „mangelnde Unterstützung analytischer Zugriffe auf transaktionsübergreifende Informationen“ (Vgl. [SV08], S. 49) dar. Typische multidimensionale Abfragen, wie sie in einem Data Warehouse für umfangreiche Analysen getätigt werden, sind aufgrund der Datenspeicherung in vielen kleinen Tabellen mit kleiner Anzahl von Attributen und der damit verbundenen umfangreichen Verbundoperationen eine übermäßig starke Belastung der OLTP-Systeme und gefährden damit die geforderte ständige Verfügbarkeit. Zudem wird nur eine aktuelle und keine historische Analyse oder Trendanalyse von diesen Systemen ermöglicht. (Vgl. [SV08], S. 49) Betriebswirtschaftlich werden diese Systeme weniger im Management-Bereich und eher im ausführenden Bereich genutzt, in welchem sie

beispielsweise Geschäftsprozesse durch ihre Transaktionen unterstützen. OLTP-Systeme unterliegen dem im Folgenden kurz dargestellten AKID-Prinzip (Vgl. [Far11], S. 51):

Das AKID-Prinzip soll für transaktionsorientierte Informationssysteme Transaktionssicherheit gewährleisten ([Lei14]) und stellt damit Anforderungen an transaktionale Systeme. Es beschreibt gewünschte Eigenschaften der Datenverarbeitung in einer Datenbank und besteht aus vier Teilen:

1. *Atomarität* von Transaktionen: Wenn ein Teil einer Transaktion fehlschlägt, so hat die gesamte Transaktion fehl zu schlagen, der Datenbankstatus muss unverändert bleiben.

2. *Konsistenz:* Datenbanken müssen immer konsistent sein, es werden nur gültige Daten in ihnen abgelegt. Würde eine Transaktion durchgeführt werden, die diese Eigenschaft verletzt, wird die gesamte Transaktion negiert und die Datenbank in einen regelkonformen Status zurückgesetzt. Wird eine Transaktion erfolgreich ausgeführt, wird „die Datenbank von einem konsistenten Status in einen anderen Status versetzt, der ebenfalls mit den Regeln konsistent ist.“ ([Ber13], S. 26)

3. *Isoliertheit:* Um Störungen zwischen Transaktionen zu verhindern, müssen diese isoliert werden.

4. *Dauerhaftigkeit:* Transaktionen müssen dauerhaft sein, wenn sie an die Datenbank übergeben wurde, bleibt sie auch dort. (Vgl. [Ber13], S. 24ff)

2.3 OLAP und OLTP im Vergleich

Die beiden Datenbankparadigmen OLAP und OLTP können – wie bereits beschrieben – aufgrund ihrer Beschaffenheit nur getrennt voneinander agieren. Während OLAP den Fokus auf die Analyse von Daten legt, ist OLTP für operationalen Nutzen vorhergesehen. Weitere Merkmale werden in der folgenden Tabelle 1 nach Müller und Lenz gegenübergestellt und sollen die Unterschiede übersichtlich in Bezug auf die Nutzung in einem Unternehmen verdeutlichen.

Es kann aus der Gegenüberstellung entnommen werden, dass sich die beiden Systeme in sehr vielen Bereichen unterscheiden. OLTP-Systeme sind operativ, werden eher von Sachbearbeitern genutzt, in ihnen findet aufgrund der geforderten Schnelligkeit keine Datenverdichtung statt. Dazu basieren sie auf aktuellen Geschäftsdaten und werden laufend aktualisiert. Daher besteht auch die Anforderung an eine kurze Antwortzeit, da sie häufig im laufenden Betrieb eingesetzt werden. OLAP-Systeme sind im Gegensatz dazu

Merkmal	**OLTP**	**OLAP**
Anwendungsbereich	Operative Systeme	Analytische Systeme
Nutzer	Sachbearbeiter	Entscheidungs- und Führungskräfte
Datenstruktur	Zweidimensional, nicht verdichtet	Multidimensional, subjektbezogen
Dateninhalt	Detaillierte, nicht verdichtete Einzeldaten	Verdichtete und abgeleitete Daten
Datenaktualität	Aktuelle Geschäftsdaten	Historische Verlaufsdaten
Datenaktualisierung	Durch laufende Geschäftsvorfälle	Periodische Datenaktualisierungen
Zugriffsform	Lesen, schreiben, löschen	Lesen, anfügen, verdichten
Zugriffsmuster	Vorhersehbar, repititiv	Ad hoc, heuristisch
Zugriffshäufigkeit	Hoch	Mittel bis niedrig
Antwortzeit	Kurz (Sekundenbruchteil)	Mittel bis lang (Sekunden bis Minuten
Transaktionsart und Dauer	Kurze Lese- und Schreiboperationen	Lange Lesetransaktionen

Tabelle 1: OLTP und OLAP im Vergleich [Mül13], S. 16

analytisch, werden von Entscheidungs- und Führungskräften genutzt und liegen in stark verdichteter Form subjektbezogen vor. Sie nutzen historische Verlaufsdaten und werden nur periodisch aktualisiert. Da diese Systeme nicht regelmäßig genutzt werden und dazu auf großen Datenbeständen agieren, wird eine mittlere bis lange Antwortzeit in Kauf genommen. Dazu unterscheiden sich die beiden Systeme durch die Zugriffsform: OLTP-Systeme erlauben Lesen, Schreiben und Löschen wohingegen OLAP-Systeme nur Lesen, Anfügen und Verdichten gestatten. Zusätzlich sei hier erwähnt, dass OLTP-Systeme aufgrund ihrer gegebenen Anforderungen und daraus resultierenden Grundstrukturen einen deutlich höheren Normalisierungsgrad aufweisen als OLAP-Systeme, die keine häufigen und schnellen Inserts und Updates durchführen müssen.

Aus diesem Vergleich lassen sich die folgenden Schlüsse ziehen: Zwar können OLAP-Systeme aus OLTP-Systemen erwachsen, so dass eine strikte Trennung nicht immer möglich und Hybridsysteme daher sinnvoll sein können (Vgl. [Bay12]), wenn beispielsweise Analysen auf aktuellen und nicht historischen Daten erzeugt werden sollen. Jedoch sollten OLTP- und OLAP-Anwendungen dabei möglichst nicht auf der gleichen physischen Datenbank ausgeführt werden. „Der Entwurf von OLTP-Datenbanken ist auf die Änderung kleiner Datenmengen durch Transaktionen ausgerichtet, wobei die Daten oft auf viele Datenbanken verteilt sind.“ ([Rög10], S. 29) OLAP-Analysen erfordern einen anderen Entwurf,

damit die Daten in konsolidierter und integrierter Form dargestellt werden können. Aufgrund der häufig komplexen OLAP-Anfragen können diese im laufenden Betrieb OLTP-Anwendungen stark in ihrer Leistung vermindern. (Vgl. [Rög10], S. 29) Echtzeitanalysen sind daher weder mit OLAP noch mit OLTP möglich, so dass keine aktuellen Analysen erstellt werden können und immer eine Zeitverzögerung besteht. Dazu können Änderungen der zu analysierenden Daten zu Testzwecken kaum durchgeführt werden. In OLTP werden diese Daten sofort im System verarbeitet und können damit unerwünschte Wirkungen erzeugen, über OLAP stehen die Daten erst nach einem aufwendigen Upload-Prozess zur Verfügung. Daher können Prognosen anhand von Testdaten nur sehr schwer in kurzer Zeit erstellt werden. Unter anderem aus diesen Beweggründen entstand der Gedanke der In-Memory-Datenverarbeitung.

3 Grundlagen für In-Memory

Um dem bestehenden Verlangen von Nutzern, Entscheidungen in Echtzeit treffen zu können, nachzukommen, mussten schnellere Möglichkeiten gefunden werden, aktuelle Informationen und Erkenntnisse zur weiteren Nutzung bereitzustellen. Unter anderem hierfür wurde die In-Memory-Technologie entwickelt, die Daten von einer Festplatte in den Hauptspeicher eines Systems verlagert und sie damit sehr schnell zur Verfügung stellen kann. (Vgl. [Ber13], S. 24)

Berg und Silvia beschreiben die Grundlage von In-Memory in der Form, dass „riesige Datenmengen im Hauptspeicher verarbeitet und Analyse- und Transaktionsergebnisse direkt bereitgestellt werden.“ ([Ber13], S. 24) SAP als ein Anbieter dieser Technologie lässt zu verarbeitende Daten in Echtzeit oder Beinahe-Echtzeit generieren, indem durch die Nutzung des Hauptspeichers als zentrales Speichermedium Datenzugriffe beschleunigt und Datenbewegungen minimiert werden.

Der aktuell schnellste Speichertyp ist Random Access Memory (RAM): Hier kann bis zu 100.000-mal schneller auf Daten zugegriffen werden, als auf Daten von einer Festplatte. (Vgl. [Ber13], S. 25) Bei der In-Memory-Technologie entfällt die Notwendigkeit der Bildung voraggregierter Datenwürfel: Bei Bedarf erfolgen die Aggregierungen, die der Nutzer des Systems gerade für seine Analyse benötigt, so dass Analysen mit aktuellen Systemdaten durchgeführt werden können. ([AG14])

Beim Start des ausführenden Systems werden alle benötigten Daten in den Hauptspeicher geladen, was jedoch auch bedeutet, dass die Größe eines Würfels generell in RAM begrenzter ist als bei einer externen Speicherform. Um das Datenvolumen geringer zu hal-

ten, als bei vorberechneten Verfahren, werden die abgeleiteten Daten dynamisch berechnet, dies verkürzt die Laufzeit. (Vgl. [SV08], S. 23)

Da In-Memory-Systeme sowohl analytische als auch transaktionsorientierte Vorgänge durchführen können sollen, müssen die Anforderungen aus dem unter Abschnitt 2.2 erläuterten AKID-Prinzip erfüllt sein. In-Memory-Technologie kann hier jedoch eine Dauerhaftigkeit nach dem Prinzip nicht ohne weiteres leisten: Fällt beispielsweise der Strom aus, gehen die im Hauptspeicher gelagerten flüchtigen Daten verloren. Somit kann die Datensicherheit eingeschränkt sein, so dass unterschiedliche Abhilfen geschaffen worden sind. Ein komplizierter Replikationsmechanismus sorgt in regelmäßigen Abständen dafür, „dass die Daten aus dem Speicher auf eine ausfallsichere persistente Datenhaltung geschrieben werden.“ ([SV08], S. 170)

SAP HANA bietet als Abhilfe für diese Fälle beispielsweise die Lösung eines Disaster Recovery an, also die Fähigkeit des Systems, sich nach einem Strom- oder Systemausfall selbst wiederherzustellen.

Hierfür ist das Medium, das von der Datenbank zum Speichern verwendet wird, in Seiten unterteilt. Ändert eine Transaktion Daten, so werden die betroffenen Seiten markiert und regelmäßig in nicht-flüchtigen Speicher geschrieben. Dazu wird ein Datenbankprotokoll in nicht-flüchtigem Speicher geführt, in dem sämtliche durch Transaktionen erzeugte Änderungen festgehalten werden. Damit kann die Dauerhaftigkeit von Transaktionen sichergestellt werden.

SAP HANA verfolgt diesen Ansatz, indem geänderte Daten an sogenannten Savepoints gespeichert werden, die im 5-Minuten-Takt asynchron in persistenten Speichern abgelegt werden. Es wird zusätzlich synchron ein Protokoll geschrieben. Fällt das System aus, so werden die Datenbankseiten mithilfe der Savepoints wiederhergestellt, zusätzlich werden die Datenbankprotokolle genutzt, um Änderungen wiederherzustellen, die nicht von den Savepoints erfasst wurden. Damit kann die Datenbank im Hauptspeicher auf den Status gesetzt werden, den sie vor einem eventuellen Systemausfall hatte. (Vgl. [Ber13], S. 25f)

3.1 Zeilen- und spaltenbasierte Organisation

Es lassen sich zwei unterschiedliche Ansätze zur Datenspeicherung und -verarbeitung in Datenbanksystemen unterscheiden: Zeilen- und spaltenbasierte Organisation. Der Zweck der Datenbanksysteme ist hierbei maßgeblich für die Wahl des Ansatzes, da von den auf der Datenbank auszuführenden Operationen anhängt, welche Organisation sich besser eignet. Für eine effiziente Transaktionsverarbeitung mit Insert-/ Update- und Delete-Operationen,

wie sie in OLTP häufig genutzt werden, wird die sogenannte zeilenbasierte Organisation genutzt, die in der folgenden Abbildung 2 kurz dargestellt ist:

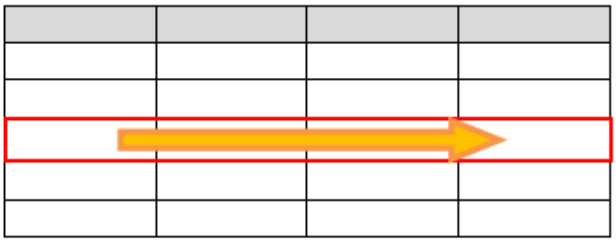


Abbildung 2: Zeilenbasierte Datenverarbeitung

Es wird dort bei den entsprechenden auszuführenden Operationen auf viele Spalten zugegriffen, da der Inhalt jeder Spalte einer Reihe gelesen werden muss, was bei sehr großen und spaltenreichen Tabellen zu sehr langen Bearbeitungszeiten führen kann.

Die spaltenbasierte Organisation dagegen ist eher für Abfragen und Aggregationen geeignet, da sie – insbesondere bei großen Tabellen – im Fall der Datenabfrage zu kürzeren Bearbeitungszeiten führt. Grund hierfür ist, dass die Tabellen spaltenweise, wie in Abbildung 3 kurz dargestellt, gelesen und damit nicht alle Daten in jeder Zeile abgerufen werden:

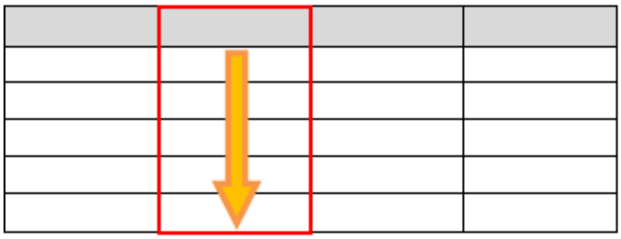


Abbildung 3: Spaltenbasierte Datenverarbeitung

Wird beispielsweise ein Datensatz gesucht, so werden dabei entweder alle Datensätze zeilenweise gelesen oder über Spalten die entsprechenden Muster gesucht. Abhängig von der Größe des Datenbestandes und Länge der gespeicherten Zeilen kann die spaltenbasierte Organisation deutlich Zeit einsparen, da dort bis zum Finden eines Datensatzes nicht erst alle Zeilen gelesen werden müssen, sondern nur bestimmte Datensätze zu einer Spalte.

3.2 Vor- und Nachteile zeilen- und spaltenbasierter Organisation

Einige Vor- und Nachteile zeilen- und spaltenbasierter Speicherung sind in der folgenden Tabelle 2 gegenübergestellt:

	Zeilenbasierte Speicherung	**Spaltenbasierte Speicherung**
Vorteile	Daten werden zusammenhängend gespeichert und können einfach eingefügt bzw. aktualisiert werden.	Nur die relevanten Spalten werden beim Auswahlprozess gelesen, und jede Spalte kann als Index oder Schlüssel zum Datenabruf dienen.
Nachteile	Bei der Auswahl müssen alle Daten gelesen werden.	Datenaktualisierungen sind bei der spaltenbasierten Speicherung nicht so effizient wie bei der zeilenbasierten Speicherung.

Tabelle 2: Vor- und Nachteile der zeilen- bzw. spaltenbasierten Speicherung [Ber13], S. 41

Da die Daten in einem Business Warehouse trotz Faktentabelle weiterhin auf mehrere Tabellen verteilt sind, wirkt sich der Nachteil, dass bei zeilenbasierter Speicherung alle Daten gelesen werden müssen, stark auf die Abfragegeschwindigkeiten aus. Effizienznachteile bei der Datenaktualisierung bei einer spaltenbasierten Speicherung entstehen dadurch, dass das System bei einer Aktualisierung erst die richtige Spalte und dann die richtige Zeile suchen muss.

Zusätzlich gibt es spaltenbasierte Indizes. Diese basieren auf der Grundidee, dass Datenwiederholungen und -muster in den Tabellen ergeben, die über Indexerstellung und Komprimierungsprozess reduziert werden können:

Werden Daten in Spaltenperspektive betrachtet statt in Zeilenperspektive, so können in Datensätzen Redundanzen gefunden werden, beispielsweise bei einem Kreditkartenunternehmen mit den Kreditartentypen SILVER, GOLD und PLATINUM, wie von den Autoren Berg und Silvia als Beispiel genutzt: Die Kundendaten weisen wenig bis keine Redundanzen auf, jedoch lassen sie sich entsprechend der Kreditkartentypen reduzieren. Dies bringt für den Datenzugriff Performancevorteile. Demnach haben spaltenbasierte Speicher „Vorteile einer hohen Komprimierungsrate, besserer Scanoperationen (einfacherer Suche) und einer In-Cache-Client-Verarbeitung der Aggregation (einfacherer Gruppierung und Aggregation)“. (Vgl. [Ber13], S. 41ff)

3.3 SAP HANA als ein Beispiel für In-Memory-Technik

SAP bietet über die SAP HANA ein Beispiel für die In-Memory-Technologie. Sie kann sowohl zeilen- als auch spaltenbasiert arbeiten: Über einen zeilenbasierten vorgeschalteten Puffer werden transaktionale Daten aufgenommen, um sie im Hintergrund in Spaltenstruktur überführen zu können. Damit ist sowohl ein schnelles Schreiben in die Datenbank als auch ein schnelles Auslesen von Daten möglich ([Gmb13b]). Die Abbildung 4 stellt ein Modell des SAP In-Memory Computing dar.

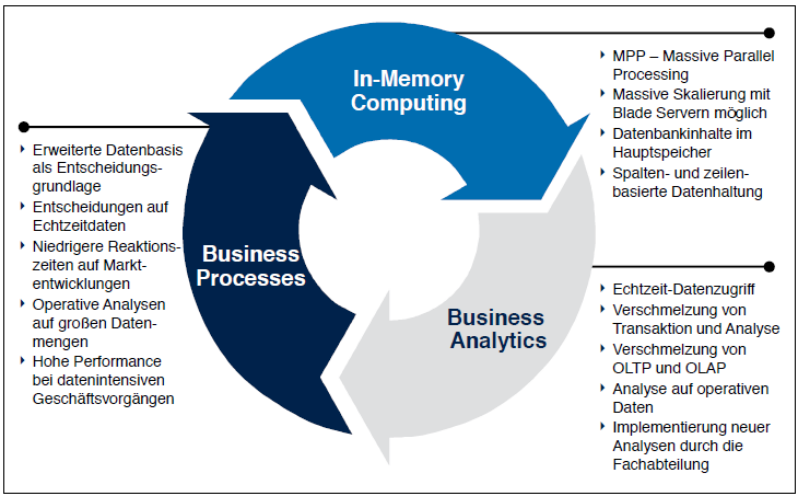


Abbildung 4: SAP In-Memory Computing [Sim12], S. 2

Wichtigste technologische Grundlage bilden hier die spaltenbasierte Datenspeicherung und eine teilweise darauf beruhende Möglichkeit zur massiven Parallelisierung (MPP, Massive Parallel Processing). Business Processes erzeugen Daten, die über das In-Memory-Computing verarbeitet werden. Sie bieten damit eine Datenbasis zur Entscheidungsfindung. Durch die Möglichkeit, über In-Memory Echtzeitanalysen durchzuführen, können diese Entscheidungen damit auch auf Echtzeitdaten basieren und ermöglichen beispielsweise eine niedrige Reaktionszeit auf Marktentwicklungen im entsprechenden Unternehmensumfeld. Durch die Möglichkeit der zeilenbasierten Verarbeitung in der SAP HANA wird eine hohe Performance bei datenintensiven Geschäftsvorgängen geboten. Über Massive Parallel Processing schafft die Datenbank niedrige Verarbeitungszeiten. Dazu werden die Datenbankinhalte in die Hauptspeicher geladen, die Haltung kann sowohl zeilen- als auch spaltenbasiert erfolgen. Daraus lässt sich ein Echtzeit-Datenzugriff in den Business Analytics durchführen, OLAP und OLTP sich miteinander verbinden. Auf operativen Daten können Analysen durchgeführt werden, neue Analysen können implementiert werden, die dann wieder auf die Business Processes einwirken. (Vgl. [Sim12], S. 2ff)

Durch diese Möglichkeiten bietet die SAP HANA, wie auch im weiteren Verlauf beschrieben, eine ideale Grundlage für das Projekt RAPID: Echtzeitanalysen und die „Verschmelzung“ von OLAP und OLTP ermöglichen umfangreiche Analysen im Mobilitätsbereich durch eine Möglichkeit der Kombination aktueller Daten mit historischen Datenbeständen.

4 Die Bedeutung von In-Memory für die Datenbankparadigmen OLAP und OLTP

Da sich Datenbanktabellen typischer Data Warehouses „entweder nur hinsichtlich eines performanten Lesezugriffs oder schneller Schreiboperationen hin konfigurieren“ ([SV08], S. 170) lassen, dazu lange Leseoperationen zu schlechten Antwortzeiten anderer Prozesse führen, was bei transaktionalen Systemen wie OLTP nicht akzeptabel ist, wurden die Systemtypen analytischer und transaktionaler Systeme bislang getrennt. Für Szenarien, in denen aktuelle Daten erforderlich sind, ist dieser Zustand jedoch nicht zweckmäßig.

Bei der In-Memory-Verarbeitung sind sowohl Lese- als auch Schreibzugriffe – anders als bei OLAP auch „gleichzeitig“ – erlaubt, so dass eine schnelle Verarbeitung auch ohne ein Behindern anderer laufender Prozesse vorgenommen werden kann und damit der operative Betrieb eines In-Memory-Systems nicht gestört wird. Dadurch wird die bislang existierende und durch die Systeme OLAP und OLTP fixierte Trennung analytischer und operativer Systeme überflüssig. Durch die Tatsache, dass sämtliche Datenänderungen in In-Memory-Systemen protokolliert werden, stehen dem Benutzer einer In-Memory-Datenbank alle Änderungen zur Verfügung und es entsteht kein Informationsverlust. Operative Daten können mithilfe logischer Sichten „ohne umfangreiche Transformationsregeln in relevante Managementinformationen überführt werden. Änderungen finden lediglich an virtuellen, logischen Objekten statt.“ ([SV08], S. 169f) Damit entfallen beispielsweise Anpassungen der „zahlreichen, bisher notwendigen“ Data Warehouse (DW)-Objekte „entlang der Fortschreibung.“ ([SV08], S. 170)

In-Memory-Systeme bedeuten eine Abkehr der traditionellen Konzepte der Separation der Systemtypen und der Integrationsfunktion, auch ohne Datenmodelloptimierung können hier durch schnellere Datenzugriffe deutlich niedrigere Antwortzeiten erzeugt werden. In-Memory kann auch für analytische und transaktionale Verarbeitung genutzt werden, durch die Nutzung des Hauptspeichers fällt die durch die für ein Data Warehouse notwendige Datenverarbeitung entstehende Latenz aus, da Datenänderungen sofort für Analysen zur Verfügung stehen. Somit ist der Informationsbereitstellungsprozess deutlich beschleunigt. (Vgl. [SV08], S. 199ff)

In-Memory ermöglicht demnach eine zeitnahe Verfügbarkeit von Datenänderungen, eine einfache Anpassung von Datenmodellen an Benutzeranforderungen und eine „zufriedenstellende Antwortzeit des Systems“, ([SV08], S. 202f) die Datenbankparadigmen OLAP und OLTP scheinen damit überflüssig.

In der Abbildung 5 ist eine vereinfachte Darstellung einer In-Memory-Architektur am Beispiel der SAP HANA dargestellt.

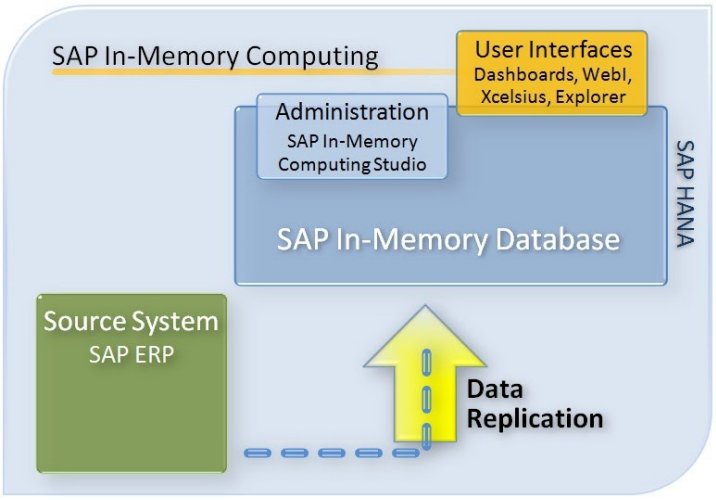


Abbildung 5: Beispiel für In-Memory-Architektur [Bus12]

Hier wird gezeigt, dass die Daten, direkt aus einem Quellsystem, beispielsweise aus einem ERP-System oder aber auch aus jeder anderen Datenbank in die In-Memory-Datenbank geladen werden können. Dieses erfolgt über Datenreplikationen, die beispielsweise auf einem ETL (Extraktion, Transformation, Laden)-Prozess basieren können. Die Administration der Datenbank erfolgt direkt, ebenso eine Schnittstelle, über die die Nutzer auf das System zugreifen und Auswertungen erstellen können. (Vgl. [Bus12])

Abbildung 6 dagegen zeigt eine vereinfachte Darstellung der OLAP-Architektur. Herauszustellen ist hier, dass die Verarbeitung von Daten aus den Datensystemen, transaktionale Systeme erst über einen ETL-Prozess in ein Data Warehouse geladen werden müssen, bevor sie über OLAP-Systeme analysiert werden können.

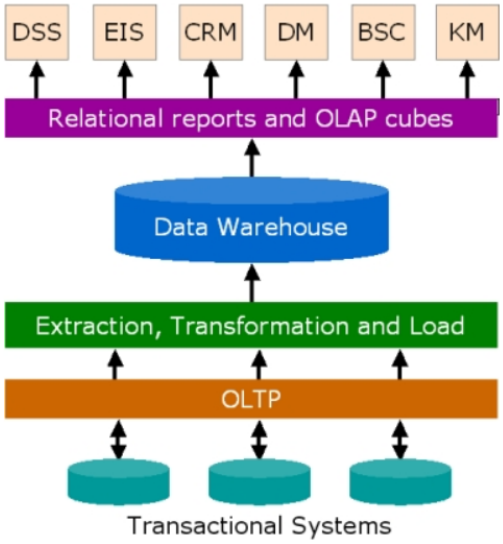


Abbildung 6: OLAP-Architektur [SeroD]

Es ist aus einem Vergleich der beiden Abbildungen ersichtlich, dass Aufbau und Handhabung bei der In-Memory-Architektur „einfacher“ ist und eine Datenaufbereitung entfallen kann.

Moderne OLAP-Datenbanken sind offener als In-Memory-Systeme, sie können auch von außen über MDX-Anfragen[3] integriert werden. OLAP benötigt eine zweistufige Beladung, die vorwiegend von Usern mit Fachkenntnissen vorgenommen werden sollten, wohingegen bei In-Memory eine einfache Beladung genügt (auch hier sind jedoch Fachkenntnisse notwendig). Mit einer entsprechenden Aufbereitung sind beide Architekturen auch von Nicht-Fachleuten nutzbar. Beide Architekturen weisen Grenzen bei sehr großen Datenmengen durch die Serververfügbarkeit auf. ([Gmb13a]) Spezialisten sehen In-Memory jedoch nicht als Alleinlösung für Probleme an, da, trotz sinkender Memory-Preise Data Warehouses im Petabyte-Bereich aufgrund ihrer Größe sinnvoller über relationale Datenbanksysteme zu verwalten und analysieren sind. ([Bay12])

Trotz der Vorteile von In-Memory mit Echtzeitverfügbarkeit werden auch Analysen mit historischen Daten seitens der Unternehmensentscheider verlangt, so dass OLAP-Systeme zudem weiterhin notwendig sein werden, da strategische Entscheidungen auf Unternehmensbasis nicht nur auf aktuellen Unternehmensdatenauswertungen getroffen werden können.

Einige Anbieter kombinieren daher OLAP und In-Memory miteinander: Es wird dabei

[3]MDX steht für Multidimensional Expressions und ist eine Datenbanksprache für OLAP.

die Performance einer In-Memory-Datenbank mit einem Self-Service-BI (Business Intelligence) kombiniert zu In-Memory-OLAP-Datenbanken. Beispiele hierfür sind QlikView von QlikTech, Palo von Jedox oder auch PowerPivot von Microsoft. ([Gmb14])

Zu beachten ist jedoch, dass eine OLAP-Architektur verbunden mit einer In-Memory-Datenbank eine Problematik der Verlangsamung von Prozessen mit sich bringen kann: Die Verarbeitung von Daten über eine externe Datenbank kann dazu führen, dass Auswertungen nicht mehr in Echtzeit durchgeführt werden können, da die OLAP-Anwendungen von der Ladegeschwindigkeit auf die In-Memory-Datenbank abhängig sind, wobei diese von der Geschwindigkeit der auswärtigen Datenbank abhängig ist. Diese so genannten Bottlenecks gilt es in Anbetracht des bestehenden In-Memory-Geschwindigkeitsvorteils zu vermeiden.

5 In-Memory mit SAP HANA und das Projekt RAPID

Das Projekt RAPID hat eine intelligente Plattform für Mobilitätsdaten zum Ziel. Die Nutzer der Plattform sollen befähigt werden, eine Region wirtschaftlich und ökologisch effizient mit Fokus auf den Verkehr zu erschließen und zu bewirtschaften. Wenn möglich, sollen die jeweiligen Daten in Echtzeit zur Verfügung stehen. Für das Projekt wird daher die In-Memory-Datenbank SAP HANA genutzt.

SAP HANA als ein flexibles und von Datenquellen unabhängiges Werkzeug, bei dem die Datenherkunft nebensächlich ist, ist aufgrund der in Abschnitt 3 beschriebenen In-Memory-Beschaffenheit in der Lage, große Mengen an Daten in Echtzeit speichern und analysieren zu können, ohne dass komplexe physisch Datenmodelle aggregiert oder erstellt werden müssen. Sie verknüpft als eine In-Memory-Datenbanklösung Hard- und Software, „die zeilen-, spalten- und objektbasierte Datenbanktechnologien miteinander kombiniert“ und nutzt damit die Funktionen paralleler Verarbeitung. (Vgl. [Ber13], S. 34) SAP HANA verfolgt das Ziel, die Datenbanken von OLTP- und OLAP-Anwendungen auf einem einzigen In-Memory-Server verschmelzen zu lassen. Die beiden Systeme sollen dann jeweils eine logische Sicht bilden und eine gemeinsame In-Memory-Datenbank nutzen, auf der Transaktions- und Analysedaten zusammenfließen. ([Sch12])

Analysen – auch auf transaktionalen Daten – sollen dabei in der SAP HANA direkt ablaufen, welches über eine wie in Abschnitt 3.1 beschriebene spaltenorientierte Struktur ermöglicht wird. Zudem können Daten sehr schnell in die Datenbank geschrieben werden. Da jedoch hierfür wie im benannten Abschnitt beschrieben, eher eine zeilenorientierte Verarbeitung stattfinden muss, um eine angemessene Geschwindigkeit zu erhalten, hat SAP nach Angaben des Engineering-Dienstleisters Ferchau „vor die eigentliche spaltenorien-

tierte Datenbank einen kleinen zeilenbasierten Puffer vorgeschaltet, in dem die Daten aus dem transaktionalen System zunächst landen," und dann in die Spaltenstruktur überführt werden. ([Gmb13b])

Somit ist die Nutzung der SAP HANA für das Projekt RAPID sinnvoll. Es können auf aktuellen und historischen Daten Analysen erstellt werden, womit die Möglichkeiten, die wie oben beschrieben über OLAP und OLTP nur eingeschränkt zur Verfügung stehen, gemeinsam auf einer Datenbank mit schneller In-Memory-Technik genutzt werden können. Mögliche Anwendungsfälle für den Mobilitätssektor, die im Rahmen des Projekts aufgearbeitet und über die In-Memory-Technik verarbeitet werden können, wären beispielsweise:

- Eine Analyse des Verkehrsaufkommens von Großveranstaltungen in der Region auf bestimmten Strecken auf der Basis historischer Verkehrsdaten (hier ist auch eine Auswertung über OLAP möglich)
- Die Auswirkungen von Kfz-Zulassungen auf den Verkehrsfluss in Korrelation zum Ausbau des Straßennetzes und die damit verbundene Auslastung des Straßennetzes einer Region auf der Basis historischer und aktueller Daten
- Erfassung von Pendlerdaten der Region zur Planung einer besseren Abstimmung, Auslastung und Taktung von öffentlichen Verkehrsmitteln sowie Schaffung zielgerichteter Park-and-Ride-Möglichkeiten auf der Basis aktueller und historischer Daten
- Reduzierung des innerstädtischen CO2-Aufkommens durch Analyse von Verkehrsstoßzeiten besonders belasteter Strecken und damit Schaffen der Möglichkeiten von sogenannten „grünen Wellen" für steten Verkehrsfluss beispielsweise im Berufsverkehr auf der Basis aktueller und historischer Daten.
- Auf Basis von aktuellen Wetterdaten in Verbindung mit aktuellen Verkehrsdaten Steuern der elektronischen Straßenverkehrssignalanlagen, dabei beispielsweise Ampeln und Geschwindigkeitsanzeigen optimieren.
- Berechnen von Verkehrsaufkommen anhand historischer Wetterdaten zur frühzeitigen Schaffung von verkehrsberuhigenden Maßnahmen und Abstimmung im Personennahverkehr.
- Erfassen der aktuellen Belastung von Fernverkehrsstraßen einer Region mittels des Toll-Collect-Systems zur Vermeidung von Staus, in Verbindung mit aktuellen Wetterdaten, um Verkehr gegebenenfalls frühzeitig umleiten zu können.

- Erfassen des aktuellen Verkehrsaufkommens auf Hauptstraßen um gegebenenfalls Busse umzuleiten, damit die Busfahrpläne zuverlässiger eingehalten und damit Busse attraktiver für die Bevölkerung einer Region werden können.
- Messen des aktuellen Verkehrsaufkommens einer Region um elektronische Busabfahrtstafeln mit genaueren Daten zu versehen und gegebenenfalls Ersatzbusse zu schicken

An den Anwendungsfällen kann man erkennen, dass es sinnvoll ist, die In-Memory-Technik zu verwenden, da die Anwendungsfälle teilweise auf historischen und teilweise auf aktuellen Werten basieren. Nur In-Memory erlaubt eine Echtzeit-Darstellung und dazu die Möglichkeit, sowohl OLAP- als auch OLTP-Anwendungsfälle durchzuführen.

Für das Projekt RAPID bedeutet dies, dass in der In-Memory-Datenbank sowohl historische als auch aktuelle Daten verwaltet werden müssen, um kurzfristig Echtzeit-Datenauswertungen zur Verfügung stellen und gegebenenfalls dann mit historischen Daten in Verbindung bringen zu können. Hierfür ist es besonders wichtig, die Schnittstellen zur SAP HANA so zu gestalten, dass gleichzeitig Echtzeit- als auch historische Daten eingespielt werden können, diese so zu formatieren, dass sie sofort verarbeitet und anschließend umgehend analysiert werden können.

Die Herausforderung wird dabei möglicherweise darin bestehen, die korrekten Formatierungen zu gestalten und die gleichzeitigen Vorgänge adäquat zu verwalten. Dazu besteht eine zentrale Herausforderung darin, die für Analysen historischer Daten benötigen Datensätze bei Analysebeginn bereits in der SAP HANA zu haben, so dass diese zuvor mit entsprechenden Verfahren ausgesiebt werden sollten. Würden alle Daten in die SAP HANA geladen werden, so könnte dies in Anbetracht eventuell durchzuführender Echtzeitdatenaufnahme und -analyse zu Platzproblemen führen.

Die beiden im Folgenden kurz dargestellten Beispiele stellen bereits aktive Systeme im Mobilitätssektor dar und zeigen eine kurze Idee der Möglichkeiten dieser Systeme aber auch verbundener noch unberücksichtigter Probleme auf: Das erste Beispiel und der daraus entstandene oben genannte Anwendungsfall, wurde sehr wahrscheinlich über ein OLAP-System realisiert: Über das System Toll Collect konnte in der Vergangenheit durch Kennzeichenüberprüfung eine Person festgenommen werden, die 2008 auf Autobahnen auf Fahrzeuge geschossen haben soll. Für die Ermittlungen hatte das Bundeskriminalamt an den Mautstationen Kennzeichenlesegeräte angebracht und die Daten analysiert. Ein genereller Einsatz solcher Methoden wird politisch jedoch aus datenschutzrechtlichen Gründen

strikt abgelehnt, was auch am zweiten Beispiel, einem nicht realisierten Projekt, zu erkennen ist:

Zur Verbesserung der Steuerung des Transportaufkommens zum Hamburger Hafen sollten LKW-Positionen im Großraum Hamburg von Toll Collect, Eurogate und der Hamburg Port Authority ausgewertet werden. Das Projekt scheiterte 2009 am Datenschutz. ([Cla13]) Somit lässt sich insbesondere für die künftige Arbeit im Projekt RAPID schließen, dass rechtliche Gegebenheiten bei der Datenbeschaffung und -analyse nicht außer Acht gelassen werden dürfen.

Eine allgemeine Sammlung und Auswertung von veröffentlichten Zahlen der statistischen Bundesämter, Unternehmen mit Genehmigung zur Datenaufzeichnung und anderer allgemeiner Quellen sollte in diesem Zusammenhang jedoch keine Probleme mit sich bringen. Ob dann auch Echtzeit-Auswertungen möglich sind, wird sich im Verlauf des Projektes herausstellen.

6 Fazit

Die Arbeit hat nach einer ausführlichen Darstellung von OLAP und OLTP inklusive verbundener Regelwerke einen direkten Vergleich der beiden Technologien vorgenommen. Hierbei hat sich gezeigt, dass diese sich nicht ohne weiteres miteinander verbinden lassen. OLAP baut zwar auch auf OLTP-Daten auf, jedoch würde OLAP den operativen Bereich beispielsweise eines Unternehmens stören, wenn es für langwierige Analysen die OLTP-Datenbestände nutzen würde. Dazu weisen sie unterschiedliche Strukturen auf.

Im weiteren Verlauf wurde In-Memory inklusive dort verwendeter Technologien vorgestellt und wiederum mit den beiden oben genannten Datenbankparadigmen verglichen. Insbesondere der Vergleich der OLAP- und In-Memory-Architektur brachte hierbei deutliche Unterschiede hervor.

Die Arbeit hat anschließend die In-Memory-Technologie der SAP HANA für die Verwendung des Projektes RAPID betrachtet und mögliche Anwendungsfälle dargestellt. Da die SAP HANA Datenbank Möglichkeiten für Echtzeitdatenauswertungen stellt, aber auch historische Daten in die Anwendungsfälle mit einbezogen werden sollten, befindet sich an dieser Stelle eine Herausforderung für die Projektgruppe des Projekts RAPID: Da sich mögliche Anwendungsfälle nicht nur auf Echtzeit-, sondern auch auf historische Daten beziehen, darf der „OLAP-Bereich“ nicht außer Acht gelassen werden und für die Analysen benötigte Daten sollten entsprechend in der SAP HANA für Analysen verfügbar sein.

Da Arbeitsspeicher, jedoch auch die zu verarbeitenden Datenmengen, immer größer und auch historische Datenauswertungen immer benötigt werden (beispielsweise will das Management eines Unternehmens Entscheidungen aufgrund von Erfahrungswerten treffen und sich nicht nur auf eine aktuell bestehende Unternehmenslage beziehen), besteht in der Größe der Arbeitsspeicher immer noch eine Grenze für die Möglichkeiten von In-Memory-Technologien. Aus diesen Gründen können diese aktuell die beiden Datenparadigmen OLAP und OLPT noch nicht vollständig ablösen. Hybridsysteme von OLAP und In-Memory können hier die Lösung sein, da In-Memory in der Lage ist, transaktionale Vorgänge auszuführen und in Kombination dazu dann über OLAP-Komponenten die Möglichkeit für eine (historische) Datenauswertung bieten kann. Dabei ist es jedoch wichtig, Bottlenecks zu vermeiden, die zu Verlangsamung der Vorgänge durch zu lange externe Ladezeiten verursacht werden.

Es hat sich demnach gezeigt, dass es wichtig ist, persistente Speicherung von aktuellen und historischen Daten in der SAP HANA zu ermöglichen, wobei durch die Platzbeschränkung eine Ausweichmöglichkeit oder ein „Verfallsdatum“ der Daten in Betracht gezogen werden sollte, die vor Auswertungsbeginn in der HANA vorhanden sein müssen. Hierbei ist jedoch auch der Aspekt des Datenschutzes nicht außer Acht zu lassen, wie sich an Beispielen der Vergangenheit herausgestellt hat.

Literatur

[AG14] Heyde (Schweiz) AG. In-memory versus olap, 2014. `http://www.heyde.ch/de/bi/data-discovery/in-memory-versus-olap.html` (27.12.2014).

[Bay12] M. Bayer. Computerwoche: In-memory-computing - zwischen it-beschleuniger und nische, 2012. `http://www.computerwoche.de/a/in-memory-computing-zwischen-it-beschleuniger-und-nische,2494293,7` (27.12.2014).

[Ber13] P. Berg, B.; Silvia. *Einführung in SAP HANA. [was ist SAP HANA, und wie funktioniert die In-Memory-Datenbank?; Datenbeschaffung und -modellierung, SAP HANA Client und Datenbankwerkzeuge; inkl. SAP Business Suite und SAP Netweaver BW auf HANA].* Galileo Press, 2. aktual. auflage, bonn edition, 2013.

[Bri13] C. Brich, S.; Hasenbalg. *Kompakt-Lexikon Wirtschaftsinformatik: 1.500 Begriffe nachschlagen, verstehen, anwenden.* SpringerGabler, Wiesbaden, 2013.

[Bus12] Businessobjects4. Sap hana replication data, 2012. `https://businessobjects4.wordpress.com/page/14/` (06.02.2015).

[Cha06] P. Chamoni, P.; Gluchowski. *Analytische Informationssysteme: Business Intelligence Technologien und -Anwendungen: mit 13 Tabellen.* 2006.

[Cla13] U. Clauß. Toll collect sammelt längst daten aller fahrzeuge. *Welt, 07.11.2013*, 2013. `http://www.welt.de/politik/deutschland/article121617584/Toll-Collect-sammelt-laengst-Daten-aller-Fahrzeuge.html` (29.12.2014).

[Far11] K. Farkisch. *Data-Warehouse-Systeme kompakt: Aufbau, Architektur, Grundfunktionen.* Springer, Berlin, 2011.

[Gmb13a] Digital Ratio GmbH. Business intelligence: Olap und in-memory der vergleich, 2013. `http://www.business-intelligence.org/business-intelligence-software/business-intelligence-olap-und-in-memory-%E2%80%93-der-vergleich/` (27.12.2014).

[Gmb13b] Ferchau Engineering GmbH. Sap hana & co. echtzeitdatenbanken auf dem vormarsch, 2013. `http://www.ferchau.de/news/`

details/sap-hana-co-echtzeitdatenbanken-auf-dem-vormarsch-1895/ (06.02.2015).

[Gmb14] POINT. Consulting GmbH. In-memory-olap-datenbanken, 2014. http://www.point-gmbh.com/index.php?id=in-memory-olap (27.12.2014).

[Lan01] F. Langenau. *Microsoft SQL Server 2000: Für Datenbankadministration und -entwicklung*. Markt und Technik, Mnchen, 2001.

[Lei14] R. Leipert. Business intelligence 24: Acid-prinzip, 2014. http://www.business-intelligence24.com/business-intelligence-definition/business-intelligence-software-technologie/business-intelligence-vergleich-oltp-olap/oltp/acid-oltp (27.12.2014).

[Mül13] H. Müller, R.; Lenz. *Business Intelligence*. SpringerVieweg, Berlin, 2013.

[Rög10] M. Röger. *Konzeption und Realisierung eines Data Warehouses zur Analyse chirurgischer Workflows*. Diplomica Verlag, Hamburg, 2010.

[Sch12] A. Schaffry. Die roadmap von sap hana, 2012. http://www.cio.de/a/die-roadmap-von-sap-hana,2901310,3 (27.12.2014).

[SeroD] B2 Business Brain Intelligence Services. Components and architecture of a business intelligence (bi) system, o.D. http://www.b2.adm.br/componentsandarchitectureofabusinessintelligencebisystem.htm (06.02.2015).

[Sim12] M. Simon, T.; Merz. In-memory-computing ein paradigmenwechsel fr das data warehouse? *S@pport, Sonderdruck, Heft 1-2/2012*, 2012. http://www.camelot-itlab.com/fileadmin/user_upload/Presse/Downloads/Sonderdruck_Beitrag_SAP_HANA-Camelot_ITLab-SAPPORT_FEB2012.pdf (28.12.2014).

[SV08] P. Schmidt-Volkmar. *Betriebswirtschaftliche Analyse auf operationalen Daten*. Gabler Edition Wissenschaft, Wiesbaden, 2008.

Abschließende Erklärung

Ich versichere hiermit, dass ich meine individuelle Projektarbeit zum Thema "Die Bedeutung von In-Memory für die Datenbankparadigmen OLTP und OLAP" selbständig und ohne fremde Hilfe angefertigt habe, und dass ich alle von anderen Autoren wörtlich übernommenen Stellen wie auch die sich an die Gedankengänge anderer Autoren eng anlegenden Ausführungen meiner Arbeit besonders gekennzeichnet und die Quellen zitiert habe.

Oldenburg, den 28. Februar 2015 Janine Haase

VERY LARGE
BUSINESS APPLICATIONS
Carl von Ossietzky Universität Oldenburg

Mobilitaetsrelevante Sensorik im Verkehr

Seminararbeit
im Rahmen der Projektgruppe RAPID

Themensteller:	Prof. Dr.-Ing. Jorge Marx Gómez
Betreuer:	Manuel Osmers

Vorgelegt von:	Olga Schwarz Heinrich-Renken-Str.6a 26127 Oldenburg 01511 7271520 olga.schwarz@uni-oldenburg.de

Abgabetermin:	09 März 2015

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einleitung

Mobilität ist ein Grundbedürfnis der Menschheit und eine der Basisvoraussetzungen für wirtschaftliche Entwicklungen. Die daraus resultierende zunehmende Verkehrsdichte auf allen Transportwegen, insbesondere im Stadtverkehr, stellt die Gesellschaft vor eine große Herausforderung. Intelligente Sensoren für den Verkehr spielen demnach eine wichtige Rolle bei der Steuerung des Verkehrsflusses und dienen nicht zuletzt dem Schutz der Insassen. Verkehrsflüsse müssen gesteuert werden, um die Transportsicherheit zu gewährleisten und um ihre Effizienz zu optimierten. Entscheidend ist der Einsatz mobilitätsrelvanter Sensorik im Verkehr. Sensormessungen liefern strukturierte Daten, die eine bestimmte physikalische Größe erfassen. Nach der Verarbeitung und Analyse der Daten können Handlungsempfehlungen und Prognosen für einen bestimmten Anwendungsfall abgeleitet werden. Die exakte Erfassung und Bereitstellung von Verkehrsinformationen tragen dazu bei, den Verkehr neu zu organisieren. Dadurch begründet, hat der verstärkte Einsatz von Sensorsystemen in den letzten Jahren zu einer Flut an Innovationen im Fahrzeug und im Verkehr geführt. Eine treibende Kraft ist das Bestreben, die Sicherheit für die Insassen eines Fahrzeug und andere Verkehrsteilnehmer stetig zu erhöhen (vgl. [Stu03]). Zudem lassen sich folgende Ziele an den heutigen Verkehr ableiten:

- Ressourcenschonend
- Umweltfreundlich
- Größere Mobilität
- Geringeres Verkehrsaufkommen
- Größere Sicherheit
- Weniger Unfälle und Staus
- Reduzierte Verkehrskosten

Automatisierte Verkehrssysteme, die überwachen, lenken oder klassifizieren, schaffen die Voraussetzung für einen fließenden Verkehr. Eine Übersicht über den derzeitigen Entwicklungsstand von Sensorsystemen im Verkehr ist das primäre Ziel der Ausarbeitung. Die Arbeit befasst sich zudem mit den Funktionen verschiedener mobilitätsrelevanter Sensoren und die Einsatzmöglichkeiten intelligenter Sensoren im Verkehr. Der Einleitung schließt sich im Kapitel 2 die Begriffserklärung und Beschreibung der Sensormessung, des Sensorprinzips und der Wirkungskette. In Kapitel 3 wird die aktuell verfügbare mobilitätsrelevante Sensorik vorgestellt und die Funktionsweise beschrieben. Kapitel 4 umfasst die

Beschreibung der Sensorsysteme im Verkehr. Die Darstellung unterschiedlicher Verkehrssensoren bildet den Schwerpunkt des Kapitels. In Kapitel 5 findet sich eine Zusammenfassung der Arbeit.

2 Sensoren

Um Signale erfassen zu können und daraus Informationen über die Umwelt und den inneren Zustand eines technischen Systems zu gewinnen, benötigt man Sensoren. Sie dienen dazu Informationen über physikalische Größen (Kräfte, Temperaturen, Magnetfelder, usw.) aus der Umwelt zu beschaffen, bzw. zu messen. Insgesamt gibt es Sensoren für mehr als 100 physikalische Größen, berücksichtigt man auch Sensoren für verschiedene chemische Substanzen, so geht die Zahl in die Hunderte. Ein Sensor wandelt die zu messende physikalische Größe und ihre Änderungen in elektrische Größen um und verarbeitet diese so, dass sie leicht übertragen und weiterverarbeitet werden können (vgl. [Rod06]). Die Umwandlung von Messsignalen in elektrische Signale ist in Abbildung 1 dargestellt.

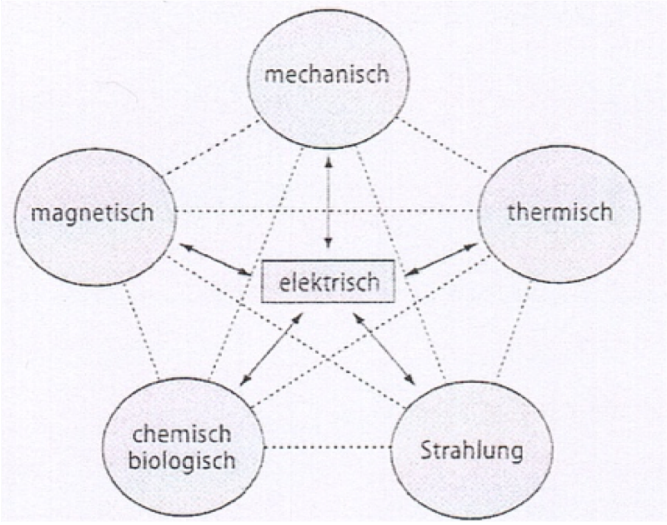


Abbildung 1: Umwandlung von Messsignalen in elektrische Signale aus [HT98]

2.1 Sensormessung

Unter einer Beobachtung (auch Messung) versteht man die Erfassung der Umgebung durch den Sensor. Das Resultat einer Messung ist ein Messvektor, der sich aus den detektierten Messmerkmalen und dem Messzeitpunkt zusammensetzt. Eine Beobachtung reduziert

die Realwelt auf die durch den Sensor detektierbaren Merkmale des Signals und liefert demnach strukturierte Daten (vgl.[Stu03]). Nachgeschaltet ist eine aktive Messschaltung so dass von einem Sensor mit normiertern Ausgangssignal gesprochen werden kann. Wenn dieses Signal zudem noch digital aufbereitet und vor verarbeitet sowie über eine Busschnittstelle verfügbar gemacht wird, spricht man von intelligenten Sensoren (vgl.[Rod06]). Der Zusammenhang und die Ebenen von Sensoren ist in Abbildung 2 dargestellt .

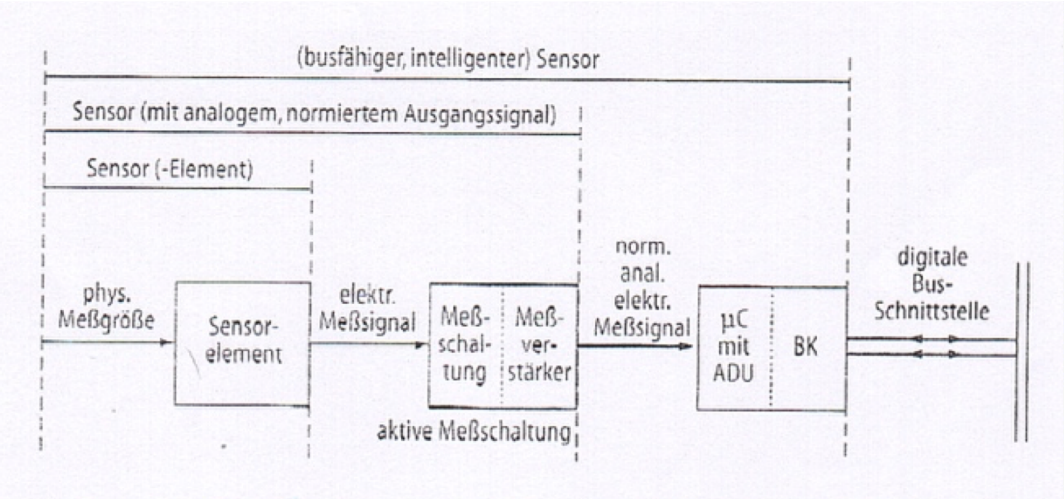


Abbildung 2: Darstellung der Ebenen von Sensoren aus [HT98]

2.2 Sensorprinzipien

Sensoren erfassen physikalische Messgrößen, welche in sogenannten SI-Basiseinheiten angegeben werden. Sensoren erfassen zunächst die zu messende Größe bzw. Einheiten und wandeln sie intern in ein elektrisches Signal um. Dies geschieht primär mit dem Sensorelement. Es gibt Sensoren für die Messung von Wegen, Geschwindigkeiten und Beschleunigungen, die benötigt werden, um den Bewegungszustand eines Systems zu erfassen und zu regeln. Die Abbildung 3 stellt die Sensoren für physikalische Größen dar. In der Regel treten physikalische Größen als analoge Werte auf. Daher liefert ein Messverfahren im Sensor, das die physikalische Größe nicht in eine analoge elektrische Größe, wie beispielsweise eine Amplitude (Spannung, Strom), sondern in eine einfache digitale Größe, wie eine Zählung von digitalen Referenzimpulsen umgesetzt. Bei einem Sensor muss zwischen dem im Aufnehmer verwendeten physikalischen Effekt und der zu messenden Größe unterschieden werden, da der Wert vieler physikalischer Größen nur aus ihrem Einfluss auf bestimmte Messeffekte rückgeschlossen werden kann (vgl.[Rod06]).

Mechanische Größen an Festkörpern	Abstand, Beschleunigung, Dehnung, Dichte, Dicke, Drehmoment, Drehzahl, Druck, Durchmesser, Form, Geschwindigkeit, Gewicht, Kraft, Länge, Höhe, Härte, Masse, Orientierung, Spannung, Weg, Winkel, usw.
Mechanische Größen an Flüssigkeiten und Gasen	Dichte, Druck, Durchfluss, Füllstand, Strömungsgeschwindigkeit, Viskosität, Volumen, usw.
Thermische Größen	Temperatur, Wärmeleitung, Wärmestrahlung, usw.
Optische Strahlung	Farbe, Intensität, Polarisation, Reflexion, Wellenlänge, usw.
Akustische Größen	Absorption, Intensität, Schalldruck, Schallfrequenz, Schallgeschwindigkeit, usw.
Kernstrahlung	Ionisationsgrad, Strahlungsenergie, Strahlungsfluss, usw.
Chemische Größen	Feuchtigkeit, Konzentration, Molekül- oder Ionensorte, Partikelform und -größe, pH-Wert, Reaktionsgeschwindigkeit, usw.
Magnetische und elektrische Größen	Dielektrizitätskonstante, Frequenz, Induktivität, Kapazität, Leistung, Phase, Strom, Spannung, Widerstand, usw.
Sonstige Größen	Anzahl, Pulsdauer, Zeit, usw.

Abbildung 3: Sensoren für physikalische Größen aus [Rod06]

2.3 Wirkungskette

Sensoren werden immer leistungsfähiger. Sie registrieren, auf die von Ihnen feststellbaren Gegebenheiten in ihrer Umgebung. Moderne Sensoren bestehen aber nicht mehr alleine aus einer Einheit zum Detektieren, sondern verfügen über eine eigene Prozessoreinheit. Sie sind somit in der Lage Berechnungen durchzuführen. Ihre Rechenleistung und Speicher sind jedoch eingeschränkt. Des weiteren sind sie in der Lage mit ihrer Umwelt zu kommunizieren. Dies geschieht entweder über Funk oder Kabelleitungen. Ein so zusammengesetzter Sensor kann in eine komplexe Wirkungskette mit Informationsverarbeitungssystem (IVS) wie folgt integriert sein. (vgl. [HT98])

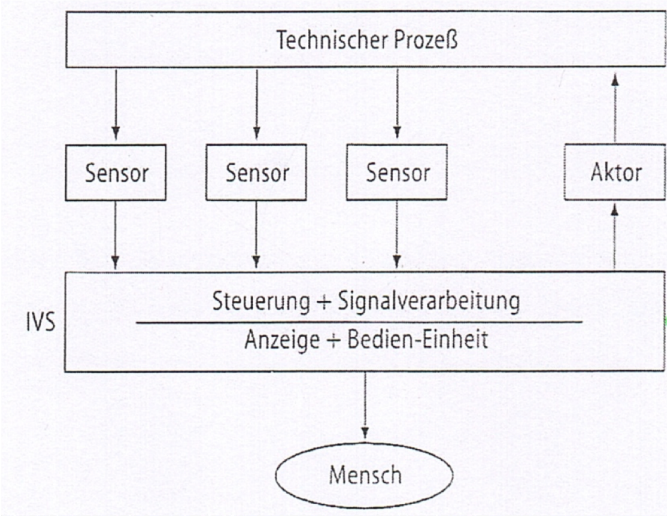


Abbildung 4: Darstellung von Wirkungsketten aus [HT98]

3 Mobilitätsrelevante Sensorik

Nachdem die Sensorprinzipien ausführlich beschrieben wurden, fällt die Aufmerksamkeit in diesem Kapitel auf die mobilitätsrelevante Sensorik und dessen Funktionsweisen. Abbildung 5 bietet einen ersten Überblick über verschiedene mobilitätsrelevante Sensoren, wobei die mit der Funktionsweise verbundenen inhärenten Nachteile dargestellt sind. Die Hauptkomponente der mobilitätsrelevanten Sensorik erschließt sich aus Sensoren für mechatronische Systeme. Da bei mechatronischen Systemen vor allem Sensoren zur Erfassung der Bewegungsgrößen (Lage, Geschwindigkeit, Beschleunigung) und der Bewegungsursache (Kraft, Drehmoment, Druck) von Bedeutung sind, werden in dem folgenden Abschnitt vor allem Sensoren zur Messung dieser Größen behandelt. In Abbildung 5 sind die wesentlichen, zur Detektion von Fahrzeugen eingesetzten Sensoren, aufgeführt.

3.1 Ohmscher Widerstandseffekt

Ohmsche Widerstandseffekte werden bei Potentiometern zur Messung von Wegen oder Winkeln verwendet. Sie sind dadurch gekennzeichnet, dass ihr ohmscher Widerstand durch die jeweilige physikalische Messgröße verändert wird.

3.2 Induktionsprinzip

Die Änderung der Induktivität einer Spule wird schon lange als Messeffekt für bearbeitende Sensoren zur Weg- und Winkelmessung eingesetzt. Bei den meisten wird eine Kombination

Radar	Mikrowellen	Ultraschall	Infrarot	Induktionsschleife	Drucksensor
• hohe Kosten • niedrige Detektionseffizienz für große Objekte • niedrige Auflösung • lange Reaktionszeit	• nicht im Boden installierbar • verursacht leicht elektromagnetische Störungen	• klimatische Abhängigkeit • Verschmutzungsgefahr • nicht im Boden installierbar	• klimatische Einflüsse • Identifikation von Objekten ist kompliziert	• sehr lokal • keine detaillierte Information über das detektierte Objekt • kurze Lebensdauer • Fahrbahn muss zur Installation gesperrt werden	• fragiler Mechanismus • keine Identifikation des detektierten Objekts

Abbildung 5: Zur Detektion von Fahrzeugen eingesetzte Sensoren und Nutzungseinschränkungen aus [HG04]

aus einer Spule, einem Magneten oder Spulenkern und einem zu erfassenden Objekt verwendet. Als Messprinzip wird in der Regel die Änderung des magnetischen Flusses durch Dreh- oder Relativbewegungen benutzt. Es können schon sehr kleine Verschiebungen erfasst werden(vgl.[Rod06]).

3.3 Positionssensor

Bei der Positionsmessung wird im wesentlichen zwischen der Abstands- und Lagemessung unterschieden. Die Abstandsmessung ist in der Regel eine Messung geringerer Distanzen und die Lagemessung die Erfassung der Position eines bewegten Objektes innerhalb eines größeren Bereiches. Die höchsten Messgenauigkeiten kann bei der Messung kleiner Wege erreicht werden. Bestimmte Messeffekte, wie die Beeinflussung der Induktivität einer Spule, oder die Kapazität einer Kondensators, reichen auch nur über relativ geringe Distanzen, liefern aber eine höhere örtliche Auslösung(vgl.[Rod06]).

3.4 Entfernungsmessung mit Ultraschallsensoren

In der Verkehrstechnik werden häufig Sensoren zur Bestimmung von Entfernung, Geschwindigkeit oder Beschleunigung von Objekten im Umfeld des Gerätes benötigt. Hierzu eignen sich Laufzeitentfernungsmesser nach dem Sonar-Prinzip oder das Laserradar (Lidar). Beiden Messprinzipien ist gemeinsam, dass kurze Impulse von Schall-bzw. Lichtwellen ausgesendet werden, die an einem Objekt reflektiert und von einem Empfänger wieder registriert werden können. Auf diese Weise kann durch Bestimmung der Signallaufzeit bei bekannter Ausbreitungsgeschwindigkeit der Wellen die Entfernung zum Reflektor

bestimmt werden. Das geht auch über größere Entfernungen reibungslos (vgl.[Rod06]).

3.5 Entfernungsmessung mit Laser

Lederentfernungsmesser arbeiten nach einem ähnlichen Prinzip. Die Strahlungserzeugung erfolgt durch eine Laser- Diode im Infrarotbereich. Die Reichweite in Luft kann abhängig von der Reflektionseigenschaften der zu detektierenden Objekte bis zu mehreren hundert Metern betragen (vgl.[Rod06]).

3.6 Geschwindigkeitssensoren

Geschwindigkeiten, die in mechatronischen Systemen bestimmt werden müssen, liegen als Linear- oder als Winkelgeschwindigkeiten vor. Da Antriebe für lineare Bewegungen meist rotierende Systeme sind, kann man die Messung von Lineargeschwindigkeiten häufig auch auf Drehzahlmessungen an der Antriebsmaschine rückzuführen (vgl.[Rod06]).

4 Sensorsysteme im Verkehr

Im Verkehrsbereich, wie in vielen anderen Bereichen auch, finden immer mehr Sensoren Einzug. Diese Sensoren liefern ihre Daten als Datenströme. Je nach Anwendungsfall sind allerdings nur Daten eines einzelnen Sensors zu einem bestimmten Zeitpunkt interessant. Eine Aggregation der Daten über die Zeit oder über viele Sensoren ist erwünscht. Die exakte Erfassung und Bereitstellung von Verkehrsinformationen tragen dazu bei, den Verkehr neu zu organisieren (vgl. [Chr14]) In Tabelle 1 sind die unterschiedlichen Sensorsysteme zur Erfassung des Verkehrs aufgelistet und die Anwendungsbereiche kenntlich gemacht.

4.1 Intelligente Verkehrssensoren

Die Verkehrsflusssteuerung obliegt einem geschlossenen Regelkreis, bei dem aus dem momentan vorhandenen Verkehrsfluss abgeleitete Parameter einen Einfluss auf die Steuerungsmechanismen haben, die dann ihrerseits wiederum den Verkehrsfluss beeinflussen. Wie in den vorherigen Kapiteln beschrieben, werden an entscheidenden Positionen des Regelkreises Sensorsysteme benötigt, die auf geeignete Weise den Verkehrsfluss quantifizieren, indem sie entweder einzelne Fahrzeuge in Bezug auf ihre Bewegungsrichtung, Geschwindigkeit und Beschaffenheit erfassen oder aber zumindest kollektive Flusseigenschaften, wie beispielsweise die Anzahl von Fahrzeugen, die pro Stunde eine gewisse Position passiert haben, detektieren. Zur Erfassung der Charakteristika von Verkehrsflüssen und zur Erfassung von Einzelfahrzeugen sind verschiedene Sensorsysteme geeignet. Wie

Sensorsysteme im Verkehr	**Anwendungsbereich**
Lesermesssysteme LMS	tragen dazu bei: • den Verkehr zu steuern • Ampeln zu überwachen • Kapazitäten besser zu nutzen • Vermeidung von Unfällen
Sichtweitenmessgerät	tragen dazu bei: • Unfälle zu vermeiden • Misst präzise die Sichtweite • Verkehrsüberwachung • Vermeidung von Staus • Verkehrsdichte • Abstandsmessung
Überhöhendetektor	Wichtig bei Tunneln und Brücken
Distanzsensor DS 60	schnell und präzise Einparken
Rotlicht/Blitzer	Geschwindigkeitsüberwachung
Mautsysteme	tragen dazu bei: • Liefern die Daten zur Erkennung • Klassifizierung und Separierung von Fahrzeugen
Automatisierte Parkhäuser	

Tabelle 1: Auflistung der Sensorensysteme im Verkehr (vgl. [Sic14])

bereits im vorderen Kapitel erwähnt können diese beispielsweise Radar, Infrarotsensoren, Ultraschallsensoren und Videokameras sein. Für den Straßenverkehr werden hauptsächlich Induktionsschleifen verwendet, die an entsprechenden Positionen in den Belag der Straße eingebracht werden, um beispielsweise Ampelanlagen zu steuern oder Verkehrsflüsse zu quantifizieren (vgl. [HG04]). In den kommenden Unterkapiteln werden 4 Beispiele, die als intelligente Verkehrssensoren gelten, beschrieben.

4.1.1 Verkehrsdichtemessung

Im heutigen Verkehr ist das keine Seltenheit, dass bei unterschiedlichen Anwendungen ein Sensor verwendet wird, um die Zusammensetzung des Straßenverkehrs zu ermitteln. Moderne Technik ermöglicht eine genaue und automatische Ermittlung der Struktur des Verkehrs. Dazu wird ein entsprechender Sensor über oder neben der Straße platziert, um den Verkehr wahrzunehmen und verschiedene Daten aufzunehmen. Dies wird in Abbildung 6 veranschaulicht. Nicht nur die Geschwindigkeit kann dabei gemessen werden, sondern vor allem auch die Verkehrsdichte. Er kann genau erkennen, wie viele Fahrzeuge die jeweilige Straßenstelle passieren und an der Geschwindigkeitsanzeige vorbeifahren. Zudem kann der sogar verschiedene Fahrzeugarten unterscheiden und diese getrennt auswerten. Somit stehen realistische und aussagekräftige Daten für die weitere Verarbeitung zur Verfügung (vgl. [vtcg14]).

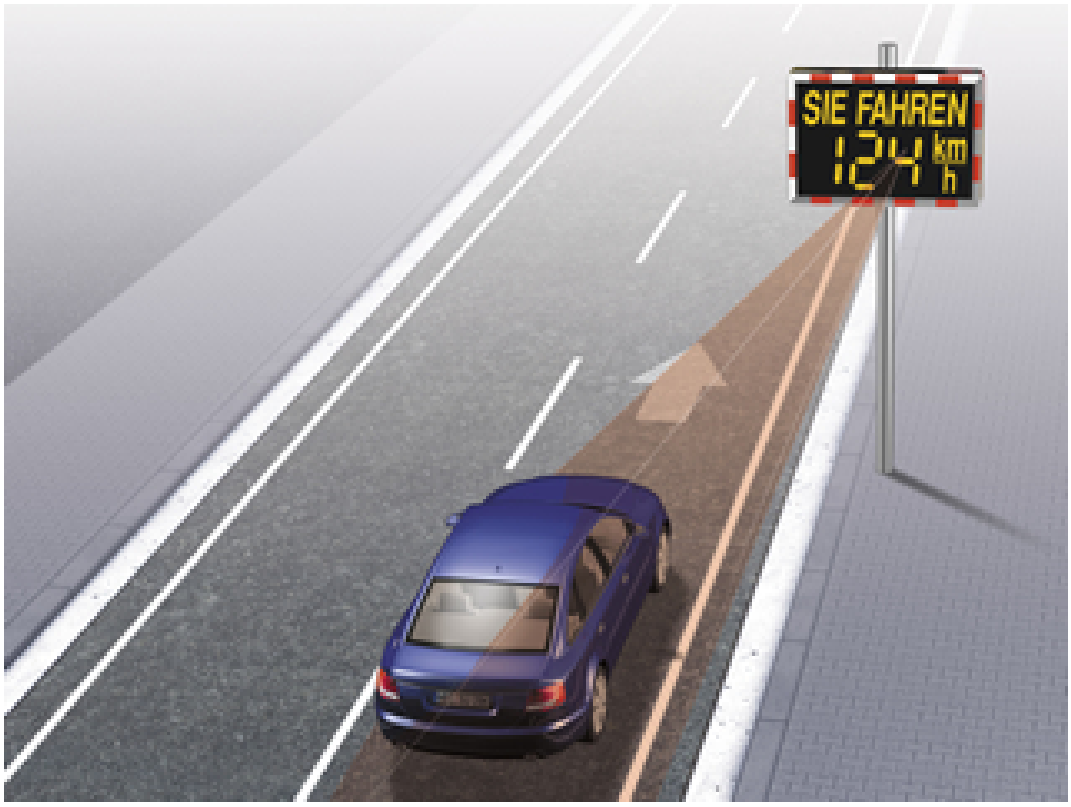


Abbildung 6: Darstellung einer Geschwindigkeitsanzeige aus [vtcg14]

4.1.2 Verkehrszählung

Sensoren werden in vielen Fällen zur Verkehrszählung verwendet, siehe dafür Abbildung 7 Vorteile des Verkehrszählers ist, dass er rund um die Uhr eingesetzt werden kann, um die Verkehrsdichte an verschiedenen Wochentagen oder zu verschiedenen Tageszeiten zu ermitteln. Dadurch ist es auch möglich, entsprechende Auswertungen mit den durch den Sensor gewonnenen Daten zu erstellen, die verwendet werden können, um die Verkehrssituation in den entsprechenden Gebieten in Zukunft positiv zu gestalten. Besonders hilfreich, kann der Sensor bei große Umbaumaßnahmen im Straßenverkehr in einer Gemeinde, oder bei Großveranstaltungen sein. Die Daten, die mit dem Verkehrszähler gewonnen werde, können in unterschiedlicher Art und Weise weiter verwendet werden (vgl. [vtcg14]).

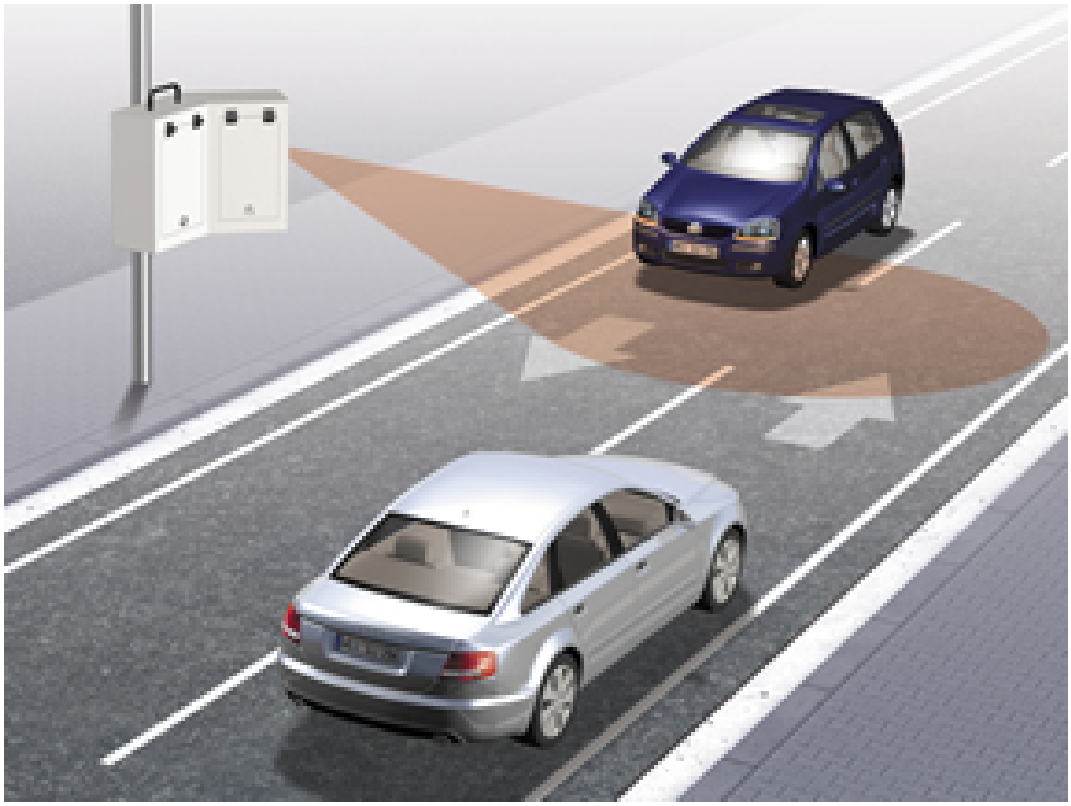


Abbildung 7: Darstellung eines Verkehrzählers aus [vtcg14]

4.2 Verkehr der Zukunft

Durch den verstärkten Einsatz verschiedener intelligenter Sensoren werden neue Möglichkeiten für die Verkehrsoptimierung geschaffen, die im nächsten Abschnitt vorgestellt wer-

den.

4.2.1 Parkplatz-App

Eine Smartphone- App ermöglicht, dass die lästige Suche nach einem freien Parkplatz entfällt und der Verkehr einen freien Parkplatz an das Smartphone sendet. Die Realisierbarkeit eines solchen Komforts für jeden Autofahrer, einen verfügbaren Parkplatz angezeigt zu bekommen, setzt eine Vernetzung ganzer Stadtviertel voraus. In der Innenstand von San Francisco wurde dies als Pilotprojekt realisiert und die Verkehrsbehörde stattet 7.000 Stellplätze mit drahtlosen Sensoren im Asphalt aus. Dadurch werden Suchfahrten-, Lärm- und Abgasbelastungen minimiert und es lassen sich ganze Verkehrsflüsse steuern (vgl [Wir14]).

4.2.2 Verkehrsflusssteuerung

Über eine Kombination von 600 Induktionsschleifen verknüpft mit Kameras und den Positionsdaten der Mobiltelefone, lassen sich ganze Verkehrsflüsse steuern. In Stuttgart werden Induktionsschleifen im Asphalt integriert, die zur Erfassung von Autos verwendet werden. Die Daten aus allen drei Sensorsystemen ermöglichen präzise Angaben von Staus (vgl. [Wir14]).

5 Zusammenfassung

Sensoren werden als Datenquellen immer verbreiteter. Sensorik und die damit verbundene Messgrößenerfassung im Verkehr, ist ein wichtiger Bestandteil bei der Verkehrsflussoptimierung. Durch Sensoren werden Größen aus der Umwelt erfasst und für weitere digitale Verarbeitung zur Verfügung gestellt. Im Rahmen der Arbeit sind unterschiedliche Sensortypen und Sensoren betrachtet worden. Schwerpunkt liegt bei mobilitäsrelevanter Sensorik, dessen Funktionsweise und Anwendungsbereiche im Verkehr. Mit Ausblick auf die Zukunft ist festzustellen, dass durch den Einsatz innovativer Sensortechnologien ein hoher Mehrwert für die Verkehrsflussoptimierung und Verkehrsflusssicherheit geboten werden kann.

Literatur

[Chr14] Björn Christmann. Verkehr (i) verarbeitung von sensordaten, 2014.

[HG04] Uwe Hartmann Haibin Gao. Einsatz hochempfindlicher magnetfeldsensoren zur erfassung von fahrzeugen, 2004.

[HT98] E.Obermeier H. Tänkler. *Sensortechnik - Handbuch für die Praxis und Wissenschaft*. 1998.

[Rod06] Werner Roddeck. *Einführung in die Mechatronik*. 2006.

[Sic14] Sick. Intelligente verkehrssensoren - starten sie mit uns eine reise zum verkehr der zukunft, 2014.

[Stu03] Dirk Stueker. *Heterogene Sensordatenfusion zur robusten Objektverfolgung im automobilen Straßenverkehr*. 2003.

[vtcg14] via traffic controlling gmbh. traffic controlling, 2014.

[Wir14] Wirtschaftswoche. Das 350-billionen-dollar-projekt, 2014.

Abschließende Erklärung

Ich versichere hiermit, dass ich meine Seminararbeit "Mobilitätsrelevante Sensorik im Verkehr" selbständig und ohne fremde Hilfe angefertigt habe, und dass ich alle von anderen Autoren wörtlich übernommenen Stellen wie auch die sich an die Gedankengänge anderer Autoren eng anlegenden Ausführungen meiner Arbeit besonders gekennzeichnet und die Quellen zitiert habe.

Oldenburg, den 10. März 2015 Olga Schwarz

VERY LARGE
BUSINESS APPLICATIONS
Carl von Ossietzky Universität Oldenburg

Data Mining
Methoden und Werkzeuge

Seminararbeit
im Rahmen des Projektes
Regional Analysis and prediction
Platform by In-Memory Data (RAPID)

Themensteller:	Prof. Dr.-Ing. Jorge Marx Gómez
Betreuer:	M. Sc. Alexander Sandau

Vorgelegt von:	B. Sc. Kamiran Tizyani Gaußstr. 5 27580 Bremerhaven 0176 / 25757862 kamiran.tizyani@uni-oldenburg.de

Abgabetermin:	09. März 2015

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einleitung

Durch die stetige Sammlung von Daten durch Unternehmen und Instituten verdoppelt sich die Menge der Daten in kurzer Zeit. Daten werden gesammelt und gespeichert, um sie später für unterschiedliche Zwecke zu nutzen. In einer Bank werden z.B. Transaktionsdaten gespeichert, in einer Firma werden Kundendaten gespeichert, u.a. über das Kaufverhalten des Kunden, ein Wetterdienst sammelt Daten über das Wetter. Die gesammelten Daten können durch leistungsfähige Computer analysiert und ausgewertet werden, um dadurch bedeutungsvolle Informationen zu gewinnen. Für die Analyse und Auswertung der Daten wird das Data Mining eingesetzt. Data Mining bietet Verfahren und Techniken mit Hilfe dessen Zusammenhänge und Muster in den Daten erkannt werden (vgl. [CL14], S. 1 – 2).

Die vorliegende Seminararbeit befasst sich mit dem Themengebiet Data Mining. Zu Beginn der Seminararbeit werden die drei Begriffe Daten, Information und Wissen in Kapitel 2 beschreiben. Im dritten Kapitel werden die zwei Begriffe Data Mining und Knowledge Discovery in Database erläutert. In diesem Kapitel wird auch auf das Prozessmodell und die Einsatzgebiete von Data Mining eingegangen, um den Einstieg in das vierte Kapitel der vorliegenden Seminararbeit zu erleichtern. Im vierten Kapitel werden die Methoden und Techniken von Data Mining detailliert beschrieben und diese anhand von Beispielen erklärt. Im letzten Kapitel der vorliegenden Seminararbeit werden die drei Werkzeuge R-Language, RapidMiner und IBM SPSS beschrieben.

2 Grundbegriffe

In diesem Kapitel wird zunächst auf die drei Begriffe „Daten“, „Informationen“, und „Wissen“ eingegangen. Dazu werden diese kurz erläutert.

2.1 Daten

Daten sind nach der Norm des internationalen Technologiestandards ISO/IEC 2382-1:1993(en) „A reinterpretable representation of information in a formalized manner suitable for communication, interpretation, or processing“ [Pla93]. Sie können u.a. im Rahmen von Experimenten, Befragungen und/oder Beobachtungen etc. erfasst und erhoben werden (vgl. [Sta D]). Daten werden in Form von unstrukturierten, semistrukturierten und strukturierten Daten unterscheiden. Im Allgemeinen versteht man unter strukturierten Daten, ähnlich strukturierte Daten oder relationale Datenbank-Tabellen. Dagegen können unstrukturierte Daten u.a. Bilder oder Texte sein. In Data Mining durchlaufen die unstrukturierten Daten einen komplexen Prozess, da sie vor der Verarbeitung in Form von strukturierten Daten umgewandelt werden müssen (vgl. [CL14], S. 37-38). Bei den semistrukturierten Daten handelt es sich um Daten, die teilweise eine Struktur aufweisen können. Sie weisen allerdings in der Gesamtheit keine eindeutige Struktur auf (vgl. [GHHM11], S. 146).

2.2 Information und Wissen

Man spricht von Informationen, soweit die einzelnen Daten zu nützlichen Einheiten zusammengefasst werden (vgl. [BGS06], S. 31). Sie können durch einen Computer gespeichert, weitergeleitet und anschließend ausgegeben werden (vgl. [Wis14]).

Ist man in der Lage eine Information sinnvoll zu nutzen und zu verarbeiten, so sprechen wir von Wissen.(vgl. [CL14], S. 38).

3 Data Mining und KDD

Im Folgenden Kapitel werden die zwei Begriffe Data Mining und Knowledge Discovery in Database eingegangen. Zunächst wird der Begriff Data Mining definiert, dessen Einsatzgebiete erläutert und anschließend den Prozessmodell grafisch dargestellt und beschrieben. Im Abschluss dieses Kapitels wird schließlich auf den Begriff „Knowledge Discovery in Database“ eingegangen.

3.1 Data Mining Definition

Mit dem Einsatz von Data Mining lassen sich wertvolle Strukturen und Muster aufdecken, die in großen Datenmengen verborgen sind und durch konventionelle Maßnahmen der Datenanalyse nicht identifiziert werden (vgl. [GGP09], S 12-13). Für die Entdeckung von Strukturen und Mustern gibt es vier verschiedene Anwendungsbereiche (Segmentierung, Klassifikation, Prognose und die Assoziation), die eingesetzt werden, um die Beziehungen der Daten festzustellen. Zu den Anwendungsbereichen gehören verschiedene Methoden (u.a. Clusteranalyse, neuronale Netze, Assoziationsanalyse etc.), auf die in Kapitel 4 detailliert eingegangen wird (vgl. [BV08] S. 255).

3.2 Einsatzgebiete des Data Mining

Die Anwendungsfelder des Data Mining sind vielfältig und lassen sich vor allem in der Betriebswirtschaftslehre einsetzen. Mithilfe der neu gewonnenen Erkenntnisse aus den analysierten Daten können Unternehmen ihr zukünftiges Handeln ableiten (vgl. [GGP09], S 139f). Im Folgenden werden einige Anwendungsfelder des Data Mining genannt:

- Marketing
  - Kundensegmentierung
  - Warenkorbanalyse
- Controlling
  - Ergebnisabweichungsanalyse
  - Entdeckung von Controllingsmustern
- Produktion
  - Materialbedarfsplanung
  - Qualitätssicherung
- Finanzdienstleistungen
  - Kreditrisikobewertung
  - Kreditkartenmissbrauch

Je nach Einsatzgebiet des Data Mining werden unterschiedliche Ziele verfolgt. Z.B. im Anwendungsgebiet des Marketings wird die Erhöhung der Kundenbindung im Rahmen des Customer Relationship Management angestrebt (vgl. [GGP09], S 139f).

3.3 Data Mining Prozessmodell

Der Data Mining Prozess besteht aus fünf Phasen, die in Abbildung 1 „Ablauf eines Data-Mining-Prozesses“ dargestellt sind und in diesem Abschnitt kurz erläutert werden.

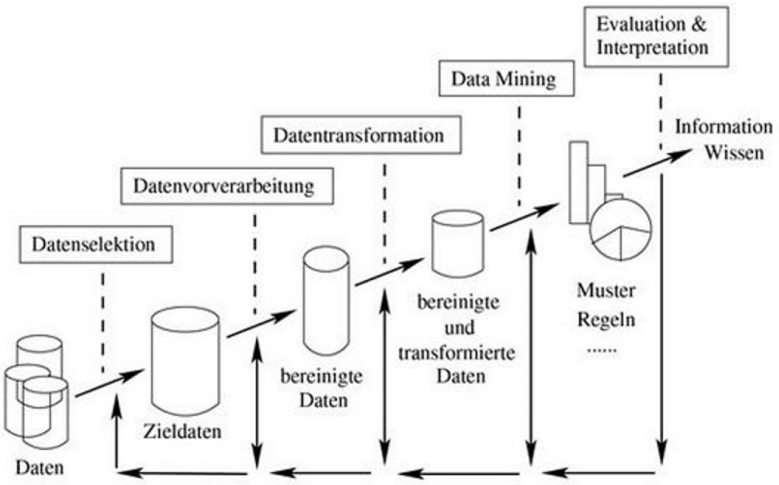


Abbildung 1: Ablauf eines Data-Mining-Prozesses ([CL14], S. 5)

Selektion:
In der ersten Phase werden die Daten zunächst gesichtet und anschließend in eine Datenbank übertragen (vgl. [CL14], S. 5).

Datenvorverarbeitung:
Die Datenvorverarbeitung befasst sich mit der Behandlung fehlender Daten, die aus der Selektion-Phase erhoben wurden. Das Ziel dieser Phase ist es, die erhobenen Daten zu bereinigen, die vorhandenen Fehler zu beseitigen und die fehlenden Werte zu korrigieren und zu vervollständigen (vgl. [CL14], S. 5).

Transformation:
In der dritten Phase erfolgt die Umwandlung der Daten in adäquate Datenformate (vgl. [CL14], S. 5). Hier können z.B. die quantitativen Daten in kategoriale Daten umgewandelt werden. Bei der Umwandlung der Daten muss man sehr vorsichtig sein, damit die vorhandenen Informationen nicht verfälscht werden oder verloren gehen (vgl. [AN00], S. 6-7).

Data Mining:
In der vierten Phase werden die Verfahren des Data Mining eingesetzt. Es wird mit der

Suche nach möglichen Mustern und Beziehungen gestartet. Hierfür wird ein Modell entworfen und eventuell ein Entscheidungsbaum erstellt (vgl. [CL14], S. 5).

Interpretation und Evaluation:
In der letzten Phase werden die neuen Daten geprüft, interpretiert und ausgewertet. Im letzten Prozess wird geprüft, ob die Daten tatsächlich neu und hilfreich sind (vgl. [CL14], S. 5).

3.4 Knowledge Discovery in Database

Data Mining stellt einen Teilprozess des Knowledge Discovery in Databases dar (vgl. [GGP09], S. 13). Der Knowledge Discovery in Database beschreibt dagegen den gesamten Prozess der interaktiven und iterativen Entdeckung und Interpretation von nützlichem Wissen aus Daten (vgl. [BV08], S. 253). Ergänzend zu dem Data Mining Prozess gibt es bei den Knowledge Discovery in Database weitere Teilprozesse, die im Folgenden kurz erläutert werden:

- Zuerst wird das Hintergrundwissen für das jeweilige Fachbereich zur Verfügung gestellt. Anschließend werden die angestrebten Prozessziele identifiziert und definiert.
- Im sechsten Schritt des KDDs, wird das gefundene Wissen anhand eines ausgewählten Modells dargestellt.
- Auf Basis des neu gewonnen Wissens, werden im letzten Schritt Handlungsempfehlungen für die Zukunft festgehalten und weitergegeben (vgl. [iI D]).

4 Data Mining Verfahren

Das vierte Kapitel dieser Seminararbeit befasst sich mit den Data Mining Verfahren. In diesem Kapitel werden die Anwendungsbereiche, Segmentierung, Klassifikation, Prognose und Assoziation detailliert beschrieben. Weiterhin wird auf die Methoden Clusteranalyse, Entscheidungsbäume, künstliche neuronale Netze, k-nächst Nachbar, Diskriminanzanalyse, Regressionsanalyse sowie die Assoziationsanalyse eingegangen und diese anhand von Beispielen erläutert.

4.1 Segmentierung

Die Hauptaufgabe der Segmentierung ist es, die Bildung von Gruppen ähnlicher Objekte. Sie wird vor allem bei der Kundensegmentierung eingesetzt. Die wesentliche Methode der

Segmentierung ist die Clusteranalyse (vgl. [BV08], S. 255).

4.2 Clusteranalyse

Mit Hilfe der Clusteranalyse verfolgt man das Ziel, eine definierte Menge in Teilmengen zu zerlegen (vgl. [CL14], S. 57). Sie wird genutzt, um komplexe Strukturen (Gruppen, Klassen) in hochdimensionalen Räumen, die mit einem Distanzmaß ausgestattet werden können, zu finden (vgl. [Nak98], S. 110). Die Elemente, die sich im Bereich eines Clusters (Gruppe) befinden, sollten sich nach Möglichkeit ähnlich sein. Dagegen sollten sich die Elemente der verschiedenen Clustern (Gruppe) nicht ähneln (vgl. [CL14], S. 57).

Bevor wir uns weiter mit der Clusteranalyse befassen, ist es von großer Bedeutung zunächst die zwei Begriffe „Ähnlichkeits-“ und „Distanzmaße“ kurz zu erläutern.

Während der Bildung von Clustern werden ähnliche Elemente in einer Gruppe zusammengeführt. Damit dies realisiert werden kann, müssen zwei Datensätze auf Ähnlichkeit quantifiziert werden. Dies erfolgt anhand von Distanzmaße. Bei Distanzmaßen wird der Abstand zwischen den Objekten in einem Raum berechnet. Ist eine Distanz zwischen den Datensätzen gering, dann ist die Ähnlichkeit groß. Dagegen ist die Ähnlichkeit gering, wenn die Distanz größer ist (vgl. [CL14], S.43-46).

4.2.1 Arten der Clusteranalyse

Das Cluster-Verfahren unterteilt sich in verschiedene Unterklassen, auf die im Folgenden eingegangen wird (vgl. [CL14], S. 135f):

Partitionierende Clustering:

Mit Hilfe der partitionierenden Cluster kann eine Menge von Elementen in k Clustern zerlegt werden. Es wird zuerst eine Anfangspartitionierung von k Clustern gewählt. Diese Elemente lassen sich nun durch die Ausgangssituation schrittweise zwischen den Clustern austauschen. Hierdurch verbessert sich die Güte der Gruppierung. Der iterative Prozess läuft solange, bis keine Elemente mehr den Clustern zugeordnet werden können. Die wesentlichen Verfahren der partitionierenden Cluster sind die k-Means und k-Medoid (vgl. [CL14], S. 135-136). Sowohl das k-Means als auch der k-Medoid verfolgen das Ziel die Partitionierung der Daten in Cluster. Der wesentliche Unterschied der zwei Methoden ist die Nutzung von Medoiden und Centroiden. Während bei der k-Means-Methode Centroiden als Zentrum des Clusters verwendet werden, werden bei der k-Medoid-Methode Medoiden genutzt. In der k-Menas-Methode wird das Zentrum als gewichteter Mittelwert angese-

hen. Dagegen wird bei der k-Medoids der im Cluster zentralste Datenpunkt als Zentrum angesehen. Im Abschnitt 4.2.2 wird die k-Means-Methode weiter betrachtet (vgl. [Sha13], S. 71).

Hierarchische Clusterbildung

Im Gegensatz zu dem partitionierenden Cluster wird bei der hierarchischen Clusterbildung eine Hierarchie von Clustern aufgebaut. Hierdurch werden die Cluster mit minimalem Abstand und größter Ähnlichkeit gebunden.

Der Ablauf der hierarchischen Clusterbildung wird in Form einer Baumstruktur (Dendrogramm) wie in Abbildung 2 dargestellt.

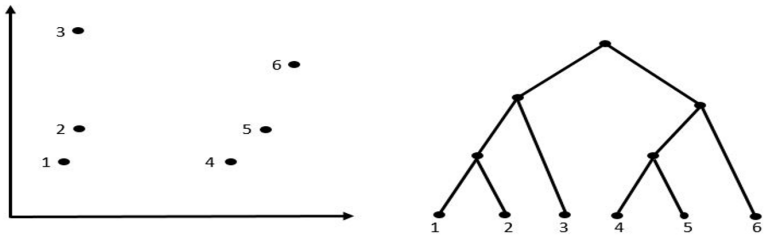


Abbildung 2: Hierarchische Clusterbildung (vgl. [CL14], S. 136)

Die Knoten der Baumstruktur stellen jeweils ein Cluster dar und verfügen über folgende Eigenschaften (vgl. [CL14], S. 136):

- „Die Wurzel repräsentiert die gesamte Datenmenge.
- Ein innerer Knoten repräsentiert die Vereinigung aller Objekte, die in den darunterliegenden Teilbäumen enthalten sind.
- Die Blätter repräsentieren einzelne Objekte“ ([CL14], S. 136).

Die hierarchische Cluster-Analyse wird in agglomerative Clusterbildung und divisive Clusterbildung unterschieden (vgl. [CL14], S. 136).

Agglomerative Clusterbildung

Bei diesem Verfahren startet man mit der niedrigsten Auflösung von Clustern. Zu Beginn beinhaltet jedes Cluster genau ein Element. Anschließend werden zwei sich ähnliche Cluster zu einer höheren Hierarchie gebunden. Der Prozess wird erst beendet, wenn am Ende nur ein Cluster vorhanden bleibt (vgl. [CL14], S. 137).

Divisive Clusterbildung

Anders als bei der agglomerativen Clusterbildung wird bei der divisiven Clusterbildung eine Hierarchie in umgekehrter Reihenfolge ausgebaut. Zu Beginn wird die komplette Datenmenge als ein Cluster behandelt. Anschließend werden sie in kleine hierarchisch tiefe

Cluster eingeteilt. Der Prozess endet, wenn in jedem Cluster nur noch ein Element enthalten ist (vgl. [CL14], S. 137).

Beide Verfahren weisen sowohl Vor- als auch Nachteile auf. Der Vorteil liegt vor allem darin, dass die Menge der Cluster (Gruppe) nicht festgelegt werden muss. Der Nachteil ist, dass bereits gebildete Cluster, die zuvor durch eine Aufteilungs- oder Verschmelzungsentscheidung entstanden sind, nicht mehr auflösbar sind.

Der Einsatz der hierarchischen Clusterbildung eignet sich gut, wenn das Interesse an einer Verbindung unter den Cluster besteht (vgl. [CL14], S. 137).

4.2.2 Der k-Means

Der k-Means-Algorithmus gehört zu den partitionierenden Verfahren der Clusteranalyse. Die Menge der gesuchten Cluster muss bei dem k-Menas Verfahren vordefiniert sein. Auf Basis der Ausgangssituation wird versucht, das Endergebnis durch Zuordnung einzelner Objekte zu Clustern (Gruppen) iterativ zu verbessern. Dieses Verfahren strebt das Ziel der Konstruktion von Clusterzentren an. Weiterhin soll mit dem k-Means-Verfahren durch die Bildung von Clustern die Fehlerstreuung in den Clustern verringert werden.
Im Folgenden werden die Ablaufschritte des k-Means-Verfahrens genannt:

1. Im ersten Schritt wird eine Anfangspartition mit g Clustern gewählt
2. Im zweiten Schritt werden neue Partitionen erzeugt. Dies erfolgt, indem jedes Objekt seinen nächst liegenden Clusterzentrum zugeordnet wird. Für die Bestimmung der Distanzmaße, nutzt man die quadrierte euklidische Distanz.
3. Im dritten Schritt werden die Clusterzentren neu ermittelt
4. Der zweite und dritte Schritt wird weiter wiederholt, bis sich keine neuen Änderungen mehr ergeben (vgl. [AN00], S. 154-155).

4.3 Klassifikation

Während bei der Clusteranalyse die Klassen erst gesucht werden, sind die Klassen bei der Klassifikation bekannt. Bei der Klassifikation werden Objekte anhand ihrer Attributwerte in Klassen eingeteilt (vgl. [ES00], 107).

Mit dem Einsatz der Klassifikation können z.B. Kunden mittels ihrer Daten in verschiedenen Klassen unterteilt werden. Die erste Klasse beinhaltet z.B. Kunden mit normaler Kreditwürdigkeit, in der zweiten Klasse sind Kunden mit sehr guter Kreditwürdigkeit vorhanden. Ein Modell wie in Tabelle 1, bei dem das Merkmal Kreditwürdigkeit bekannt ist, lässt sich mit Hilfe einer vorgegebenen Trainingsmenge von Kundendaten entwickeln (vgl. [CL14], S. 59).

Name	Alter	Einkommen	Kreditwürdigkeit
Adam	≤ 30	Niedrig	Normal
Beate	≤ 30	Niedrig	Sehr gut
Clemens	31...40	Hoch	Sehr gut
Diana	>40	Mittel	Normal
Egon	>40	Mittel	Normal
Frank	31...40	Hoch	Sehr gut

Tabelle 1: Klassifikation - Lernphase (vgl. [CL14], S. 60)

Anhand von Testdaten lässt sich dieses Modell prüfen und falls notwendig auch korrigieren. Die Korrektur erfolgt solange, bis die Testdaten nur noch wenige Fehler aufweisen (vgl. [CL14], S. 60).
Bei der Klassifizierung gibt es verschiedene Analyseverfahren, die je nach Anwendungsbereich eingesetzt werden können. Dazu gehören Entscheidungsbäume, künstliche neuronale Netze, Diskriminanzanalyse (vgl. [GGP09], S. 15), sowie der k-Nearest Neighbour Verfahren (vgl. [CL14], S. 61).

4.3.1 Entscheidungsbaum

Ein Entscheidungsbaumverfahren wird eingesetzt, um Objekte anhand geeigneter Merkmale in Gruppen einzuteilen. Es hat eine Baumstruktur, die aus einer Wurzel, aus Blatt (Knoten), inneren Knoten und Kanten besteht (vgl. [BV08], S.273). Ein Entscheidungsbaum weist die folgenden Eigenschaften auf:

- „ein innerer Knoten repräsentiert ein Attribut,
- ein Blatt repräsentiert eine der Klassen,
- eine Kante repräsentiert einen Test auf dem Attribut des Vaterknotens“ ([ES00], S. 126).

Ein Entscheidungsbaum wird mit Hilfe einer Trainingsmenge in Form einer grafischen Darstellung erstellt. Aus einem grafisch dargestellten Baum können wiederum Regeln abgeleitet werden (vgl. [ES00], S. 126 - 127). Durch die grafische Darstellung lassen sich Entscheidungen besser interpretieren und begründen (vgl. [CL14], S. 71). Im Folgenden wird zum besseren Verständnis ein Beispiel betrachtet:

In einem Versicherungsunternehmen werden zwei unterschiedliche Risikolebensversicherungen (Typ1 und Typ2) angeboten. Die Unterteilung des Versicherungskunden in Typ1 oder Typ2 wird je nach Lebenssituation der versicherten Person empfohlen. Anhand der vorliegenden Versicherungskundendaten, die in Zusammenhang zum Versicherungstyp stehen, kann ein Versicherungsunternehmen die richtige Risikolebensversicherung anbieten. In Tabelle 2 sind einige Beispieldaten der Risikolebensversicherung dargestellt (vgl. [BV08], S. 274).

Nr.	Geschlecht	Alter	Einkommen	Versicherungstyp
1.	Männlich	20	Mittel	1
2.	Weiblich	37	Hoch	1
3.	Weiblich	48	Hoch	1
4.	Männlich	29	Mittel	1
5.	Weiblich	52	Mittel	2
6.	Männlich	42	Niedrig	2
7.	Männlich	61	Mittel	2
8.	Weiblich	26	Niedrig	2

Tabelle 2: Entscheidungstabelle für die Risikolebensversicherung (vgl. [BV08], S. 274)

Anhand der vorgegebenen Daten aus der Entscheidungstabelle kann ein Entscheidungsbaum, wie in Abbildung 3 dargestellt ist, entwickelt werden.

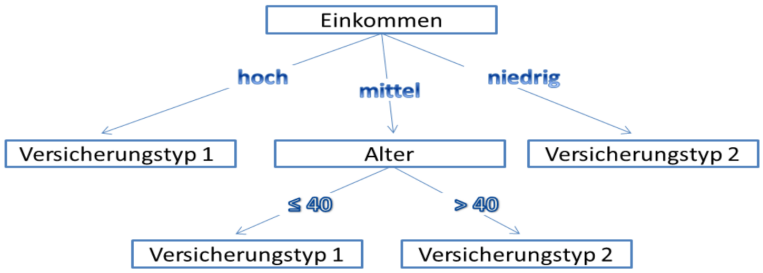


Abbildung 3: Entscheidungsbaum für die Risikolebensversicherung (vgl. [BV08], S. 275)

Als Nächstes werden die Merkmale des Einkommens betrachtet. Aus dem dargestellten Entscheidungsbaum kann eine sofortige Empfehlung gegeben werden, falls das Einkommen hoch oder niedrig ist. Ist dies nicht der Fall, dann wird das Alter des Versicherten geprüft und anhand dessen eine Empfehlung für den Versicherungstyp gegeben (vgl. [BV08], S. 275).

Aus dem Entscheidungsbaum in Abbildung 3 lassen sich die folgenden Regeln ableiten:

- wenn (einkommen = hoch) dann Versicherungstyp1
- wenn (einkommen = niedrig) dann Versicherungstyp2
- wenn (einkommen = mittel) und (alter $\leq$ 40) dann Versicherungstyp1
- wenn (einkommen = mittel) und (alter >40) dann Versicherungstyp2

4.3.2 Künstliche Neuronale Netze

Mit Hilfe eines künstlichen neuronalen Netzes wird ein Wissensspeicher geschaffen, der die Arbeitsweise eines menschlichen Gehirns ähnelt. Es wird aus einer Menge von Neuronen gebildet. Die Neuronen werden miteinander, durch gerichtete und gewichtete Graphen verbunden (vgl. [CL14], S. 47). In Abbildung 4 ist eine typische Struktur eines neuronalen Netzes dargestellt.

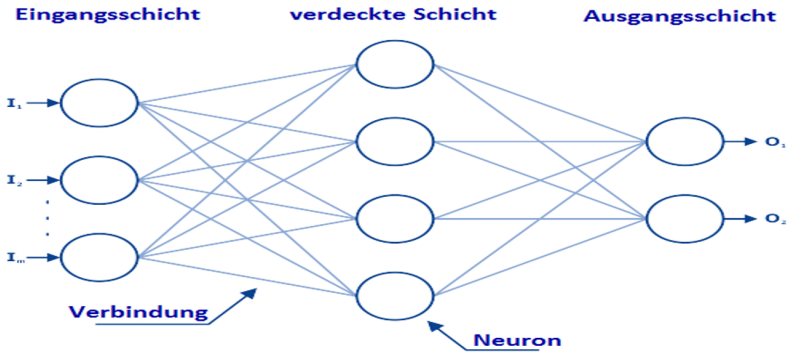


Abbildung 4: Typische Struktur eines neuronalen Netzes (vgl. [Nac D])

Das künstliche neuronale Netz in Abbildung 4 besteht aus einer Eingangsschicht, einer versteckten Schicht und einer Ausgangsschicht. Im Folgenden wird die Funktionsweise des Netzes kurz erläutert: Zuerst werden die ankommenden Eingaben durch die Neuronen zu

Ausgaben verarbeitet. Hiernach werden die Ausgaben mit den anderen Neuronen verbunden. Nun werden mit Hilfe der Eingangsschicht die Informationen in das Netz weitergeleitet. Der Prozess endet, sobald die Signale von allen Schichten des Netzes verarbeitet und die Ausgangsschicht erreicht worden ist (vgl. [Nac D]). Bei den neuronalen Netzen unterscheidet man zwischen einer Trainings- und einer Testphase.

Trainingsphase: In dieser Phase erfolgt eine Anpassung der Gewichte unter den Neuronen. Es gibt unterschiedliche Regeln, die sich in überwachtes und nicht überwachtes Lernen unterteilen. Im Gegensatz zu überwachtem Lernen wird bei unbewachtem Lernen der korrekte Output nicht vorgegeben (vgl. [BH10], S. 137).

Testphase: In der Testphase wird zwischen den Ausgangsreizen und den neuen Reizen unterscheiden. Bei den Ausgangsreizen wird untersucht, ob das Netz die Trainingsdaten gelernt hat. Mit Hilfe der neuen Reize kann festgestellt werden, ob eine Generalisierung des erlernten Wissens erfolgt ist (vgl. [BH10], S. 138).

4.3.3 k–Nearest-Neighbour

Bei dem k-Nearest-Neighbour handelt es sich um ein instanzenbasiertes Verfahren. In diesem Verfahren werden die unbekannten Datenobjekte klassifiziert, um sie auf Ähnlichkeit mit den bereits vorliegenden Datenobjekten einer bekannten Klasse zu untersuchen. Eine Vorhersage der Klassenzugehörigkeit kann getroffen werden, wenn das neue Objekt mit den k nächstliegenden Datenobjekten ähnlich ist (vgl. [CL14], S. 83). Für die Bestimmung der Abstandsmaße wird die euklidische Distanz genutzt. Im Folgenden wird die k-Nearest-Neighbour Methode anhand des Beispiels (Abbildung 5) betrachtet.

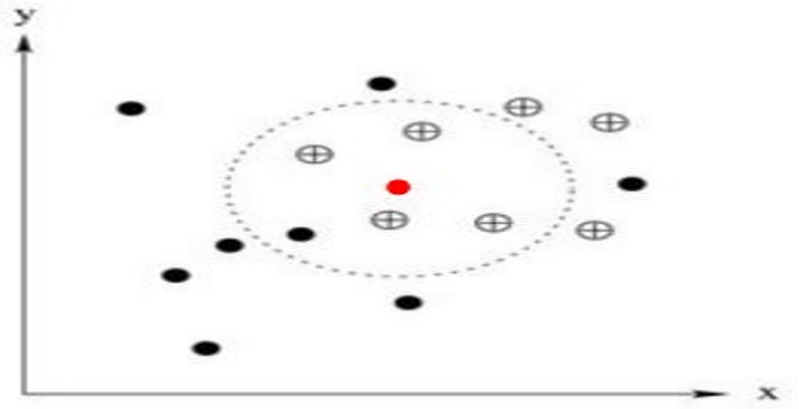


Abbildung 5: k-Nearest-Neighbour 1 (vgl. [CL14], S. 84)

In diesem Beispiel sind zweidimensionale Datensätze vorgegeben. Die in der Abbildung dargestellten Punkte gehören zu zwei Klassen, die unterschiedlich sind. Die in Dunkel dargestellten Objekte gehören zu Klasse 1 und Objekte mit dem Pluszeichen in der Mitte zu Klasse 2. Das Objekt mit der unbekannten Klasse ist rot gekennzeichnet. Um das neue Objekt einer der zwei Klassen zuzuordnen, wird das unbekannte Objekt mit den nächstliegenden fünf bekannten Objekten auf Ähnlichkeit untersucht. Hierdurch wird das unbekannte Objekt der zweiten Klasse zugeordnet, da diesem einen Wert von 4:1 im Gegensatz zu Klasse 1 vorweist (vgl. [CL14], S. 84).
In diesem Verfahren können Konflikte entstehen, sobald sich die Anordnung der Punkte wie in Abbildung 6 dargestellt ist ändert.

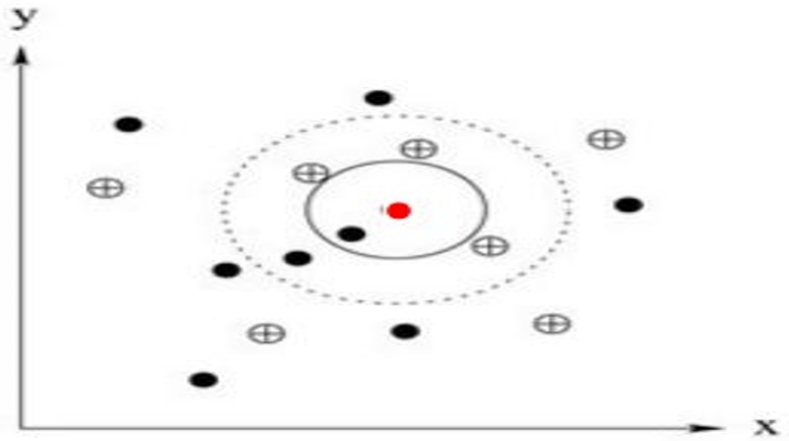


Abbildung 6: k-Nearest-Neighbour 2 (vgl. [CL14], S. 84)

In diesem Beispiel werden zwei unterschiedliche Varianten in Betracht gezogen (k = 1 und k = 5). In der ersten Variante (k = 1), wird das unbekannte Objekt mit nur einem Nachbarn auf Ähnlichkeit betrachtet. Das heißt, dass bei k = 1, das unbekannte Objekt der ersten Klasse zugeordnet wird. Bei der zweiten Variante (k = 5) wird das unbekannte Objekt mit den fünf nächstliegenden Nachbarn auf Ähnlichkeit untersucht. In diesem Fall wird das unbekannte Objekt der zweiten Klasse zugeordnet. Wie sich in diesem Beispiel herausstellt, ist die Wahl der k Objekte für das Endergebnis von großer Bedeutung. Daher sollte man in diesem Verfahren mehrere Möglichkeiten in Betracht ziehen (durch Berechnung k unterschiedlicher Objekte), um das unbekannte Objekt der richtigen Klasse zuzuordnen (vgl. [CL14], S. 84).

4.3.4 Diskriminanzanalyse

Anders als bei der Clusteranalyse in Kapitel 4.2 ist die Gruppenzugehörigkeit bei der Diskriminanzanalyse bereits bekannt. Das Ziel der Diskriminanzanalyse ist die Unterteilung von Objekten zu verschiedenen Klassen, Gruppen, und Teilpopulationen. Die Zuordnung der neuen Objekte in Klassen, Gruppen, und/oder Teilpopulationen erfolgt, nachdem eine optimale, Trennung der Gruppen angesichts der vorhandenen Merkmale durchgeführt worden ist. Die Diskriminanzanalyse kann u.a. bei Erkennung von Krankheiten, bei der Überprüfung von Kreditwürdigkeiten etc. eingesetzt werden (vgl. [EH07], S. 240-245).

4.4 Prognose

Prognosen sind Vorhersagen von Ereignissen. Sie entstehen auf Basis empirischer Beobachtungen der Vergangenheit sowie aus Untersuchungen über Marktsituation, Konkurrenzverhalten und Umfeld-Daten. Mit der Prognose werden Vorhersagen über zukünftige Entwicklungen auf Basis von Ausgangssituationen getroffen. Auf Grundlage einer Vorhersage kann schließlich eine Planung, in der eine Entscheidung getroffen wird, besser geplant und durchgeführt werden. Für ein besseres Verständnis der Vorhersage wird das Beispiel Wetterprognose kurz erläutert: In einem Fernseher oder Radio wird mitgeteilt, wie das Wetter für den nächsten Tag werden soll. Hierdurch trifft der Zuhörer auf Basis der Vorhersage eine der folgenden zwei Entscheidungen:

- wenn am nächsten Tag Regen vorhergesagt wird, dann wird ein Regenschirm benötigt.
- wenn am nächsten Tag kein Regen vorhergesagt wird, dann wird kein Regenschirm benötigt (vgl. [Wir Da]).

Bei der Prognose unterscheidet man zwischen den vier folgenden Klassen:

1. “Intuitive Prognosen (Expertenurteile)
2. Intuitiv-strukturierte Prognosen (z.B. Brainstorming, Delphi-Methode, Szenario-Technik)
3. Induktiv-mathematische Methoden (Regressionsanalyse, Trendverfahren)
4. Deduktiv-nomologische Prognosen (auf der Basis von Gesetzmäßigkeiten)“ ([Wir Da]).

4.4.1 Regressionsanalyse

Bei der Regressionsanalyse handelt es sich um ein statisches Prognoseverfahren. In diesem Verfahren erfolgt eine Untersuchung der Beziehung zwischen einer Zielvariablen und einer

erklärenden Variablen. Das bedeutet, man untersucht die Abhängigkeit einer Variablen Y von mehreren anderen Variablen. Die Regressionsanalyse testet den Einflussfaktor und die Einflussstärke der unterschiedlich unabhängigen Variablen auf die abhängige Variable Y ausüben (vgl. [Wir Db]). In diesem Verfahren wird zwischen der Einfachregression und der Mehrfachregression (auch multiple Regression genannt) unterschieden. Man spricht von einer Einfachregression, soweit ein Merkmal T von einem zweiten Merkmal Z abhängig ist. Dagegen betrachtet man bei der Mehrfachregression die Funktionalbeziehungen zwischen mindestens drei Merkmalen (vgl. [SP12], S. 132).

4.5 Assoziation

In diesem Abschnitt wird die Assoziationsanalyse beschrieben, die Assoziationsregel grafisch dargestellt und erläutert, anschließend anhand eines Beispiels verdeutlicht.

4.5.1 Assoziationsanalyse

Mithilfe der Assoziationsanalyse werden Daten analysiert, um die Beziehung und Abhängigkeit der Daten untereinander zu ermittelten (vgl. [BV08], S. 261). Sie wird vor allem in der Warenkorbanalyse eingesetzt, um festzustellen, welche Produkte gemeinsam gekauft werden. Hierzu müssen die Verkaufsdaten protokolliert werden, um die folgende Aussage treffen zu können: (vgl. [CL14], S. 63) „Wer Produkt A kauft, kauft häufig auch Produkt B“ ([CL14], S. 63). Eine Aussage dieser Form unterstützt ein Unternehmen in der Zukunft, seine Produkte besser anzuordnen (vgl. [CL14], S. 63). Die Warenkorbanalyse verfolgt das Ziel, die Gewinne des Unternehmens zu steigern. Weiterhin kann mit der Warenkorbanalyse auch die Kundenzufriedenheit erhöht werden (vgl. [CL14], S. 64). Die Assoziationsanalyse wird auch in der Medizin genutzt, um eine Wechselwirkung von gleichzeitig eingenommenen Medikamenten zu untersuchen (vgl. [BV08], S. 261).

4.5.2 Assoziationsregeln

Für die Feststellung der Zusammenhänge und Gemeinsamkeiten sind bestimmte Regeln definiert, die in Abbildung 7 dargestellt sind. Für ein besseres Verständnis der Regeln werden zunächst die Begriffe „Menge“, „Transaktion“ und „Datenbasis“ kurz erläutert.

Menge:

Die Menge I = i1,. . . .,im stellt die Ausgangssituation einer Assoziationsanalyse dar. Hierbei kann es sich z.B. um einen Artikel handeln, der in einem Supermarkt angeboten wird (vgl. [BV08], S. 261).

Transaktion:
Mit der Transaktion T wird die Teilmenge von I dargestellt. Eine Transaktion kann z.B. eine Kaufaktion sein, in der der Einkauf eines Kunden mit diversen Artikeln erfasst worden ist (vgl. [BV08], S. 261).

Datenbasis:
Die Datenbasis D ergibt sich aus der Transaktion (vgl. [BV08], S. 261).

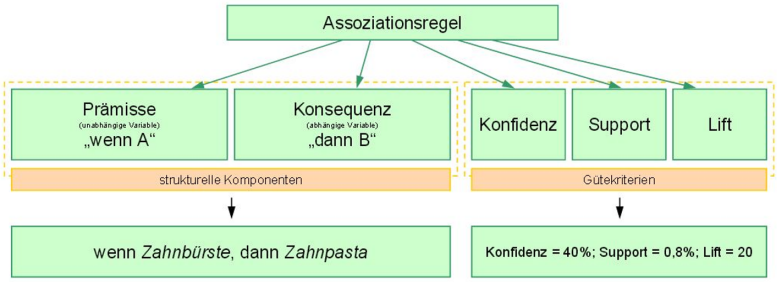


Abbildung 7: Schema einer Assoziationsregel ([Koe04], S. 7)

Support:
„Der Support ist ein Maß für die Häufigkeit, mit der die Kombination aus Vor- und Nachbedingung einer Regel A impliziert B in den Datensätzen auftritt“ ([CL14], S. 223).

Konfidenz:
„Die Konfidenz gibt die bedingte Wahrscheinlichkeit eines Item B bei gegebenen Item A an“ ([CL14], S. 223). Mit dem Einsatz der Konfidenz kann die Stärke der Beziehung zwischen A und B dargestellt werden (vgl. [CL14], S. 223).

Lift:
Mit dem Lift kann festgestellt werden, ob eine Unterscheidung der Verteilung zwischen einem Item in der Gesamt- oder der Teilmenge vorhanden ist (vgl. [CL14], S. 227).

Anhand des folgenden Beispiels soll die Assoziationsanalyse verdeutlicht werden:
In einen Supermarkt worden an der Kasse 100.000 Transaktionen erfasst. In den Transaktionen wurde festgestellt, dass 2.000 Zahnbürsten und 800 Zahnpasta gekauft wurden. Die Assoziationsregel besagt an dieser Stelle (vgl. [Koe04], S. 8): „Wenn Zahnbürste gekauft wird, dann wird auch Zahnpasta ge-kauft“ ([Koe04], S. 8).

Der Support ergibt sich, in dem die Menge der gekauften Zahnpasta durch die gesamte Transaktion geteilt wird (800/100.000 = 0,008). Dies bedeutet, dass in dieser Transaktion 8% der Zahnpasta verkauft wurden.

Die Konfidenz ergibt sich, in dem die Menge der verkauften Zahnpasta durch die Menge der verkauften Zahnbürsten geteilt wird (800/2.000 = 0,4%).

Die Transaktion beinhaltet 2.000 Zahnbürsten. Das bedeutet es wird eine Konfidenz von 0,02 % erwartet, die durch die Teilung der gesamten Menge der Zahnbürsten durch die gesamten Transaktionen ausgerechnet werden (2.000/100.000 = 0,02%).

Der Lift lässt sich aus den Ergebnissen der Konfidenz (Zahnpasta geteilt durch Zahnbürste) und der erwarteten Konfidenz (Zahnpasta geteilt durch Transaktion) berechnen. Das bedeutet 0,4 / 0,02 ergeben 20 (vgl. [Koe04], S. 8).

5 Data Mining Werkzeuge

Im fünften Kapitel werden die drei Data Mining Tools, R-Language, IBM SPSS und Rapidminer erläutert.

Auf der Suche nach einer geeigneten Software für die Data Mining Methoden, wurde festgestellt, dass diverse Tools auf dem Markt angeboten werden (u.a. R-Language, IBM SPSS und Rapidminer, Statistica, Minitab). Während der Suche der Software wurden die vier Kriterien Funktionalität, Benutzbarkeit, Verbreitung und Kosten festgelegt, um die Auswahl der Software zu beschränken. Die zwei Kriterien Funktionalität und Benutzbarkeit wurden aus der ISO Norm 9126 entnommen. Bei der Funktionalität ist es besonders wichtig, dass die Software die nötigen Funktionen für die Untersuchung und Analyse der Daten abdeckt. Bei der Benutzung der Software ist es wichtig, dass die Software leicht zur erlernen, einfach zu bedienen und verständlich ist (vgl. [Ins14]). Die Verbreitung und Nutzung der Software von verschiedenen Unternehmen und Instituten bedeutet, dass die Software bereits häufiger in der Anwendung eingesetzt worden ist. Ein weiterer wichtiger Aspekt der Untersuchung ist die Frage der Kosten: Wird auf dem Markt nur kostenpflichtige Software angeboten oder gibt es auch kostenfreie Software-Lösungen.
Nach Absprache mit dem Betreuer wurde die Auswahl der Software bestätigt.

5.1 R-Language

Bei der ersten Software (R-Langauge) handelt es sich um eine interpretierbare Programmiersprache, mit deren Hilfe Daten analysiert und graphisch dargestellt werden können. Die kostenfreie R Software wurde 1992 an der Auckland Universität von Ross Ihaka und Robert Gentleman entwickelt und den Nutzern im Netz (unter http://www.r-project.org/) zur Verfügung gestellt. R-Language bietet seinen Nutzern Schnittstellen zu unterschiedlicher Software z.B. Statistiksoftware, Datenbanken, MS Office etc. (vgl. [Fae07], S 9 - 10). Des Weitern stehen dem Nutzer ca. 2000 Zusatzpakete zur Verfügung. Bei den Paketen handelt es sich um eine Erweiterung der Funktionen. Einige der Pakete werden automatisch mit der Installation von R, installiert. Mit Hilfe der Pakete werden u.a.

- geographische Informationssysteme genutzt,
- Bilder und Töne bearbeitet,
- Zugriffe auf Datenbanken vorgenommen etc. (vgl. [BP11], S. 14).

Der Nutzer kann Zusatzpakete, die er benötigt, auch selber entwickeln (vgl. [Fae07], S 54). Seit der Entwicklung von R wurden diverse Bücher von verschiedenen Autoren verfasst. Mit Hilfe der Bücher erhält der Nutzer eine Einführung über die Funktionen und Befehle von R und kann diese anhand von Beispielen umsetzten. Im Folgenden werden einige Bücher zur R genannt:

- R Einführung durch angewandte Statistik 2. aktualisierte Auflage (Reinhold Hatzinger, Kurt Hornik, Herbert Nagel und Marco J. Maier)
- Stichproben: Methoden und praktische Umsetzung mit R (Göran Kauermann und Helmut Küchenhoff)
- Einführung in die Statistik mit R (Andreas Behr und Ulrich Pötter)
- Einführung in R: Ein Kochbuch zur statischen Datenanalyse mit R (Günter Faes)

5.2 RapidMiner

Die Entwicklung der YALE Software, einer flexiblen und leistungsfähigen Data-Mining-Software-Umgebung, begann im Jahr 2001 an der Technischen Universität Dortmund. Der Name der Software wurde später von YALE in Rapidminer umbenannt. Rapidminer bietet ihren Nutzern Software-Lösungen und Dienstleistungen für erweiterte Analysen, Vorhersage-Analysen, Data Mining und Text Mining. Die erweiterte Analyse beinhaltet

in-Memory, in-Datenbank und in-Hadoop-Analysen, mit deren Hilfe die Verarbeitung von große Datenquellen vereinfacht wird. Rapidminer wird in verschiedenen Bereichen eingesetzt, die im Folgenden kurz genannt werden:

- Branchen
  - Autoindustrie
  - Bankwesen
  - Versicherung
  - Telekommunikation
- Anwendungsfelder
  - Kundensegmentierung
  - Qualitätssicherung
  - Risikomodellierung
  - Produktanschaffungsneigung

Rapidminer bietet den Anwendern fünf verschiedene Produkte: RapidMiner Studio, RapidMiner Server, RapidMiner Radoop, RapidMiner Streams und RapidMiner Cloud.

Rapidminer Studio

Rapidminer Studio ist eine moderne Analyseplattform und umfasst, Text Mining, Data Mining, Vorhersageanalysen, Business Analysen und maschinelles Lernen.

RapidMiner Server

Beim RapidMiner Server handelt es sich ebenfalls um eine moderne Analyseplattform, mit deren Hilfe skalierbare Rechen- und Zusammenarbeit innerhalb einer Gruppe ermöglicht wird.

Die zwei Produkte RapidMiner Studio und RapidMiner Server werden auf der Homepage in fünf verschiedenen Editionen angeboten:

- Starter
- Personal
- Professional
- Professional Plus

- Enterprise

Die Starter Edition steht den Nutzern kostenfrei zur Verfügung. Alle anderen Editionen können kostenpflichtig erworben werden.
Die Anwendungen von Rapidminer werden in mehr als 50 Ländern eingesetzt. Sie werden u.a. von Lufthansa, PayPal, Pepsi, Sanofi, Siemens, Telenor und Volkswagen verwendet (vgl. [Rap14]).

5.3 IBM SPSS

„IBM ist ein global integriertes Technologie- und Beratungsunternehmen mit Sitz in Armonk, New York“ ([IBM14]). Das Unternehmen wurde im Jahr 1911 gegründet und hat weltweit mehr als 170 Niederlassungen (vgl. ([IBM14]). Auf der Webseite des Unternehmens werden die folgenden vier kostenpflichtigen IBM SPSS Produkte angeboten:

- SPSS Modeler
- SPSS Data Collection
- SPSS Statistics-Produkte
- Analytical Decision Management (vgl. [IBM D]).

IBM SPSS Data Collection dient der Gewinnung von Informationen für Markt- und Umfrageforscher über Einstellungen, Vorlieben und Meinungen. Sie kann direkt vor Ort oder in einer Cloud implementiert werden (vgl. [IBM D]).

Mit Hilfe der IBM Analytical Decision Management können durch Maßnahmenempfehlungen bessere Ergebnisse in Echtzeit besser abgegeben werden (vgl. [IBM D]).
Mit der Plattform IBM SPSS Modeler lassen sich Entscheidungen und Ergebnisse durch vorausblickende Analysen darstellen. IBM SPSS Modeler beinhaltet zahlreiche Methoden, intelligente Algorithmen, Entscheidungsmanagement und -optimierung sowie Entitätsanalyse und Textanalyse. Des Weiteren werden die folgenden Möglichkeiten angeboten (vgl. [Bue14]):

- „Verbesserung von Entscheidungen und Ergebnissen
- Einfachere Wertschöpfung aus Daten
- Leichtere Integration in Ihre bestehenden Systeme ([Bue14])“

Der IBM SPSS Modeler ist in drei verschiedenen Versionen, IBM SPSS Modeler Professional, IBM SPSS Modeler Premium und IBM SPSS Modeler Gold, erhältlich (vgl. [Bue14]). Im Folgenden werden einige Funktionen der verschiedenen Editionen kurz genannt:

- Datenverständnis
- Datenaufbereitung
- Bereitgestellte Modellierungsalgorithmen(u.a. K-Nearest-Neighbour, Apriori Algorithmus, Algorithmen zum Aufbau von Entscheidungsbäumen, k-Means, Diskriminanz, Clustering und Segmentierungsalgorithmen)
- Modellierung und Evaluierung
- Bereitstellung
- Modeler-Server (vgl. [IBM12], S. 5 - 6).
- Text Analyse (für unstrukturierte Datenquellen)
- Objektanalyse (für die Verbesserung der Kohärenz und Konsistenz von Daten)
- Analyse sozialer Netzwerke (vgl. [Bue14]).

Ebenfalls wie bei R-Language wurden seit der Entwicklung der IBM SPSS diverse Bücher von verschiedenen Autoren verfasst. Im Folgenden werden einige Bücher zur IBM SPSS genannt:

- IBM SPSS Syntax „eine anwendungsorientierte Einführung“ 2. Auflage (Autoren: Sarstedt, Marko; Schütz, Tobias; Raithel, Sascha)
- Statistik mit SPSS „Fallbeispiele und Methoden“ 2.Auflage (Autoren: Hatzinger, Reinhold ; Nagel, Herbert)

6 Fazit

Immer mehr Unternehmen und Instituten sammeln und speichern Daten u.a. für wissenschaftliche Zwecke, Wettervorhersagen etc. Mit Hilfe des Data Mining Verfahrens können die gesammelten und gespeicherten Daten analysiert und ausgewertet werden, um daraus wertvolle Informationen zu extrahieren. Durch die Auswertung und Analyse der Daten werden u.a. Wettervorhersagen getroffen, Produkte in Supermärkten besser angeordnet etc. Für die Auswertung und Analyse der Daten bietet Data Mining verschiedene Verfahren und Methoden an, mit deren Hilfe Zusammenhänge und Muster in den Daten untersucht werden. Die Anwendungsbereiche des Data Mining Verfahrens sind vielfältig, dazu gehören u.a. Marketing, Controlling, Produktion und Finanzdienstleistungen.

Die Auswertung und Analyse der Daten erfolgt mit Hilfe von Data Mining Werkzeugen. Auf dem Markt stehen dem Nutzer diverse Data Mining Werkzeuge zur Verfügung. Im fünften Kapitel wurden die drei Data Mining Werkzeuge R-Language, Rapidminer und IBM SPSS dargestellt, die viele Funktionen des Data Mining Bereichs abdecken. Mit Hilfe dieser Funktionen und den bereitgestellten Modellierungsalgorithmen können Daten analysiert und ausgewertet werden, um hierdurch Entscheidungen besser zu treffen. Die Nutzung der zwei Werkzeuge Rapidminer und IBM SPSS ist einfacher, da die Oberfläche einfach dargestellt ist und die Funktionen schnell gefunden werden. Dagegen ist der Einstieg in R-Language aufwändiger, da sich der Nutzer zunächst mit den Befehlen und Funktionen auseinander setzen muss, die für die Nutzung von R-Language notwendig sind.

Literatur

[AN00] Paul Alpar and Joachim Niederreichholz. *Data Mining im praktischen Einsatz. Verfahren und Anwendungsfälle für Marketing, Vertrieb, Controlling und Kundenunterstützung.* Friedr. Vieweg und Sohn Verlagsgesellschaft mbH, Braunschweig Wiesbaden, 1. edition, 2000.

[BGS06] Kurt Badertscher, Josef Gubelmann, and Johannes Scheuring. *Wirtschaftsinformatik Grundlagen. Informations- und Kommunikationssysteme gestalten: Grundlagen mit zahlreichen Illustrationen, Beispielen, Repetitionsfragen und Antworten.* Compendio Bildungsmedien, 1. edition, 2006.

[BH10] Gerhard Bandow and Hartmut H. Holzmüller. *Das ist gar kein Modell!. unterschiedliche Modelle und Modellierungen in Betriebswirtschaftslehre und Ingenieurwissenschaften.* Gabler GWV Fachverlag GmbH, Wiesbaden, 1. edition, 2010.

[BP11] Andreas Behr and Ulrich Pütter. *Einführung in die Statistik mit R.* Verlag Franz Vahlen GmbH München, 2. edition, 2011.

[Bue14] Fabian Buerger. Ibm spss modeler, 2014. `http://www.software-express.de/business-intelligence/ibm-spss-modeler/` (09.01.2015).

[BV08] Udo Bankhofer and Jürgen Vogel. *Datenanalyse und Statistik. Eine Einführung für Ökonomen im Bachelor.* Verlag Dr. Th. Gabler GWV Fachverlag GmbH Wiesbaden, 1. edition, 2008.

[CL14] Jürgen Cleve and Uwe Lämmel. *Data Mining.* Oldenbourg Wissenschaftsverlag GmbH München, 2014.

[EH07] Bärbel Elpelt and Joachim Hartung. *Multivariate Statistik. Lehr- und Handbuch der angewandten Statistik.* R. Oldenbourg Verlag München Wien, 7. edition, 2007.

[ES00] Martin Ester and Jörgen Sander. *Knowledge Discovery in Databeses Techniken und Anwendungen.* Springer-Verlag Berlin Heidelberg, 2000.

[Fae07] Günter Faes. *Einführung in R. Ein Kochbuch zur statistischen Datenanalyse mit R.* Books on Demand GmbH Norderstedt, 2007.

[GGP09] Roland Gabriel, Peter Gluchowski, and Alexander Pastwa. *Data Warehouse und Data Mining*. W3L GmbH Herdecke Witten, 2009.

[GHHM11] Marcus Gebauer, Holger Hinrichs, Knut Hildebrand, and Michael Mielke. *Daten- und Informationsqualität. Auf dem Weg zur Information Excellence*. Vieweg und Teubner Verlag Springer Fachmedien Wiesbaden GmbH, 2. edition, 2011.

[IBM12] IBM. Ibm spss modeler professional. bessere entscheidungen durch geschäftsrelevantes vorhersagewissen, 2012. `http://www.spss-statistics.de/fileadmin/pdf/IBM_SPSS_Modeler_Professional.pdf` (09.01.2015).

[IBM14] IBM. Ibm in deutschland, 2014. `http://www-05.ibm.com/de/ibm/unternehmen/` (09.01.2015).

[IBM D] IBM. Spss software. lösungen und software für predictive analytics, o. D. `http://www-01.ibm.com/software/de/analytics/spss/index.html` (09.01.2015).

[iI D] iBusiness Intelligence. Aktives analysesystem für die wissensentdeckung in daten, o. D. `http://www.ibusiness-intelligence.info/m/business-intelligence/aufbereitung-und-anwendung-der-daten/aktives-analysesystem-fuer-die-wissensentdeckung-in-daten.html` (13.01.2015).

[Ins14] Johner Institut. Software-qualitätseigenschaften, 2014. `https://www.johner-institut.de/blog/tag/iso-9126` (14.01.2015).

[Koe04] Frank Koester. Datawarehousing und knowledge discovery in databases. assoziationsanalyse, 2003/2004. `http://www-is.informatik.uni-oldenburg.de/~koester/Vorlesung/DWH-und-KDD--VL-19---Assoziationsanalyse.pdf` (04.12.2014).

[Nac D] Heribert Nacken. Künstliche neuronale netze, o. D. `http://www.lfi.rwth-aachen.de//index.php?page=kunstl-neuronale-netze` (04.01.2015).

[Nak98] Gholamreza Nakhaeizadeh. *Data Mining Theoretische Aspekte und Anwendungen*. Physica-Verlag Heidelberg, 1998.

[Pla93] Online Browsing Platform. Information technology – vocabulary – part1 1 (1993): Fundamental terms., 1993. https://www.iso.org/obp/ui/#iso:std:iso-iec:2382:-1:ed-3:v1:en (14.01.2015).

[Rap14] Rapidminer. Analytics for anyone, 2014. https://rapidminer.com/ (11.01.2015).

[Sha13] Armin Sharafi. *Discovery in Databases Eine Analyse des Änderungsmanagement in der Produktentwicklung*. Springer Fachmedien Wiesbaden, 2013.

[SP12] Peter M Schulze and Daniel Porath. *Statistik mit Datenanalyse und ökonometrischen Grundlagen*. Oldenburg Wissenschaftsverlag GmbH München, 7.auflage edition, 2012.

[Sta D] Statista. Definition daten, o. D. http://de.statista.com/statistik/lexikon/definition/42/daten/ (01.12.2014).

[Wir Da] Wirtschaftslexikon24. Prognose, o. D. http://www.wirtschaftslexikon24.com/d/prognose/prognose.htm (08.01.2015).

[Wir Db] Wirtschaftslexikon24. Regressionsanalyse, o. D. http://www.wirtschaftslexikon24.com/d/regressionsanalyse/regressionsanalyse.htm (08.01.2015).

[Wis14] IT Wissen. Information, 2014. http://www.itwissen.info/definition/lexikon/Information-information.html (02.12.2014).

Abschließende Erklärung

Ich versichere hiermit, dass ich meine Seminararbeit (Data Mining: Methoden und Werkzeuge) selbständig und ohne fremde Hilfe angefertigt habe, und dass ich alle von anderen Autoren wörtlich übernommenen Stellen wie auch die sich an die Gedankengänge anderer Autoren eng anlegenden Ausführungen meiner Arbeit besonders gekennzeichnet und die Quellen zitiert habe.

Oldenburg, den 9. März 2015 Kamiran Tizyani

VERY LARGE
BUSINESS APPLICATIONS
Carl von Ossietzky Universität Oldenburg

Agiles Projektmanagement: Konzepte, Werkzeuge und Anwendung

Seminararbeit im Rahmen der Projektgruppe RAPID

Themensteller:	Prof. Dr.-Ing. Jorge Marx Gómez
Betreuer:	M.Sc. Daniel Stamer
Vorgelegt von:	B. Sc. Kai Hänig Damm 39 26135 Oldenburg 0151/67306898 kai.haenig@uni-oldenburg.de
Abgabetermin:	09.03.2014

Inhaltsverzeichnis

Abbildungsverzeichnis

1 Einleitung

Diese Arbeit beinhaltet eine Literaturanalyse bezüglich agiler Projektmanagement Methoden, die in diesem Rahmen insbesondere in IT-Umgebungen Anwendung finden. Hierzu wird das traditionelle Projektmanagement anhand von mehreren Beispielen vom agilen Projektmanagement abgegrenzt und erläutert.
Neben der Erörterung des exakten Vorgehensschemas der einzelnen Methoden, wird ebenfalls ein Vergleich zwischen diesen angestellt. Ziel dieser Arbeit ist es ein gutes Verständnis sowohl vom traditionellen- als auch vom agilen Projektmanagement Methoden zu erhalten. Hierbei wird einerseits präzise auf das Wasserfallmodell eingegangen, welches die traditionelle Methodik repräsentiert und den planungsorientierten Ansatz dieser Variante unterstreicht. Auf der anderen Seite werden das Extreme Programming sowie Scrum als Repräsentanten für das agile Projektmanagement betrachtet und die Funktionsweise dieser Methodik näher erörtert und analysieret. Es wird somit festgestellt, welche Anforderungen und welche Gegebenheiten zu welchem Modell führen sollten. Hierbei werden insbesondere die Natur des zu bewältigenden Projektes, als auch das zur Verfügung stehende Personal sowie sämtliche interne- und externe Einflüsse berücksichtigt.
Das Ergebnis dieser Arbeit schlussendlich wird eine Handlungsempfehlung für die Projektgruppe RAPID darstellen. Diese beinhaltet neben einer detaillierten Situationsanalyse auch eine konkrete Empfehlung für eine Projektmanagementmethode, die zusätzlich eine Umsetzungsmöglichkeit aufzeigt, anhand der die Projektgruppe erste Schritte einleiten und das Projekt strukturiert beginnen kann.

Darüber hinaus wird in **Teil 3.2.3** eine Erweiterung für agile Methodiken, das Pair Programming, vorgestellt. Dieses bietet einen sehr interessanten Ansatz, um komplexe Themen, mit denen das Entwicklungsteam noch nicht vertraut ist, effektiv umzusetzen. Hierbei spielt neben der Kommunikation auch die Kollaboration eine sehr wichtige Rolle, wie sich im Folgenden zeigen wird.

2 Projektmanagement

Das Projektmanagement kann als das Initiieren, Planen, Steuern, Kontrollieren und Abschließen von Projekten betrachtet werden [Rou15]. Es beinhaltet somit alle nötigen Schritte, die aus organisatorischer Sicht durchgeführt werden müssen, um ein Projekt jeglicher Art erfolgreich umzusetzen. In dieser Arbeit soll nun das agile Projektmanagement synonym zur agilen Softwareentwicklung verstanden werden, um eine bessere Betrachtung des agilen Prinzips bei der Umsetzung von Projekten in IT Umgebungen zu ermöglichen.
Im Folgenden sollen nun die Bereiche der traditionellen- und anschließend der agilen Softwareentwicklung näher betrachtet werden.

2.1 Traditionelle Softwareentwicklung

Die traditionelle Softwareentwicklung basiert auf dem planungsorientierten Projektmanagement, welches auf der exakten Planung und Vorhersehung sämtlicher Probleme aufbaut. Diese Probleme werden anschließend bewertet und in der Zeit- und Kostenplanung entsprechend berücksichtig.

Die traditionelle Softwareentwicklung ist grundsätzlich aus den folgenden vier Phasen zusammengesetzt [LLTT12, S. 162]. Zu Beginn eines jeden Projektes steht die Aufnahme aller Anforderungen seitens des Auftraggebers. In dieser Phase ist es essentiell, alle potentiellen Möglichkeiten zu eruieren, um sämtliche Probleme, die auftreten könnten vorherzusehen und um diese bereits in den Ablauf mit einzuplanen. Aus dem Lastenheft, welches sämtliche Anforderungen enthält, wird ein Pflichtenheft generiert, auf dessen Basis erfolgt die Festlegung der benötigten Zeit, welche die Implementierung der einzelnen entwickelten Softwarekomponenten in Anspruch nehmen wird [LLTT12, S. 162].
Der zweite Schritt ist der Übergang in die Architektur- und Planungsphase. An dieser Stelle wird die grundlegende Infrastruktur festgelegt, um mögliche technische Probleme zu erkennen und einzuplanen [LLTT12, S. 162f].
Die dritte Phase beinhaltet anschließend die eigentliche Entwicklung des Codes, um den Bedarf des Kunden zu befriedigen. Eine Unterteilung des Gesamtprogrammieraufwandes wird vorgenommen und in einzelne Aufgabenpakete untergliedert. Diese werden entsprechend bearbeitet [LLTT12, S. 163]..
Zuletzt wird die Test-Phase durchlaufen, welche die Qualitätskontrolle und die Erfüllung des Pflichtenheftes zur Aufgabe hat. Dieser Schritt läuft häufig parallel zur Entwicklungsphase ab. An dieser Stelle wird ebenfalls der Kunde in den Prozess mit eingeschlossen und muss die Funktionalität überprüfen und abnehmen. Die traditionelle Softwareentwicklung erfordert eine sehr präzise Planung im Vorhinein, die sich im Verlauf des Projektes ledig-

lich minimal verändern darf. Der Grund dafür liegt darin, dass bei einer Abweichung von maßgebenden Faktoren, wie beispielsweise dem Budget, dem zeitlicher Horizont oder der Qualität, diese negative Effekte für die gesamte Planung nach sich ziehen können. Nichtsdestotrotz können durch die detaillierte Planung die genauen Kosten kalkuliert und ein exakter Zeitplan aufgestellt werden [LLTT12, S. 163].

Um das Prinzip der traditionellen Softwareentwicklung besser zu verdeutlichen, soll im Folgenden das Beispiel des Wasserfallmodells herangezogen werden. Anhand von diesem Modell, können sowohl die hohen Planungsaufwände, als auch die Problematik der Planabweichung besser dargestellt werden.

2.1.1 Das Wasserfallmodell

Das Wasserfallmodell ist eine sequentielle Entwicklungs- und Managementmethode, bei der der Fortschritt durch die Abarbeitung festgelegter Schritte erfolgt. Hierbei gibt es lediglich eine Richtung, wobei eine Rückkopplung bzw. Fehlerkorrektur aus den vorangegangenen Schritten nur sehr begrenzt möglich ist. Der Fokus dieser Methode liegt auf einer sehr präzisen Planung, die alle Eventualitäten vorhersehen muss und bei der jeder Schritt sorgfältig geplant und durchgeführt wird. Erst nach erfolgreichem Abschluss eines Teilschritts, wird der nächste begonnen [Bas12, S. 742]. Somit ergibt sich ein allgemeines Vorgehensmuster, welches bei der Softwareentwicklung die folgenden fünf Schritte, wie in Abbildung 1 dargestellt, beinhaltet.

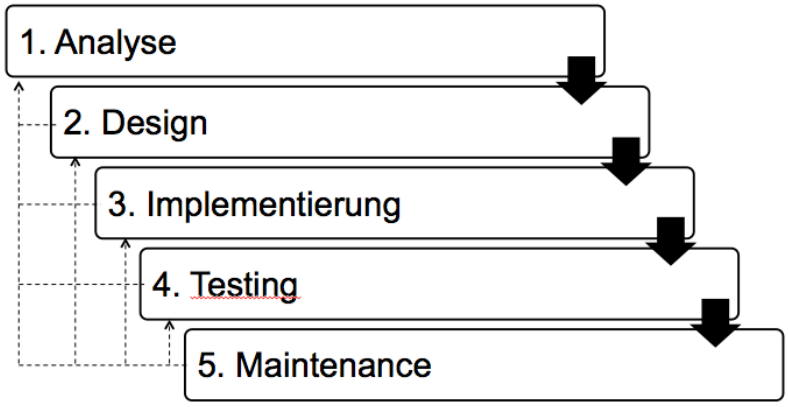


Abbildung 1: Wasserfallmodell [Roy70]

Begonnen wird mit dem ersten Schritt, der Analysephase. In dieser werden sämtliche Anforderungen eines Kunden an das System festgelegt. An dieser Stelle kann ein Kunde

sowohl interner als auch externe Natur entspringen. Bei der Festlegung der Anforderungen, handelt es sich sowohl um funktionelle-, als auch um technische Anforderungen. Ein klarer Umriss des gesamten Bereiches, der von der Software abgedeckt werden muss, wird an dieser Stelle klar definiert und festgehalten. Somit entsteht ein Lastenheft, welches später als Benchmark für den Erfolg des Projektes herangezogen wird. Diese Anforderungen beinhalten somit unter anderen, die Funktionen, die Nutzercharakteristiken, Use-cases, funktionelle Spezifikationen, Schnittstellendefinitionen oder Datenbankdetails [Bas12, S. 743]. Erst wenn diese Analyse vollständig abgeschlossen ist, kann mit der Designphase fortgefahren werden.
In der Designphase wird eine konkrete Lösung für das Problem, inklusive aller Anforderungen seitens des Kunden erarbeitet. An dieser Stelle wird jedoch noch kein Code geschrieben, sondern lediglich alle Szenarien berücksichtigt. Der Output dieser Phase sind beispielsweise die Softwarearchitektur, Algorithmen, Datenbankkonzepte oder grafische Nutzerdesigns. Somit wird ein Pflichtenheft erstellt, welches als Grundlage für die Implementierungsphase genutzt wird [Bas12, S. 743].
In der Implementierungsphase erfolgt die konkrete Umsetzung aller Anforderungen und bereits erstellten Designkonzepte in eine nutzbare Software. Code, Datenbanken und sonstige Anforderungen werden in einen nutzbaren Zustand überführt und nach erfolgreichem Abschluss der Testphase übergeben [Bas12, S. 743].
Die Testphase beinhaltet zwei Hauptaufgaben. Einerseits wird die Software anhand des Pflichtenheftes auf die Nutzbarkeit und Richtigkeit untersucht. Es wird festgestellt ob das Initialziel des Kunden erreicht wurde und die Software den angedachten Zweck innerhalb der gesteckten Parameter erfüllen kann. Zweitens wird die Software auf Bugs überprüft, um den Entwicklern vor Übergabe des Produkts eine Nachbesserung zu ermöglichen und dem Kunden eine hohe Qualität zu sichern [Bas12, S. 743].
Die letzte Phase ist die Wartungsphase. In dieser wird die ausgelieferte Software kontinuierlich von Fehlern befreit, die Leistungsfähigkeit optimiert oder Anpassungen in Form von Schnittstellen durchgeführt [Bas12, S. 743].

Diese fünf Schritte werden immer in derselben Reihenfolge durchlaufen. Im Folgenden sollen nun die Vor- und Nachteile des Modelles gegenüber gestellt werden.

2.1.2 Vor- und Nachteile des Wasserfallmodells

Beginnend mit den Vorteilen, wird beim Wasserfall Modell sehr früh über das Gesamtkonzept nachgedacht und ein exakter Plan erstellt. Dieser umfasst sowohl die Funktionalität einer Software jedoch auch mögliche Probleme oder Fehler. Diese Probleme werden somit in einer sehr frühen Phase der Entwicklung gefunden und können bereits in der Designphase behoben werden. Somit werden hohe Kosten vermieden, die aufgrund eines spät gefundenen Designfehlers entstehen könnten [McC96, S. 72]. Weiterhin wird sehr viel Zeit in die eigentliche Definition der Anforderungen und die Konzeption eines Projektes investiert. Letztendlich wird ein detaillierter Plan erarbeitet, der nicht mehr verändert werden kann, und dem alle Beteiligten zustimmen müssen [Oxa14]. Ein anderes Argument für diesen Ansatz ist die gute Dokumentationsmöglichkeit, die Aufgrund der guten Strukturierung des Projektes und der Transparenz der einzelnen Arbeitsschritte ermöglicht wird. Diese Charakteristiken kommen auch der Einarbeitung neuer Programmierer zugute, die durch den hohen Grad an Übersichtlichkeit und Planung schnell in den Prozess eingearbeitet werden können. Darüber hinaus sorgen die detaillierten Strukturen für ein hohes Maß an Disziplin [Hug09].

Auf der anderen Seite gibt es ein Hauptargument, welches gegen diese Methode, des Wasserfallmodells spricht. Oft ist es bei sehr großen Projekten unmöglich exakte Definitionen über eine zu entwickelnde Software festzulegen, da sich Anforderungen kontinuierlich ändern können oder Fehler auftreten, die nicht vorhersehbar waren. Dies würde einen erneuten Durchlauf der Designphase nach sich ziehen, was aus Projektmanagementsicht nicht gewollt ist. Das Resultat wäre ein immenser Anstieg bei den Kosten und der Zeit. Weiterhin ist es auch möglich, dass in der Designphase das Potential einer Software noch gar nicht vollständig erkannt wird und somit Funktionalitäten nicht konzipiert werden, welche jedoch umsetzbar wären [PC86].
Das Wasserfallmodell repräsentiert die traditionelle Softwareentwicklung. Es gibt angepasste Modelle, wie beispielsweise das V-Modell oder das Incremental-Model. Die Umsetzung weicht bei jedem Modell ab, der grundlegende Ablauf ist jedoch identisch.

Im Gegensatz zur traditionellen und somit konsekutiven Softwareentwicklung steht die agile Softwareentwicklung. Diese soll im nachfolgenden Abschnitt näher untersucht werden.

2.2 Agile Softwareentwicklung

Die agile Softwareentwicklung ist eine Untergruppe des agilen Projektmanagements und beschäftigt sich ausschließlich mit IT-Projekten. Sie wird als neue Denkweise im Projektmanagement charakterisiert und bildet einen Gegensatz zur traditionellen Softwareentwicklung [DDM14, S. 277].
Aufgrund sich schnell ändernder Anforderungen und geringerer Zeiträume in denen Software entwickelt werden muss, ist der Bedarf von Unternehmen nach agilen Methoden oder Frameworks immens gestiegen [MA12, S. 249]. Diese Methoden sollen dem Entwicklungsprozess eine große Flexibilität und Leichtigkeit bieten, wobei die entwickelte Software gleichzeitig anpassungsfähiger und skalierbarer wird [MM11, S. 489].

Diese agilen Methoden und Frameworks basieren auf dem agilen Manifest, welches aus vier Werten besteht und das Fundament für die heutige Nutzung des Wortes agil in Kontext mit agiler Softwareentwicklung bildet [KB12, S. 46]. Diese vier Werte bilden den Grundstein für eine verbesserte Softwareentwicklung.
Der erste Wert besagt, dass das Individuum und die Interaktionen zwischen Individuen stets über den eigentlichen Prozess der Entwicklung und die genutzten Werkzeuge zu stellen ist.
Als zweites ist eine funktionierende Software stets wichtiger als eine umfassende Dokumentation.
Als dritten Wert wird die Zusammenarbeit mit dem Kunden über die Vertragsverhandlungen gestellt und das Reagieren auf Veränderungen als wichtiger erachtet als das Befolgen eines Plans.
Der letztere Wert besagt, dass man nicht strikt dem Plan folgen sollte, sondern auf abweichende Rahmenbedingungen reagieren muss.
Diese vier Werte beschreiben, was heute allgemein als der Inbegriff der agilen Softwareentwicklung angesehen wird [Be01].

Es kann somit festgehalten werden, dass die agile Softwareentwicklung eine inkrementelle beziehungsweise eine iterative Entwicklung darstellt. Dies bedeutet, dass eine Untergliederung des Entwicklungsprozesses in Inkremente oder Iterationen stattfindet, wobei jede Phase der agilen Entwicklung den Prozess der traditionellen Softwareentwicklung im kleinen Maße und auf eine spezifische Aufgabe fokussiert durchläuft [LLTT12, S. 163].

Es gibt verschiedene Ansätze, wie die Idee der agilen Softwareentwicklung bestmöglich

umgesetzt werden kann. Auf diese Vorgehensmodelle soll später detaillierter eingegangen werden. Vorerst wird jedoch die grundlegende Theorie des agilen Projektmanagements und des agilen Prinzips, sowie dessen Verbindung zur Praxis näher eruiert werden. Um den Nutzen und insbesondere den Bedarf des agilen Projektmanagement zu verstehen, soll das Scope-Triangle herangezogen werden, welches in Abbildung 2 dargestellt ist.

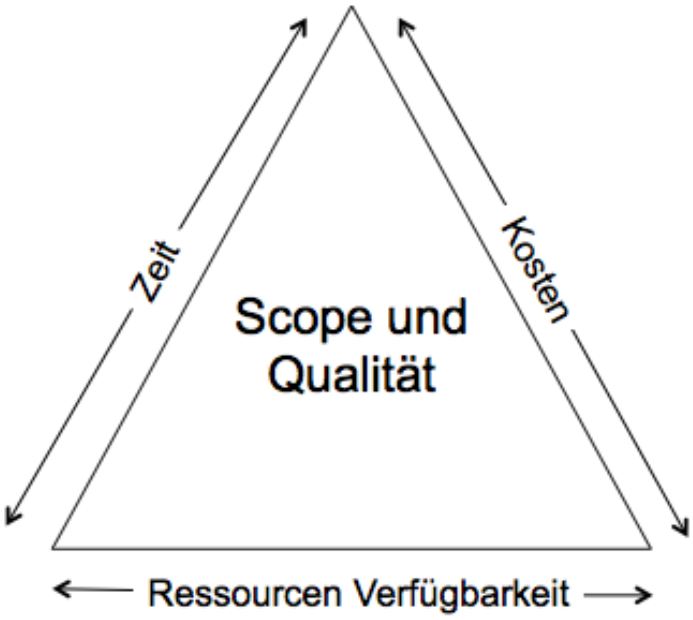


Abbildung 2: Scope-Triangle [Wys09, S. 12]

In diesem Zusammenhang kann das Wort „Scope“ als das Aufgabengebiet eines Projektes angesehen werden, welches einen klaren Rahmen absteckt. Es definiert, welche Aufgaben zu erledigen sind und was nicht Teil des Projektes ist [Wys09, S. 9]. Dieses Aufgabengebiet ist direkt mit der Qualität des Ergebnisses des Projekts, sowie der Qualität der Abwicklung verbunden. Eine Veränderung des Aufgabenbereichs ist jederzeit möglich, jedoch sollte die Qualität stets konstant bleiben. Eine mögliche Veränderung kann sowohl seitens der Anforderungen erfolgen und somit das „Scope“ betreffen, wodurch sämtliche Parameter betroffen sind, andererseits jedoch auch durch einen der drei wichtigsten Parameter an sich hervorgerufen werden. Diese sind Kosten, Zeit und Ressourcen. Die Kosten beschreiben den gesamten monetären Aufwand, den ein Projekt verursacht. Dieser Parameter kann auch als das Budget angesehen werden. Die Zeit ist ebenfalls ein kritischer Faktor, sie beschreibt die maximale Menge an Zeit, die ein Klient einem Projekt zur Verfügung stellt, bis die finale Umsetzung erfolgt sein muss. Als letzter Faktor sind die

Ressourcen zu nennen, die dem Projekt zur Verfügung stehen. Hierrunter fallen Personal, Werkzeuge oder Inventar, welche genutzt werden können [Wys09, S. 11]. Sollte sich einer dieser Parameter minimal verändern, müssen die anderen Parameter sowie das Aufgabengebiet ebenfalls angepasst werden, um das Projekt erfolgreich umzusetzen [Wys09, S. 11f].

Basierend auf dem beschriebenen Scope-Triangle, kann geschlussfolgert werden, dass aufgrund dynamischer Marktgegebenheiten, die sich kontinuierlich und unvorhersehbar verändern, die Notwendigkeit eines unkomplizierten und agilen Projektmanagements essentiell ist. Somit besteht die Möglichkeit, dass Änderungen oder Abweichungen im Projektplan nicht immer auf Fehler individueller Personen zurückzuführen sind, sondern ebenso das Resultat sich verändernder Gegebenheiten sein können [Wys09, S. 13].
Das Projektmanagement hat somit die Aufgabe einerseits allgemeine Veränderungen des „Scopes“ zu kontrollieren und andererseits individuelle Veränderungen der drei Parameter zu managen. Neben den Marktgegebenheiten sollten Individuelle Fehler jedoch nicht außer Acht gelassen werden. Die drei gängigsten Gründe für eine nötige Anpassung des Projektmanagements sind zum ersten eine mangelnde Kommunikation. Beispielsweise melden Teammitglieder nicht, dass sie den Zeitplan für eine überlassene Aufgabe nicht einhalten können, wodurch der Abgabezeitpunkt der individuellen Aufgabe das Projekt nicht direkt eingehalten werden kann.
Ein zweiter Grund ist ein Stillstand der Entwicklung aufgrund der Komplexität der Teilaufgabe. Der Programmierer ist hierbei temporär überfordert und kann seine Aufgabe nicht rechtzeitig fertig stellen.
Ein dritter Grund ist, dass Teammitglieder nach eigenem Ermessen neue Features in das Projekt aufnehmen, da sie davon ausgehen, dass der Kunde diese als nützliche erachten wird. Dies führt ebenfalls dazu, dass Zeitpläne nicht gehalten werden können, da Entwickler zusätzliche Stunden in nicht genehmigte Entwicklungen investieren [Wys09, S. 13f].
Der große Vorteil der agilen Softwareentwicklung kann somit in der verbesserten Kommunikation, schnelleren Umsetzung der Aufgabe und flexiblere Reaktion auf eventuelle Fehler oder sich ändernde Anforderungen und Parameter gesehen werden [KB12, S. 47f].

Im Folgenden soll nun auf spezifische Methoden der agilen Softwareentwicklung eingegangen werden.

2.2.1 Scrum

Scrum ist ein agiles, iteratives Framework für die Softwareentwicklung, welches auf einer absoluten Flexibilität basiert, jedoch gleichzeitig ein ausreichendes Maß an Kontrolle über alle Projektabläufe ermöglicht [Sch95, S. 8]. Scrum wird häufig bei komplexen Projekten eingesetzt, die vielen externen Variablen unterliegen, welche zu Projektbeginn noch nicht feststehen und sich während des Projektablaufes ändern können. Diese Art des Projektmanagements ist optimal für Situationen geeignet, bei denen das Projektteam sehr nah an der Grenze zum Chaos arbeitet, jedoch gleichzeitig ein hohes Maß an Ordnung wahren kann [Sch95, S. 8].
Die Struktur eines Scrum-Teams ist recht einfach und ist in Abbildung 3 dargestellt. Der Product-Owner ist der Projektleiter des gesamten Teams und repräsentiert das Gesamtprojekt gegenüber allen Stakeholdern. Die Hauptaufgaben des Product-Owners sind neben der allgemeinen Kommunikation, die Priorisierung der einzelnen Themen innerhalb des Projektes und die Pflege des Backlogs [Pic10].

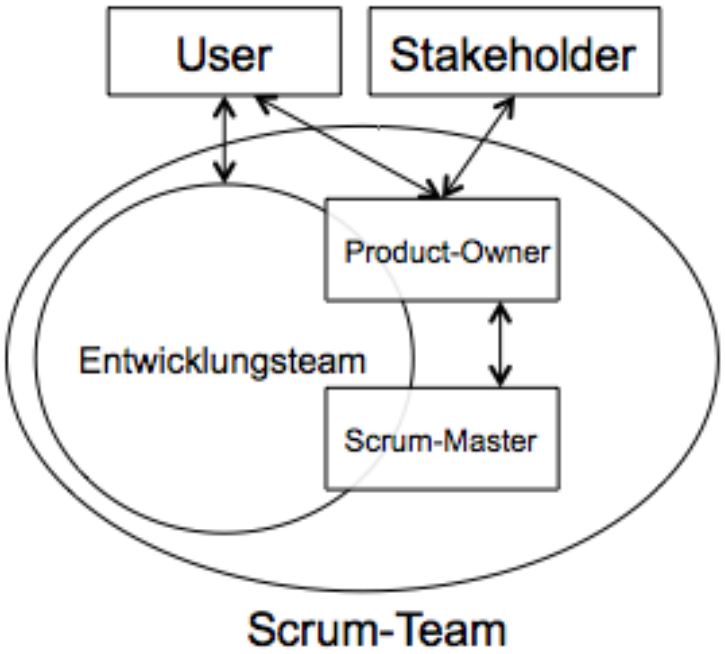


Abbildung 3: Rollenverteilung bei Scrum [Pic10]

Das Backlog repräsentiert dabei sämtliche zu bewältigenden Aufgaben in Form von User Stories, sodass es keine Unklarheiten bei der Umsetzung seitens der ausführenden Seite

gibt. Dieses Backlog enthält sowohl Anforderungen als auch Bug-fixes oder sonstige notwendigen Aufgaben, die vom Projektteam während der Projektdauer umgesetzt werden müssen [Hig09].
Dem Scrum-Team übergeordnet ist der Scrum-Master. Dieser hat die Aufgabe, alle Anforderungen an sein Team umzusetzen und sämtliche äußeren Einflüsse von diesem fern zu halten. Ebenfalls ist der Scrum-Master für die Kommunikation an die Product-Owner zuständig, hat jedoch keine Personalverantwortung innerhalb des Scrum-Teams [DBLV12].
Schlussendlich wird sämtliche Arbeit vom Development-Team umgesetzt. Dieses besteht aus 3-9 Personen, die unterschiedliche Fähigkeiten haben und sich optimalerweise ergänzen. Beispiele hierfür sind Analyse, Design, Testing oder Dokumentation. Jedes Development-Team ist selbstorganisierend und hat somit keinen direkten Projektmanager im klassischen Sinne. Die zeitlichen Entwicklungsintervalle für ein Development-Team sind in Sprints unterteilt.
Bei einem Sprint wird dem Team eine oder mehrere Anforderungen aus dem Backlog und ein Zeitraum zugewiesen, in dem die Aufgaben umgesetzt werden müssen. In welchen Intervallen dies geschieht und wer welche Aufgabe übernimmt, bleibt dem Team selbst überlassen. Sprints werden dazu benutzt das finale Produkt schrittweise zu entwickeln und stellen somit das hohe Maß an Agilität und die iterativer Funktionalität sicher [Sch95, S. 9ff].

Ein Projekt, welches mit Scrum durchgeführt wird, besteht aus drei aufeinanderfolgenden Phasen, wie in Abbildung 4 dargestellt.

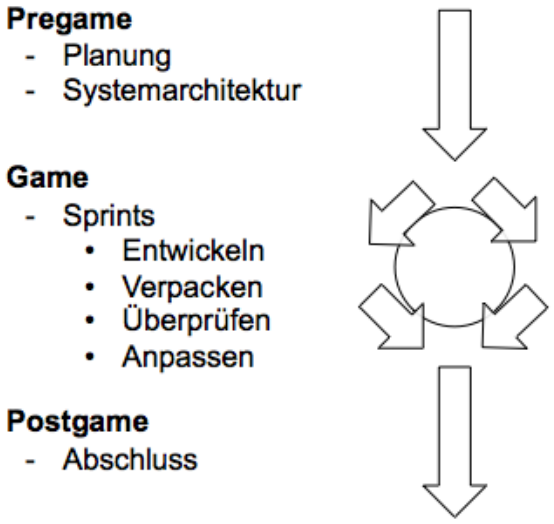


Abbildung 4: Der Scrum-Prozess [Sch95, S. 12]

In der Pregame-Phase wird exakt eruiert, was über das zu entwickelnde System bekannt ist. Ebenfalls wird ein grober Plan aufgestellt, welcher die Erstellung einer Backlogliste, Zeitplänen, Zuweisung der Anforderungen aus der Backlogliste in Releasezyklen, sowie Abschätzung der Kosten der kompletten Umsetzung beinhaltet. Weiterhin wird die grundlegende Architektur festgelegt und mögliche bestehende Systeme so angepasst, dass diese problemlos durch die zu erstellende Software erweiterbar sind[Sch95, S. 12f].
In der zweiten Phase, der Game-Phase, werden die Sprints durchgeführt. Dies geschieht solange, bis die Software auf dem gewünschten Stand ist, beziehungsweise alle Anforderungen für das erste Release abgearbeitet sind. Diese Phase ist somit iterativ und wird kontinuierlich nach dem folgenden Prinzip durchlaufen [Sch95, S. 12].
Zugewiesene Backlog-Anforderungen an das Team werden in Pakete verpackt, sodass diese anschließend entwickelt werden können. Nach erfolgreicher Programmierung werden die Pakete geschlossen und in eine ausführbare Form gebracht. Daraufhin werden die Pakete auf Funktionalität und Qualität überprüft. Schlussendlich werden eventuelle Fehler korrigiert oder nötige Anpassungen für die Integration in bestehende Systeme vorgenommen [Sch95, S. 12]. Nach jedem Sprint kommt das Scrum-Team zusammen, um eventuelle Anpassungen am Backlog vorzunehmen, die sich während des Sprints ergeben haben, und um zu überprüfen ob einerseits das gewünschte Ziel erreicht und andererseits ob alle gesteckten Parameter eingehalten wurden. Die letzte Phase, die Postgame-Phase, beinhaltet anschließend die Vorbereitung der fertigen Software für das Release. Es wird die finale Dokumentation verfasst und Tests durchgeführt. Wenn diese erfolgreich verlaufen, wird die Software released und an den Kunden übergeben [Sch95, S. 12].

Der große Vorteil dieses Framework liegt in der absoluten Flexibilität sowie schnellen Anpassungsfähigkeit in einer sich rapide verändernden Umgebung. Abweichende Variablen wie beispielsweise Budget, Anforderungen oder Zeithorizonte können vom Scrum-Team sehr schnell übernommen werden. Auch kann das Projekt bereits begonnen werden, wenn noch Unklarheiten bezüglich einzelner Parameter herrschen. Diese können zu einem späteren Zeitpunkt problemlos in das Backlog aufgenommen und umgesetzt werden. Neben der Flexibilität besteht auch ein enger Zusammenhalt im Scrum Team, da jedes Teammitglied die Kompetenzen der anderen kennt, wodurch eine effektive Zusammenarbeit ermöglich wird [Sch95, S. 12ff].
Auf der anderen Seite bietet Scrum keine Erfolgsgarantie, da es immer ein hohes Risiko mit sich bringt, ein Projekt zu starten ohne eine sehr präzise Anforderungsliste erstellt zu haben. Ebenfalls ist eine Überprüfung des Projektfortschritts kaum möglich, da es von Beginn des Projektes an zu Differenzen zwischen Soll- und Ist-Zustand kommen wird. Auch

im Projektteam kann es zu Problemen kommen, da eine absolute Gleichstellung und hohe Eigenmotivation im Team herrschen muss, da jedes Mitglied viele verschiedene Aufgaben übernimmt. Somit ist die Stimmung im Team auch ein guter Indikator für dessen Produktivität [Sch95, S. 12ff].

Scrum ist somit eine optimale Methode, um komplexe Projekte schnell zu beginnen und kontinuierlich weiterzuentwickeln. Es ist jedoch eine sehr vage Methode und wird deshalb auch lediglich als Framework bezeichnet, da es wenige fixe Bestandteile gibt, sondern mehr eine grobe Richtung aufzeigt. Neben Scrum gibt es jedoch noch weitere agile Methoden, die einen etwas präziseren Projektablauf vorgeben. Diese Alternativmethode wird im Folgenden betrachtet.

2.2.2 Extreme Programming

Extreme Programming ist eine weitere agile Entwicklungsmethode, die auf den ursprünglichen Abläufen von Scrum basiert. Das Wort „Extreme“ steht in diesem Zusammenhang für die Nutzung von bereits bestehenden Prinzipien auf einem neuen, extremeren Level [ASRW02, S. 19]. Der Lebenszyklus des Extreme Programming besteht aus sechs Phasen, die in Abbildung 5 dargestellt sind. Dieser besteht aus: Exploration, Planning, Iterations, Release, Productionizing, Maintenance und Death.

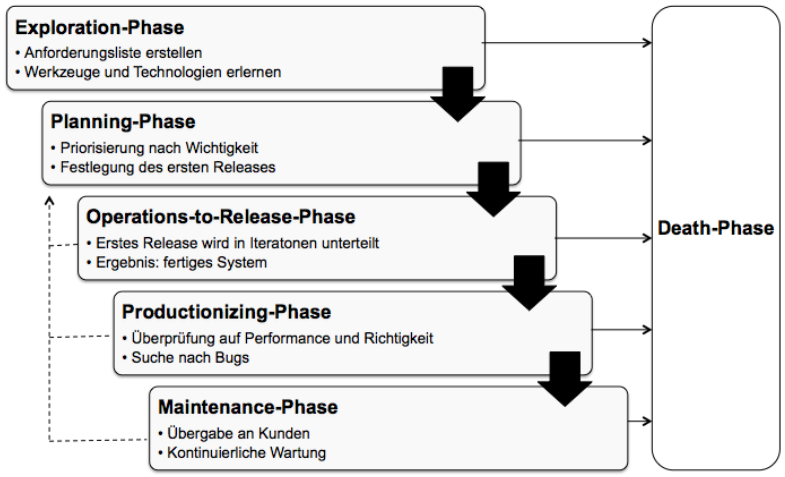


Abbildung 5: Das Extreme Programming Vorgehensmodell [Sch95, S. 12]

Beginnend mit der Exploration-Phase, kommt in dieser das Projektteam zusammen, um Story-Cards zu erstellen, wobei jede Story-Card für ein Feature beziehungsweise eine Anforderung steht. Während dieser Phase lernt das Projektteam alle notwendigen Werkzeuge

und Technologien kennen und erstellt einen ersten Prototyp für eine mögliche Architektur der Software. Diese Phase kann zwischen ein paar Wochen bis zu einigen Monaten dauern. Dies ist abhängig von der Größe und Komplexität des Projektes.
In der Planning-Phase werden anschließend Prioritäten an die einzelnen Story-Cards vergeben und es wird entschieden, welche Features im ersten Release sind. Die Hauptprämisse ist es, das erste Release innerhalb von 2 Monaten durchzuführen, wobei die eigentliche Programmierphase einige Tage nicht überschreiten sollte.
Die Iterations-to-release-Phase beinhaltet die Aufgliederung der festgelegten Features für das erste Release. Die benötigten Iterationen werden in dieser Phase definiert und geplant. Diese Iterationen dauern ein bis vier Wochen und werden vom Projektteam festgelegt. Grundlegendes Prinzip ist es, dass Iterationen, die maßgebend für die Architektur der Software sind, zu Beginn umgesetzt und implementiert werden. Jede Iteration wird auf ihre Funktionalität getestet und abgenommen, sodass am Ende der Iterations-to-release-Phase ein fertiges System steht.
In der vierten Phase, der Productionizing-Phase, werden sämtliche Features auf Performance und Richtigkeit überprüft, bevor eine Übergabe an den Kunden stattfinden kann. Performancefehler oder Bugs, die in dieser Phase entstehen, müssen schnell gefixt werden, wobei eine neue Iteration eröffnet wird, die innerhalb kürzester Zeit durchgeführt werden sollte. Ebenfalls werden sämtliche Änderungen dokumentiert, um spätere Erweiterungen zu erleichtern.
In der fünften Phase, der Maintenance-Phase, wird das System dem Kunden übergeben und in das operative Geschäft überführt. Gleichzeitig werden jedoch weitere Iterationen durchgeführt, um sämtliche Story-Cards, die vom Kunden gefordert sind, umzusetzen und zu implementieren. Durch die kontinuierliche Wartung der bestehenden Software dauern Iterationen von nun an etwas länger. Die letzte Phase ist die Death-Phase. Hierbei wird die komplett fertige Software dem Kunden übergeben, wenn dieser mit der Umsetzung der Features, der Vollständigkeit, der Performance und der Zuverlässigkeit einverstanden ist. Daraufhin wird die finale Dokumentation erstellt und dem Kunden ausgehändigt. Die Death-Phase tritt ebenfalls ein, falls das Projekt aufgrund mangelnder Qualität der Umsetzung oder zu hohen Folgekosten der Weiterentwicklung abgebrochen wird.

Das Modell des Extreme Programming basiert somit einerseits auf dem Wasserfallmodell, welches der konsekutiven Softwareentwicklung entstammt [Bec99, S. 70ff]. Die Phasen werden sukzessiv durchlaufen wobei jede Phase eine spezifische Aufgabe hat. Andererseits basiert die Methode ebenfalls auf Scrum, da durch kontinuierliche Iterationen ein hohes Maß an Flexibilität während der Programmierung ermöglicht wird. Ebenfalls können

eventuelle Fehler, die nicht vorhersehbar waren, schnell behoben oder neue Anforderungen den Zyklen hinzugefügt werden [Sch95, S. 9ff]. Man kann diese Methodik somit als Hybrid zwischen der traditionellen- und der agilen Softwareentwicklung sehen.
Weiterhin soll das Extreme Programming kein allgemeingültiges Prinzip sein, sondern soll für jedes Projekt zugeschnitten werden, sodass es optimal auf spezifische Projekte zugeschnitten ist. Idealerweise wird diese Methode bei kleinen beziehungsweise mittleren Projektteams genutzt, die weniger als 20 Teammitglieder haben [ASRW02, S. 24ff].
Vorteile sind die gute Kommunikation zwischen Teammitgliedern sowie zwischen Team und dem Kunden. Es ist klar ersichtlich, welche Features vom Kunden gefordert werden und durch kurze Entwicklungszyklen wird ein schnelles Feedback ermöglicht, sodass die Programmierung falscher Features umgangen wird [ASRW02, S. 25]. Ebenfalls ist es ein Hauptziel ein schnelles erstes Release durchzuführen, um anschließend darauf aufzubauen. Dadurch kann der Kunden schnell einen Teil der Software in Betrieb nehmen und diese anschließend erweitern. Ebenfalls kommt es zu einem signifikanten Anstieg der Produktivität durch den Einsatz dieser Technik. Sowohl die Anzahl der Code Zeilen, als auch die Anzahl der implementierten Features steigt an [MM02].
Auf der anderen Seite führt eine solch schnelle Entwicklung zu möglichen Problemen bei der Umsetzung der Features nach dem ersten Release. Falls weitere Features seitens des Kunden gefordert werden, die nicht in die bereits existierende Architektur implementierbar sind, werden weitere Iterationen notwendig um eine Lösung zu finden [ASRW02, S. 20f].
Darüber hinaus ist Extreme Programming nicht für jedes Projekt einsetzbar, sodass eine Abwägung vorab unerlässlich ist, ob diese Methode zum Einsatz kommen kann [ASRW02, S. 26].

Somit ist auch das Extreme Programming eine attraktive Möglichkeit, ein Projekt mit Hilfe einer agilen Softwareentwicklungsmethode umzusetzen. Als mögliche Erweiterung von agilen Methoden soll nun das Pair Programming betrachtet werden, welches einen interessanten Ansatz liefert und für eine erhöhte Produktivität des Entwicklungsteams sorgen kann.

2.2.3 Pair Programming

Pair Programming ist durch die Entstehung der agilen Softwareentwicklung immer beliebter geworden. Es ist somit keine eigenständige agile Methodik, sondern ergänzt bestehende Frameworks oder Methoden, wie beispielsweise Scrum oder das Extreme Programming

[Wil08, S. 1].
Das Pair Programming ist eine Programmiertechnik, welche die zu erledigende Aufgabe auf zwei Programmierer aufteilt, die zusammen an einem Rechner arbeiten. Hierbei gibt es zwei Rollen die zu besetzen sind, den Driver und den Navigator. Diese Rollen sollten im Laufe des Projektes kontinuierlich gewechselt werden. Der Driver ist dabei für das eigentliche schreiben des Codes verantwortlich. Er bringt Ideen ein und macht Vorschläge, wie eine Anforderung bestmöglich umgesetzt werden kann. Der Navigator hingegen beobachtet den geschriebenen Code und macht seinerseits Verbesserungsvorschläge oder antizipiert den zu schreibenden Code. Das wichtigste Element bei dieser Methodik ist die kontinuierliche Kommunikation zwischen den beiden Programmierern [CPL+05, S. 44].
Durch die Präsenz eines zweiten Programmierers werden somit bessere Entscheidungen beim Design, der Umsetzung und des Debuggings erreicht. Dies basiert auf der Idee, dass ein einzelner Programmierer kontinuierlich zwischen verschiedenen Programmierstufen wechseln muss. Einerseits muss ein Design erstellt und dessen Überprüfung durchgeführt werden, andererseits parallel grundlegender Programmcode geschrieben werden. Auch die Interpretation der angenommenen Anforderung ist sehr subjektiv und kann mit Hilfe eines zweiten Programmierers wesentlich besser verstanden und umgesetzt werden [CPL+05, S. 44].
Durch die Anwesenheit eines Navigator kann somit eine vorrausschauende Denkweise etabliert werden, sodass dem Driver mehr Zeit für das Coding zur Verfügung steht, da die konzeptionellen Elemente vom Navigator bereits vorbereitet werden können. Diese Methodik bietet insbesondere dann Vorteile, wenn das Programmierteam mit Aufgaben konfrontiert wird, bei denen bisher noch wenige Erfahrungswerte herrschen [CPL+05, S. 45].

Das Pair Programming ist somit eine sehr interessante Alternative, um qualitativ hochwertigeren Code in der gleichen Geschwindigkeit zu erzeugen. Jedoch ist in der bisherigen Forschung weiterhin unklar, in welchen spezifischen Projekten Pair Programming effektiv ist und in welchen weniger, da lediglich bewiesen werden konnte, dass die Effektivität in spezifischen Case-Studies gesteigert wurde, dies jedoch nicht in jedem Fall zutrifft [CPL+05, S. 45].

2.3 Agile- vs. traditionelle Softwareentwicklung

Die agile und die traditionelle Softwareentwicklung unterscheiden sich grundlegend. Der Hauptunterschied besteht darin, dass bei der traditionellen Variante sämtliche Anforderungen bereits vor Beginn des Projektes bekannt sein müssen. Eventuelle Änderungen

während einer der fortgeschrittenen Projektphasen können zu erheblichem Mehraufwand und somit zu hohen Kosten und einer Verzögerung des Projektes führen [McC96, S. 72]. Dies liegt an der konsekutiven Abfolge der einzelnen Phasen, wodurch ein fixer Zeitraum für das Design und die Architektur eingeplant wird und erst anschließend die Programmierung und das Testen stattfindet. Auch bekommen die Stakeholder erst zum Schluss des Projektes ein fertiges Produkt ausgeliefert, welches eventuell nicht mit Sicherheit den exakten Anforderungen entspricht oder es bereits nötige Erweiterungen gibt, die implementiert werden müssen.
Auf der anderen Seite steht die agile Softwareentwicklung, welche diese Probleme durch die Nutzung einer iterativen Methodik zu umgehen versucht. Es werden zu Beginn lediglich grundlegende Ecksteine gelegt, anhand deren die Programmierung bereits zu einem sehr frühen Zeitpunkt startet. Dies liegt daran, dass die Auftraggeber meist zu Beginn des Projektes noch nicht alle Anforderungen kennen, beziehungsweise diese sich während der Projektlaufzeit aufgrund von neuen Technologien ändern können [LLTT12, S. 163]. Durch kontinuierliche Iterationen können stets abweichende Anforderungen durch die Stakeholder eingebracht und vom Entwicklungsteam umgesetzt werden. Auch können fertige Module mit den Kunden auf deren richtige Funktionalität überprüft werden und bei Abweichungen schnell, mit geringem Aufwand, eine Lösung erarbeitet werden. Diese fördert den möglichen Return of Investment, den die Auftraggeber mit dem Projekt zu erreichen versuchen [Ric08].
Jedoch gibt es nicht nur Vorteile der agilen Ansätze im Vergleich zu der traditionellen Softwareentwicklung. Durch die modulare Aufteilung ist es für neue Entwickler häufig sehr schwer sich im Projektablauf zurechtzufinden und die Logik hinter der programmierten Software zu verstehen. Dies führt zu häufigen Fragen seitens der neuen Entwickler an die bereits Eingearbeiteten, was wiederum zu Verzögerungen und somit höheren Kosten führt [Vij08, S. 1ff]. Bei der traditionellen Softwareentwicklung besteht dieses Problem nicht, da es keine Unterteilung in Iterationen gibt, sondern die gesamte Software in einer Phase programmiert wird.
Ein weiterer Nachteil der agilen Methodik sind die engen Zeitintervalle, in denen die Entwickler die Iterationen durchführen müssen. Dies bringt einerseits hohen Druck mit sich und fordert andererseits ein hohes Maß an Kommunikation und Teamwork. Dies bedeutet, dass die interne Teamverbundenheit eine direkte Auswirkung auf die Effektivität hat [LLTT12, S. 165]. Auch die traditionelle Softwareentwicklung unterliegt der guten Gruppenkohäsion, jedoch sind die zeitlichen Rahmen nicht so eng vorgegeben, beziehungsweise gibt es mehr Flexibilität bei der Fertigstellung der Software. Ein direkter Vergleich der beiden Entwicklungsmethoden kann wie in Abbildung 6 zusammengefasst werden.

	Agil	Traditionell
Nutzer Anforderungen	iterative Akquisition	detaillierte Anforderungen müssen zu Beginn des Projektes bekannt sein
Nachbearbeitungskosten	niedrig	hoch
Entwicklungsrichtung	anpassbar	fix
Testen	nach jeder Iteration	nach Abschuss der Entwicklungsphase
Kundeninvolvierung	hoch	niedrig
spezifische personelle Anforderungen an Entwickler	kommunikative Fähigkeiten	keine
Optimaler Projektumfang	klein bis groß	sehr groß

Abbildung 6: Vergleich zwischen traditioneller- und agiler Methodik [LLTT12, S. 165]

Die beschriebenen Aspekte sollen nun nochmals am Beispiel des Wasserfallmodelles als Repräsentant für die traditionelle- und Scrum als Repräsentant für die agile Softwareentwicklung verdeutlicht werden. Der Schwerpunkt liegt insbesondere auf den Faktoren, die zur Nutzung einer spezifischen Methodik führen.
Das Wasserfallmodell, als traditionelles Entwicklungsmodell, fordert eine exakte Anforderungsanalyse zu Beginn eines jeden Projektes. Sämtliche Parameter müssen bekannt sein und werden benötigt um eine umfangreiche Analyse- und Designphase durchzuführen, in denen grundlegende Parameter gesteckt und die Architektur festgelegt wird. Lediglich in diesen Phasen ist ein Eingreifen seitens des Kunden noch möglich. Im Anschluss, in der Implementierungsphase, wird die vollständige Software anhand der Vorgaben programmiert und in den operativen Zustand überführt. Erst zum Schluss, in der Test-Phase, wird dem Kunden die fertige Software präsentiert und auf Funktionalität und Performance überprüft.
Somit kann festgehalten werden, dass sich diese Methodik lediglich für Projekte eignet, bei denen klare Anforderungen, vollständig und bereits beim Start des Projektes bekannt sind. Diese Anforderungen sollten keinen hochvolatilen Markt betreffen, bei dem sich Technologie oder Parameter schnell ändern können, da ein nachträgliches Eingreifen in den Projektablauf lediglich mit hohen Kosten möglich ist. Andererseits bietet diese Methodik eine gute Kosten- und Zeitplanung und ermöglicht gleichzeitig von Beginn an Soll-Ist-Vergleiche, welche bei der Projektsteuerung helfen.
Auf der anderen Seite steht die agile Softwareentwicklung, mit Scrum als Repräsentant. Bei diesem Entwicklungsframework werden dem Projektmanagement viele Freiheiten gelassen und die Annahme getroffen, dass Analyse, Design und Entwicklung nicht vorhersehbar sind [Sch95, S. 10]. Lediglich grundlegende Funktionen können seitens des Kunden zum Projektstart vorgegeben werden. Zu Beginn werden in einer Analysephase alle bereits be-

kannten Anforderungen in einem Backlog zusammengetragen. Dieses kann während des gesamten Projektes beliebig erweitert oder verkürzt werden. Die Entwicklungsorganisation ist in Sprints unterteilt, die 1-4 wöchige Iterationen darstellen, in welchen ausgewählte Anforderungen vom Entwicklungsteam umgesetzt werden. Zum Abschluss dieser Sprints werden Meetings mit dem Kunden abgehalten, in denen Ergebnisse besprochen und der aktuelle Stand des Projektes überprüft wird. Anschließend wird der nächste Sprint durchgeführt. Dies wird solange wiederholt bis der Kunde mit dem Resultat zufrieden ist.
Agile Softwareentwicklungsmethoden sind somit gut für Projekte geeignet, bei denen zu Beginn nicht alle Anforderungen feststehen, beziehungsweise eine Erweiterung der zu entwickelnden Funktionen sehr wahrscheinlich ist. Auch eignet sich diese Methodik gut für High-Tech Märkte in denen schnelle Anpassungen nötig sein können. Auch die Kommunikation zwischen Auftraggeber und Entwicklungsteam ist wesentlich besser als bei traditionellen Methoden.
Negativ hingegen sind die hohen Erwartungen und engen Zeitfenster, die durch die Sprints vorgegeben werden, was bei Entwicklern zu einem erhöhten Stresslevel führt. Auch muss das Scrum-Team ein hohes Maß an Teamwork, Kommunikation und Eigeninitiative mitbringen, um das Konzept erfolgreich umzusetzen, da die Abarbeitung der zugewiesenen Anforderungen selbst organisiert werden muss.

Man kann somit festhalten, dass Scrum ein sehr vielfältig einsetzbares Werkzeug ist, welches dem durchführenden Projektteam gleichzeitig viele Freiheiten einräumt, jedoch auch ein hohes Maß an Kommunikation fordert. Auf der anderen Seite steht die traditionelle Methode des Wasserfallmodells, welches ein gutes Projektmanagement bietet, da eine gute Zeit- und Kosteneinteilung ermöglicht wird, gleichzeitig jedoch einen fixen Rahmen vorgibt, in dem wenig Spielraum für Abweichungen enthalten ist.

3 Fazit und Empfehlung für die Projektgruppe

Beginnend mit einer Situationsanalyse des momentanen Fortschritts der Projektgruppe, kann man festhalten, dass lediglich ein Teil der Gesamtanforderungen zum Zeitpunkt des Verfassens dieser Arbeit bekannt und somit geplant werden können. Es ist somit zu erwarten, dass während der Projektphase noch zahlreiche interne- und externe Anforderungen hinzu kommen werden. Ebenfalls ist ein schneller Beginn der Programmierarbeiten gewünscht, da dem Team lediglich ein kleines Zeitfenster von circa acht Monaten zur Verfügung steht, um die Aufgabe umzusetzen. Darüber hinaus müssen weder exakte Budgetplanung, noch kontinuierliche Soll-Ist-Vergleiche angestellt werden. Ebenfalls ist durch

die geringe Größe des Entwicklungsteams eine grundlegende Flexibilität bereits gegeben. Somit kann man festhalten, dass eine traditionelle Softwareentwicklungsmethode auszuschließen ist, da diese aufgrund der oben genannten Gegebenheiten sehr unvorteilhaft wäre. In Betracht zu ziehen sind somit agile Softwareentwicklungsmethoden, insbesondere Extreme Programming und Scrum.
Die Methode des Extreme Programmings zielt insbesondere auf ein schnelles erstes Release ab, welches direkt in das operative Geschäft des Auftraggebers übernommen werden kann. Dies würde neben zusätzlichen Wartungsaufwänden seitens der Techniker auch eine verlangsamte weitere Entwicklung in der Projektphase bedeuten. Diese Anforderung der schnellen Bereitstellung erster Ergebnisse besteht an die Projektgruppe RAPID jedoch nicht. Weiterhin sind bei dieser Methode umfangreiche Planungsaufgaben vor- und während des Projektstartes nötig, was nicht nur einen hohen Zeitaufwand bedeutet, sondern das Projekt auch aufgrund der Einordnung zwischen traditioneller- und agiler Softwareentwicklung in einen engeren Rahmen zwingt und somit die Flexibilität und Agilität der Entwicklung hemmt [KB12, S. 47].
Das agile Softwareentwicklungsframework Scrum hingegen bietet für diese Projektgruppe viele Vorzüge. Neben der hohen Flexibilität und kontinuierlichen Erweiterbarkeit des Anforderungskataloges, bietet Scrum ebenfalls nicht die klassische Rollenverteilung in einem Projekt. Dies ist sehr wichtig, da durch die geringe Teamgröße mehrere Rollen mehrfach besetzt werden müssen. Es können beispielsweise nicht nur Programmierer, Designer, Tester und Manager festgelegt werden, die lediglich einen Job erledigen. Sondern es müssen sämtliche Anforderungen durch das Team umgesetzt werden, wobei jedes Mitglied verschiedene Rollen bekleiden muss. Der von Scrum angebotene iterative Entwicklungsprozess unterstützt nochmals die Entscheidung für eine agile Methodik, da ein kontinuierlicher Wandel des Backlogs nicht im Vorhinein eingeplant werden kann, sondern stets während der Projektdurchführung aktualisiert werden muss.
Als Erweiterung zur Scrum Methoden, kann innerhalb der Scrum-Teams das Pair-Programming zum Einsatz kommen. Diese bietet neben einer sehr guten Kommunikation zwischen einzelnen Entwicklern, auch die Möglichkeit komplexe Aufgaben im Team effizient zu lösen.

Somit kann als Fazit festgehalten werden, dass der Verfasser dieser Arbeit, der Projektgruppe RAPID die Softwareentwicklungsmethode Scrum empfiehlt. Diese sollte individuell nochmals zugeschnitten werden, wobei ein Product-Owner als Projektleiter mit der Aufgabe der Kommunikation und übergeordneten Organisation des Backlogs betraut wird. Weiterhin sollten zwei Scrum-Teams erstellt werden, die jeweils einen Scrum-Master beinhalten, der den Fortschritt oder die Probleme des Entwicklungsteams an den Product-Owner

kommuniziert. Sobald die grundlegende Organisation errichtet wurde und eine Planungsphase basierend auf allen bereits bekannten Anforderungen durchgeführt und das Backlog aufgesetzt wurde, kann der erste Sprint geplant und umgesetzt werden.

Der Autor sieht somit die Scrum-Methode als das bestmögliche Konzept, um die Projektgruppe RAPID erfolgreich umzusetzen.

Literatur

[ASRW02] Abrahamsson, P. ; Salo, O. ; Ronkainen, J. ; Warsta, J. ; TechTarget, Inc. (Hrsg.): *Agile Software Development methods – Review and Analysis*. http://www2.vtt.fi/inf/pdf/publications/2002/P478.pdf. Version: 2002

[Bas12] Bassil, Y.: A Simulation Model for the Waterfall Software Development Life Cycle. In: *International Journal of Engineering & Technology* (2012)

[Be01] Beck, K. ; et. al.: *Manifest für Agile Softwareentwicklung*. http://www.agilemanifesto.org/iso/de/. Version: 2001

[Bec99] Beck, K.: Embracing Change with Extreme Programming. In: *Computer* (1999)

[CPL+05] Chong, J. ; Plummer, R. ; Leifer, L. ; Klemmer, S.R. ; Eris, O. ; Toye, G.: *Pair Programming: When and Why it Works*. http://www.ppig.org/papers/17th-chong.pdf. Version: 2005

[DBLV12] Deemer, P. ; Benefield, G. ; Larman, C. ; Vodde, B.: *The Scrum Primer*. http://www.infoq.com/resource/news/2013/02/scrum-primer-book-download/en/resources/scrumprimer2_optimized.pdf. Version: 2012

[DDM14] Dybå, T. ; Dingsøyr, T. ; Moe, N.: *Software Project Management in a Changing World*. Springer Berlin Heidelberg, 2014

[Hig09] Higgins, T.: *AUTHORING REQUIREMENTS IN AN AGILE WORLD*. http://www.batimes.com/articles/authoring-requirements-in-an-agile-world.html. Version: 2009

[Hug09] Hughey, D. ; Missouri, University of (Hrsg.): *Comparing Traditional Systems Analysis and Design with Agile Methodologies*. http://www.umsl.edu/~hugheyd/is6840/waterfall.html. Version: 2009

[KB12] Kumar, G. ; Bhatia, P. K.: Impact of Agile Methodology on Software Development Process. In: *International Journal of Computer Technology and Electronics Engineering* (2012)

[LLTT12] Leau, Y. B. ; Loo, W. K. ; Tham, W. Y. ; Tan, S. F.: Software Development Life Cycle AGILE vs Traditional Approaches. In: *IPCSIT* (2012)

[MA12] Mehta, M. ; Adlakha, N.: Manifestation of agile Methods for Prompt Software Development: A Review. In: *International Journal of Research in IT & Management* (2012)

[McC96] McConnell, S.: *Rapid Development*. Microsoft Press, 1996

[MM02] Maurer, F. ; Martel, S.: *On the Productivity of Agile Software Practices: An Industrial Case Study*. `http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=6426456435B5FAB83D9EEA37B3A17A41?doi=10.1.1.19.1925&rep=rep1&type=pdf`. Version: 2002

[MM11] Meredith, J. R. ; Mantel, S. J.: *Project Management: A Managerial Approach*. Wiley, 2011

[Oxa14] Oxagile ; Oxagile (Hrsg.): *Waterfall Software Development Model*. `http://www.oxagile.com/company/blog/the-waterfall-model/`. Version: 2014

[PC86] Parnas, D.L. ; Clements, P. C. ; Victoria, University of (Hrsg.): *A RAational Design Process: How and why to fake it*. `http://www.oxagile.com/company/blog/the-waterfall-model/`. Version: 1986

[Pic10] Pichler, R.: *Agile product management with Scrum : creating products that customers love*. Addison-Wesley Professional, 2010

[Ric08] Rico, Dr. David F.: *What is the ROI of Agile vs. Traditional Methods?* `www.gilb.com/dl244`. Version: 2008

[Rou15] Rouse, M. ; TechTarget, Inc. (Hrsg.): *Project Management*. `http://searchcio.techtarget.com/definition/project-management`. Version: 2015

[Roy70] Royce, W.: Managing the Development of Large Software Systems. In: *IEEE WESCON* (1970)

[Sch95] Schwaber, K.: *SCRUM Development Process*. `http://navegapolis.net/files/Scrum_Development_Process.pdf`. Version: 1995

[Vij08] Vijayasarathy, L. R.: Agile Software Development: A survey of early adopters. In: *Journal of Information Technology Management* (2008)

[Wil08] Williams, L.: *Pair Programming*. `http://collaboration.csc.ncsu.edu/laurie/Papers/ESE%20WilliamsPairProgramming_V2.pdf`. Version: 2008

[Wys09] Wysocki, R. S.: *Effective Project Management: Traditional, Agile, Extreme*. Wiley, 2009

Abschließende Erklärung

Ich versichere hiermit, dass ich meine Seminarausarbeitung im Rahmen der Projektgruppe RAPID selbständig und ohne fremde Hilfe angefertigt habe und dass ich alle von anderen Autoren wörtlich übernommenen Stellen wie auch die sich an die Gedankengänge anderer Autoren eng anlegenden Ausführungen meiner Arbeit besonders gekennzeichnet und die Quellen zitiert habe.

Oldenburg, den 3. März 2015

Potentiale von Big Data für das Customer Relationship Management

Seminararbeit im Rahmen der Projektgruppe RAPID

Themensteller:	Prof. Dr.-Ing. Jorge Marx Gómez
Betreuer:	Dipl.-Oec. Benjamin Wagner vom Berg
Vorgelegt von:	B. Sc. Christian Janßen Myliusstraße 23 26135 Oldenburg 0441/21719228 ch.janssen@uni-oldenburg.de
Abgabetermin:	09.03.2015

Inhaltsverzeichnis

Abbildungsverzeichnis

Abkürzungen

BDSG Bundesdatenschutzgesetz

CIC Customer Interaction Center

CRM Customer Relationship Management

ERP Enterprise Resource Planning

ETL Extract,Transform,Load

GG Grundgesetz

OLAP Online Analytical Processing

OLTP Online Transaction Processing

SCM Supply Chain Management

1 Einleitung

Die stetig steigende Informatisierung stellt Unternehmen in der heutigen schnelllebigen Zeit vor eine große Herausforderung. Dabei kann ein Informationsvorteil gegenüber Konkurrenten eine strategisch wichtigen Bedeutung aufweisen. Aber wie können Informationsvorsprünge gegenüber Konkurrenten erzielt werden? Die Antwort liegt in den vorhandenen Datenmengen, die unterschiedlichste Gegebenheiten beschreiben können. Die bisherigen Technologien stoßen jedoch an die Grenzen des Möglichen, sodass neue Konzepte entwickelt werden müssen. Je schneller Daten analysiert werden, desto größer ist der Informationsgewinn und der Informationsvorteil steigt. Gerade im Bereich der Kundeninteraktion ist ein schnelles Handeln erwünscht, sodass Kunden optimal mit Angeboten oder Informationen versorgt werden können. Was nützen allerdings Daten, die bereits weit zurückliegen? Je mehr Zeit bei der Analyse des Marktes verschwendet wird, desto unflexibler wird die mögliche Handlungsweise des Unternehmens.

In diesem Bereich werden vermehrt Big Data Konzepte eingesetzt. Besonders im Bereich des CRM generiert eine schnelle Analyse vorhandener Kundendaten, einen erheblichen Mehrwert, sodass die angesprochene unflexible Handlungsweise ausbleibt.

Ziel dieser Ausarbeitung ist es, Potenziale bzw. einen Mehrwert von Big Data für das CRM darzustellen. Hierbei wird zunächst eine grundlegende Einführung in die Bereiche Big Data sowie CRM durchgeführt. Für das beiderseitige Verständnis werden zum einen Technologien von Big Data vorgestellt, die im Bereich des CRM nützlich erscheinen und zum anderen Komponenten von CRM Systemen, die einzelne Phasen der Kundenbeziehung sowie des Managements darstellen. Im zweiten Teil der Ausarbeitung werden konkrete Potenziale bzw. Szenarien beschrieben, die einen Einsatz fördern. Abschließend wird eine Beispielarchitektur einer Big Data Anwendung vorgestellt.

2 Hintergrund

Kundenbeziehungen sind für jedes Unternehmen essenzielle wichtig. In den meisten Fällen sind diese langfristig ausgerichtet und wirken sich auf den Unternehmenserfolg aus. Um Kundenbeziehungen richtig zu pflegen oder zu gestalten, wird das CRM eingesetzt um eine systematische Beziehungsgestaltung zu ermöglichen. Das CRM bietet die Möglichkeit mittels Software und Datenbanken Kunden zu klassifizieren und in Kategorien einzuteilen. Die damit verbundenen Kundendaten werden als kostbarer „Rohstoff“ angesehen, welche im Unternehmen vorhanden sind. Mit Hilfe der Kundendaten können individuelle Strategien ausgearbeitet werden, um den Kunden optimal zu bedienen. Doch in den letzten Jahren nehmen Datenmengen exponentiell zu und die Komplexität der Verarbeitung und Analyse für Unternehmen nimmt zu.

Neue Konzepte, insbesondere Big Data Konzepte und die damit verbundene Technologieentwicklung im Speicherbereich, unterstützen Unternehmen bei der Analyse ihrer Kundendaten innerhalb des CRM. Potenziale bei der Verwendung liegen auf der Hand. Ein Großteil von Daten werden unstrukturiert abgelegt. Big Data Konzepte greifen diesen Aspekt auf und arbeiten sowohl mit strukturierten als auch mit unstrukturierten Datenmengen. „Das große Problem, bei welchem uns Big Data nun unterstützen will, ist die Auswertung dieser Datenmengen mit einem vertretbaren Aufwand“[Bun12]. Konzepte des Data Minings helfen bei Analysen jeglicher Art. Allerdings ist die schnelle Verfügbarkeit von Informationen extrem wichtig, um auf schnelle Veränderungen im Kundenverhalten zu reagieren.

3 Big Data - Große Datenmengen

Im folgenden Kapitel wird eine Einführung in das Thema Big Data vorgenommen. Dabei wird zunächst der Begriff Big Data erläutert und die zeitliche Entwicklung dargestellt. Zum Abschluss des Kapitels werden die Kerntechnologien vorgestellt und anhand einer Grafik anschaulich verdeutlicht.

3.1 Begriffsbestimmung

Der Begriff Big Data steht nicht nur für riesige Datenmengen, die bei der Speicherung von Daten jeglicher Art entstehen, sondern setzt sich auch mit der Verarbeitung sowie einer Analyse der Daten auseinander. Damit einhergehend sind die drei „Vs“, „Volume“, „Variety“, „Velocity“ sowie Analytics zu nennen. In Abbildung 1 wird dieses Zusammenspiel dargestellt.

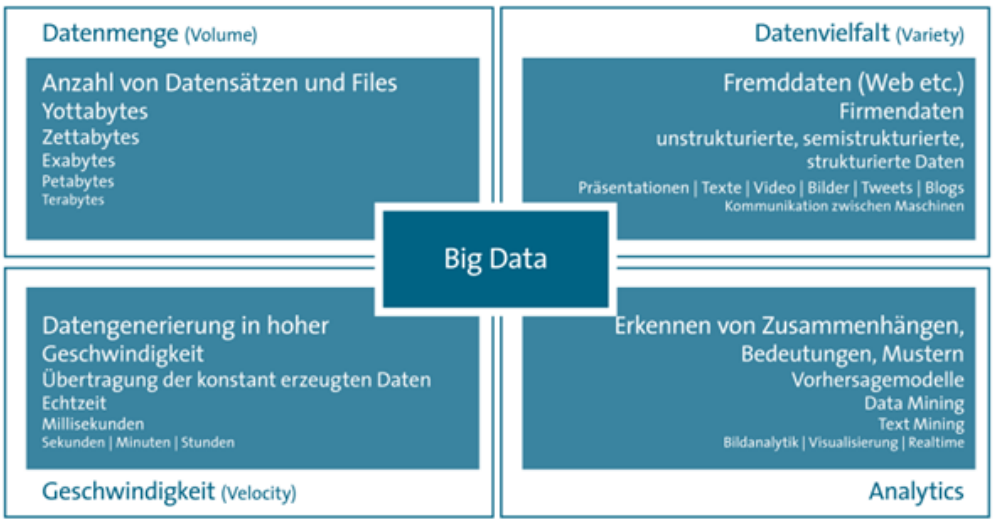


Abbildung 1: Begriffsübersicht [Bun13]

„Immer mehr Organisationen und Unternehmen verfügen über gigantische Datenberge, die von einigen Terabytes bis hin zu Größenordnungen von Petabytes führen. Unternehmen sind oft mit einer riesigen Zahl von Datensätzen, Dateien und Messdaten konfrontiert“ [Bun13] . Dies wird unter dem Begriff „Volume“ zusammengefasst.
Mit der Datenvielfalt („Variety“) wird die Art der vorliegenden Daten beschrieben. Zum einen können die Daten aus dem eigenen Unternehmen stammen, die eine strukturierte oder unstrukturierte Beschaffenheit aufweisen und zum anderen können auch Fremddaten verwendet werden. Daten könnten beispielsweise in Form von Texten, Präsentationen,

Bildern oder Blogeinträgen vorliegen. Gerade im Bereich des CRM werden Daten aus unterschiedlichen Quellen generiert und weisen verschiedenste Formate auf.
Um die Datenströme zu steuern und eine Generierung in adäquater Zeit durchführen zu können, greift der Bereich der Geschwindigkeit („Velocity“) auf Techniken zurück, um Daten möglichst in Echtzeit zu erstellen und zu analysieren. Die In Memory Technologie bietet eine gute Möglichkeit diese Anforderung zu erfüllen.
Im Bereich der Analytics müssen Zusammenhänge oder auch Muster erkannt werden, um etwaige Vorhersagemodelle entwickeln zu können. Beispielsweise werden im operativen CRM Kaufempfehlungen aus dem Kaufverhalten der Kunden mittels Analysemethoden entwickelt. Dabei ist es wichtig, dass u.a Verfahren aus dem Data Mining berücksichtigt werden.
„Zusammenfassend bezeichnet Big Data den Einsatz großer Datenmengen aus vielfältigen Quellen mit einer hohen Verarbeitungsgeschwindigkeit zur Erzeugung wirtschaftlichen Nutzens. Big Data liegt immer vor, wenn eine vorhandene Unternehmensstruktur nicht mehr in der Lage ist, diese Datenmengen und Datenarten in nötiger Zeit zu verarbeiten“[Bun13].

3.2 Entwicklung

„Big Data stellt Konzepte, Technologien und Methoden zur Verfügung, um die geradezu exponentiell steigenden Volumina vielfältiger Informationen noch besser als fundierte und zeitnahe Entscheidungsgrundlage verwenden zu können. [...] Die mit Big Data verbunden neuen Chancen entstehen nicht automatisch. Unternehmen müssen sich mit Herausforderungen auseinandersetzen, die primär mit dem Management von Daten zusammenhängen“[Bun13].

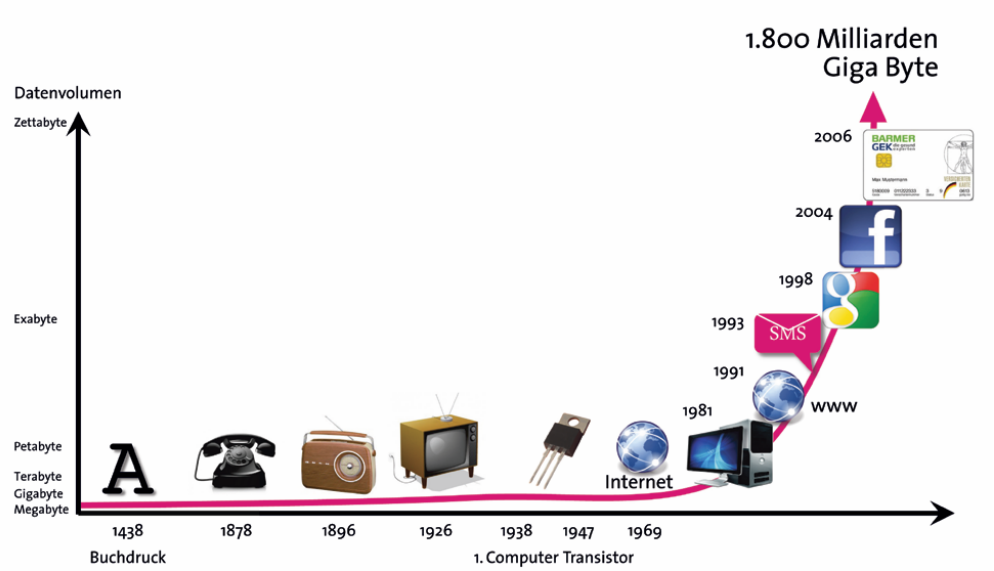


Abbildung 2: Datenwachstum [Bun13]

In der digitalen Welt werden Daten mittlerweile als vierter Produktionsfaktor angesehen. Daten können dabei aus verschiedensten Quellen z.B. „Internet Transaktionen, Social Networking, mobilen Endgeräten oder Messungen generiert werden“[Bun13].
Die exponentielle Zunahme von Daten (vgl. Abbildung 2) zwingt Experten und Forscher in bisher unbekannten Sphären zu denken und neue Verarbeitungsmethoden zu entwickeln. Durch die stetige Zunahme der Datenmenge steigt auch die Herausforderung für Unternehmen sich diesem Wandel anzuschließen. Die bisherige Bedeutung von herkömmlicher Hard- und Software als Beitrag zur Wertschöpfung nimmt ab, da das Kapital an vorhandenen Daten mit Hilfe von Analysemethoden neue Geschäftsbereiche erschließen kann.
Im Bereich des CRM lassen sich Kunden auf Basis von großen Datenmengen in ein neues Licht rücken, um die Bedeutung und Kategorisierung zu steigern. Top Kunden können somit besser bedient werden. Durch die bisherige, in den meisten Unternehmen unstrukturierte Ablage von Daten, stellt Big Data nun eine Unterstützungsmöglichkeit zur Verfügung, um die Auswertung der Datenmengen, mit einem vertretbaren Aufwand, zu ermöglichen.

3.3 Technologien

Das CRM bindet eine Vielzahl von Kundeninformationen in die Entscheidungsbasis für ein besseres Kundenverständnis ein. In Abbildung 3 ist die Architektur eines üblichen CRM Systems dargestellt. Die CRM Typen werden in Kapitel 4.4 betrachtet.

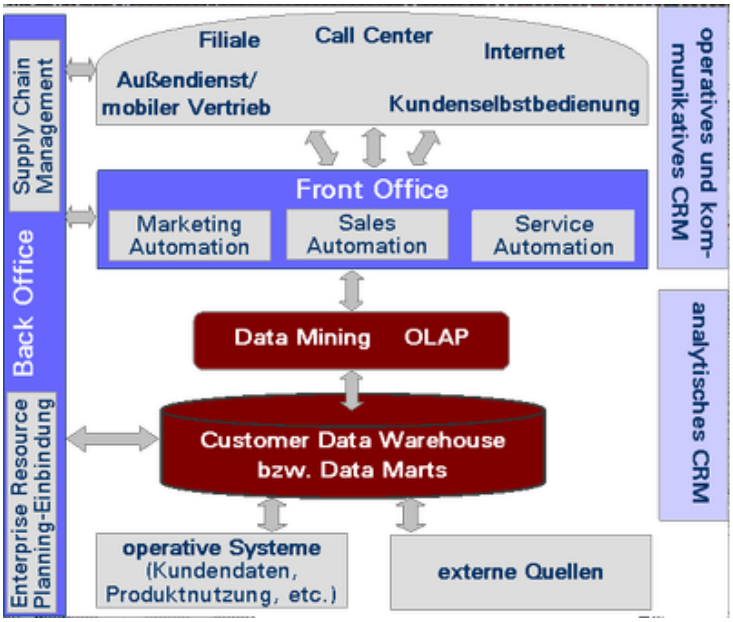


Abbildung 3: CRM-Komponenten Übersicht [CRMb]

Innerhalb des Customer Data Warehouse werden Kundendaten aus verschiedenen operativen Systemen, einfachen externen Quellen oder aus Systemen des Back Office gesammelt und gespeichert. Die Daten stehen zentral zur Verfügung und erlauben mit Hilfe von statischen Methoden, beispielsweise im Data Mining, neue Muster zu erkennen und diese dem Front Office zur Verfügung zu stellen. Innerhalb des Front Office nutzen automatisierte Software-Plattformen für die Bereiche des Marketings, Verkaufs und Services die ermittelten Muster, um effektiver Kundenkampagnen zu gestalten und den Service für den Kunden zu verbessern. Darüber hinaus können alle Unternehmensbereiche, die im Prozess der Kundeninteraktion involviert sind, auf diese Daten zugreifen.
Der Einsatz einer Big Data Anwendung erscheint sinnvoll, wenn die klassischen Technologien, wie das bereits erwähnte Data Warehouse, die Fülle an Informationen nicht mehr speichern und verarbeiten können. Mit zunehmenden Druck des Wettbewerbs vermehrt Informationen zu sammeln und auszuwerten, um einen Wettbewerbsvorteil gegenüber Mitkonkurrenten zu erzielen, ist der Einsatz unausweichlich. „Der Zweck jeder Big Data Lösung ist es, Daten in entscheidungsrelevante Informationen umzuwandeln. Die Vielfalt an Datentypen und Big Data Einsatz Szenarien erfordert auch vielfältige Werkzeuge auf jeder Schicht einer Technologie-Landschaft“ [Bun12]. In Abbildung 4 wird ein Technologie-Baukasten vorgestellt, der wesentliche Technologien für Big Data Anwendungen bereit stellt.

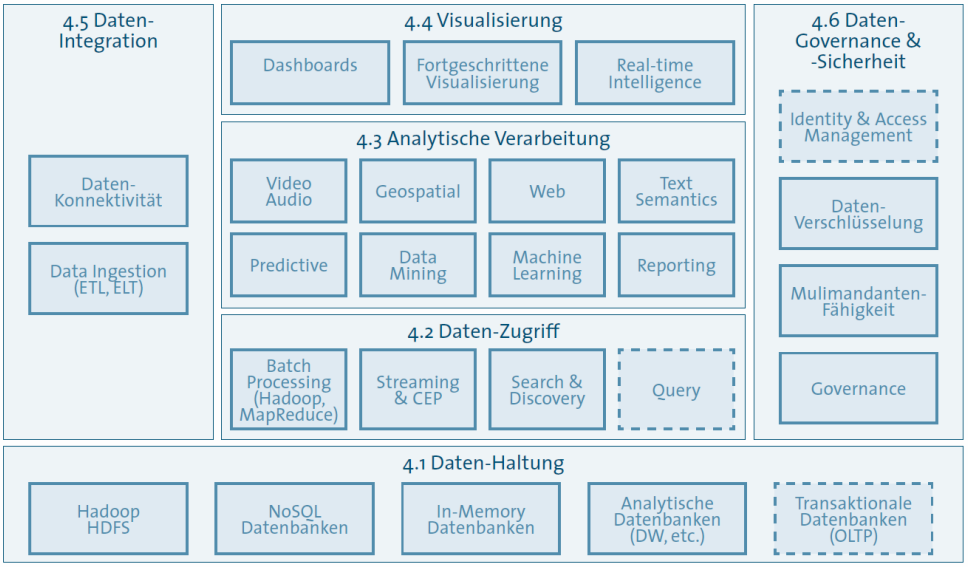


Abbildung 4: Technologieübersicht [Bun12]

Der Baukasten ist in insgesamt 6 Schichten aufgeteilt. Die Daten Haltung (4.1), der Daten Zugriff (4.2), die analytische Verarbeitung (4.4) und die Visualisierung (4.4) zeigen einen möglichen Weg der Verarbeitung von Rohdaten bis hin zu Erkenntnissen, die im weiteren Verlauf wichtig für das Vorgehen sind. Die Daten-Integration (4.5) und die Daten-Governance und -Sicherheit (4.6) dienen als Eingliederung der Daten in existierende Standards und bereits vorhandener existierender Technologien eines Unternehmens. Die Eingliederung innerhalb des CRM wird somit vereinfacht, da die Vielfalt von Technologien viele unterschiedliche Szenarien abdeckt. Mit Hilfe von ETL Prozessen, können vorhandene Kundendaten aus dem bisherigen Data Warehouse an die neue Big Data Anwendung angebunden werden und mit weiteren umfangreicheren externen Quellen erweitert werden. Für das CRM ist eine Datenbank mit Echtzeitaspekt ratsam, da Informationen schnell zur Verfügung stehen müssen. Die gewonnenen Erkenntnisse aus der Analytischen Verarbeitung können vom Front Office unmittelbar verwendet werden. Anhand eines Beispiels in Kapitel 7 wird ein beispielhafter Technologieeinsatz aufgezeigt.

4 Das Customer Relationship Management

Im folgenden Kapitel wird eine Einführung in das CRM vorgenommen. Dabei wird zunächst eine Definition gegeben und die Kernaufgaben des CRM anhand eines Kundenlebenszyklus aufgezeigt. Abschließend werden Ziele des CRM sowie die unterschiedlichen Typen, analytisches, operatives, kommunikatives und kollaboratives CRM vorgestellt.

4.1 Definition

Aufgrund des vielschichtigen Einsatzes des CRM im Unternehmen ist eine einheitliche Definition mitunter schwierig zu finden. Experten interpretieren den Begriff unterschiedlich, sodass in der Literatur keine all umfassende Definition vorliegt. Harald Löbig beschriebt das CRM wie folgt:

„CRM ist ein ganzheitlicher Ansatz zur Unternehmensführung. Es integriert und optimiert abteilungsübergreifend alle kundenbezogene Prozesse in Marketing, Vertrieb, Kundendienst sowie Forschung und Entwicklung. Dies geschieht auf der Grundlage einer Datenbank mit einer entsprechenden Software zur Marktbearbeitung und anhand eines vorher definierten Verkaufsprozesses. Zielsetzung von CRM ist dabei die Schaffung von Mehrwerten auf Kunden- und Lieferantenseite im Rahmen von Geschäftsbeziehungen“[Har09].

Somit steht die Kundenorientierung im Vordergrund. Geschäftsprozesse werden so ausgerichtet, dass sie dem Nutzen des Kunden optimal dienen. „Alle betroffenen Abteilungen nutzen eine einheitliche Datensammlung, um das Kundenmanagement im Unternehmen und zum Kunden abteilungsübergreifend zu optimieren“[Har09]. Dabei werden häufig neue Technologien eingesetzt, die eine Gewinnung von Daten ermöglichen und neue Bedürfnisse der Kunden bestimmen. „Die gewonnenen Informationen können letztendlich zur gezielten Kundenansprache verwendet werden“[CRMb].

4.2 Aufgaben

Die Hauptaufgaben des CRM lassen sich anhand des Kundenlebenszyklus nach Stauss [Sta00] darstellen. Dabei werden die „Aufgaben in drei Klassen dem Interessentenmanagement, Kundenbindungsmanagement und Rückgewinnungsmanagement unterteilt“[CRMb].

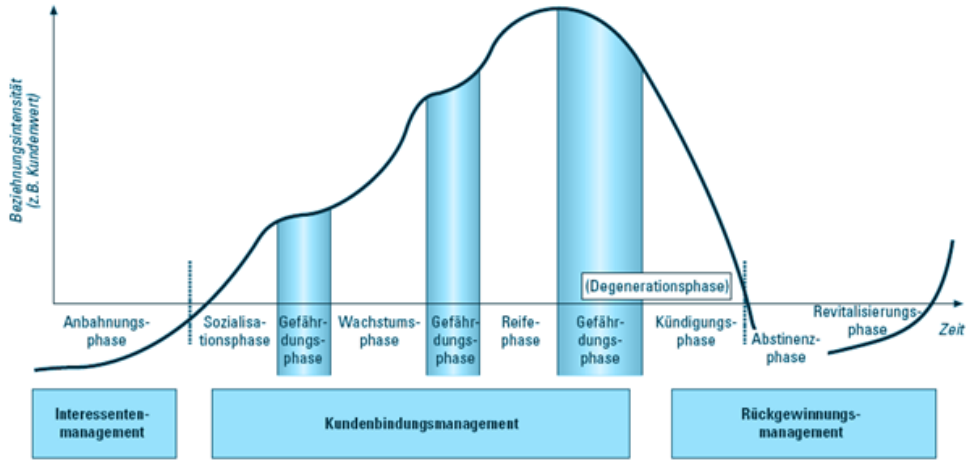


Abbildung 5: Kundenlebenszyklus [Sta00]

In der ersten Klasse, dem Interessensmanagement, befindet sich der Lebenszyklus in der Anbahnungsphase, welche sich mit der Gewinnung von Neukunden sowie mit der Ansprache von bisherigen Kunden durch neue Produkte beschäftigt. Dabei ist jedoch wichtig zu erwähnen, dass noch kein Kauf stattfindet. Der Kunde beschäftigt sich lediglich mit Produkten des Unternehmens und deren Vorteilen.[Sta00]
Die zweite Klasse beschäftigt sich mit dem Kundenbindungsmanagement. In unterschiedlichen Phasen lässt sich der Kundenwert darstellen. Der Wert eines Kunden wird dabei ermittelt um „die Betreuungskosten auf die richtigen, profitablen Kunden zu verteilen um daraus langfristig Gewinn zu erzielen“ [wer]. Für die Ermittlung werden verschiedenen Methoden wie die ABC-Kundenanalyse verwendet. Zunächst tätigt der Kunde in der Sozialisationsphase einen Kauf. Je nach Erfolg der Transaktion erfolgen weitere Käufe und der Kunde tritt in die Wachstumsphase ein. Hierbei generiert der Kunde einen gesteigerten Umsatz für das Unternehmen. Sind die Bedürfnisse des Kunden gestillt oder es kommen keine neuen Produkte auf den Markt die dem Kunden zusagen, sinken die Umsätze und der Kunde befindet sich in der Reifephase. Interveniert das Unternehmen in diesem Fall nicht, droht der Absprung des Kunden bzw. die Kündigung in der Kündigungsphase.[Sta00]
Der Übergang in das Rückgewinnungsmanagement erfolgt in diesem Fall fließend. Bereits in der Kündigungsphase arbeitet das Unternehmen an möglichen Lösungen, um den Kunden im Unternehmen zu halten. Ist dieser jedoch bereits abgesprungen, werden in der Revitalisierungsphase mögliche Schritte unternommen, um den Kunden zurückzugewinnen. [Sta00]

4.3 Ziele

Das Hauptziel des CRM ist die Integration der Kommunikations-, Distributions- und Angebotspolitik des Unternehmens sowie der Gewinnung und dem Erhalt von profitablen Kunden. Dadurch sollen vor allem Umsätze gesteigert werden und Kosten auf Seiten des Unternehmens für falsche Produktofferten vermieden werden. Mittels Analyseverfahren können Kunden in Zielgruppen segmentiert werden und besonders ertragsfähige Kundengruppen ausfindig gemacht werden.
Ein weiteres Ziel ist die langfristige Bindung des Kunden, also ein möglichst langes herauszögern der Reifephase innerhalb des Kundenlebenszyklus. Um dies zu unterstützen, zielt das CRM auf die Personalisierung von Verkaufsprozessen ab, sodass der Kunde ein individuelles Gefühl entwickelt, für das Unternehmen besonders zu sein. Unterstützend dazu, können zwischen Produkten Korrelationen erkannt werden und eine Optimierung von Angebotsverhaltensweisen vorgenommen werden. Zudem zielt das CRM auf die Optimierung von Marketing-Maßnahmen ab, sodass möglichst früh ein Trendwechsel oder eine Verhaltensänderungen im Kundenverhalten erkannt werden kann.
Die Analyse von Ausreißern ist insofern spannend, dass zusätzliche Informationen beispielsweise in Zyklen generiert werden können, um Hochphasen von Produkten zu ermitteln, die durch ein exzessives Kaufverhalten seitens des Kunden hervorgerufen werden. Um abschließend nochmal den Kundenlebenszyklus aufzurufen, zielt das CRM darauf ab, Kündigungen möglichst zu vermeiden und den Kunden im Unternehmen zu binden. [Rig]

4.4 CRM Typen

4.4.1 Analytisches CRM

Der Bereich des analytischen CRM beschäftigt sich mit der „Auswertung der im operativen Geschäft gewonnenen Kundendaten“[CRMa]. Für eine umfangreiche und genaue Analyse müssen alle zur Verfügung stehenden Daten über Kunden, Transaktionen und Produkte betrachtet und gespeichert werden. Dadurch können Verhaltensweisen der Kunden dahingehend analysiert werden, um Wünsche oder Rezensionen optimalerweise anzupassen und innerhalb eines für alle Unternehmensbereiche zugänglichem Kundenprofils zu speichern. Neben den Standardkundendaten wie Name oder beispielsweise Wohnort, können auch soziale Daten wichtig sein. Ein spezielles Hobby könnte wiederum den eigentlichen Interessensbereich des Kunden bestimmen oder vorhandene erweitern.
Im Rahmen der Datenaufbereitung sind Datawarehouse-Systeme vgl. Abbildung 3 als technische Grundlage zu nennen. Hier werden die gesammelten Daten gespeichert sowie integriert. Mit Hilfe von Data Mining, OLAP, Business Intelligence oder Data Know-

ledge Management Komponenten können Auswertungen durchgeführt werden und bisher unbekannte Datenverbindungen ermittelt werden. Somit ist eine „fortlaufende Optimierung operativer CRM Prozesse“[CRMa] durch den Einsatz das Hauptziel des analytischen CRM.

4.4.2 Operatives CRM

„Das operative CRM umfasst alle CRM-Funktionalitäten, die den direkten Kundenkontakt unterstützten“ [Gab]. Das Ziel „besteht darin, Kontakte zu potentiellen Kunden anzubahnen, diese in eine stabile (und ökonomische relevante) Kundenbeziehung zu überführen und die Beziehung zu den alten und neuen Bestandskunden zu pflegen“ [CRMd]. Dadurch können Dialoge mit dem Kunden initiiert werden und Geschäftsprozesse in verschiedenen Unternehmensbereichen, wie zum Beispiel Vertrieb oder Marketing, verbessert werden. Um Kompatibilitätsprobleme mit Systemen aus anderen Unternehmensbereichen zu vermeiden, müssen „leistungsfähige Schnittstellen an vorhandene Back Office Lösungen, wie ERP-Systeme“ [Gab] verwendet werden.

4.4.3 Kommunikatives CRM

„Das kommunikative CRM als Teilbereich des operativen CRM umfasst das integrierte Multi-Channel-Management aller Kommunikationskanäle zwischen dem Unternehmen und seinen realen oder potentiellen Kunden sowie die Gewährleistung bidirektionaler Kommunikationsprozesse, welche den aktiven und themenspezifischen Kontakt des Kunden zum Unternehmen unterstützen“ [CRMc]. Mit Hilfe des Multi-Channel-Managements werden alle vorhandenen Kommunikationsinteraktionen zwischen dem Unternehmen und dem Kunden gesteuert und hinsichtlich ihrer Effektivität und Effizienz beurteilt. Dadurch können wichtige Daten über die Erreichbarkeit des Unternehmens generiert und die Kundenzufriedenheit bestimmt werden. In diesem Zusammenhang ist das CIC zu erwähnen, welches als allgemeine Weiterentwicklung des Callcenters gesehen wird. Auf dieser zentralen Plattform laufen verschiedene Kommunikationsdienste wie zum Beispiel dem Telefonie oder E-Mail Verkehr zusammen und werden im Kontaktzentrum bearbeitet.

4.4.4 Kollaboratives CRM

Aufbauend auf dem Grundgedanken des operativen CRM zur Verbesserung von Geschäftsprozessen, versucht das kollaborative CRM die interne Zusammenarbeit aller Organisationseinheiten sowie die Kooperation mit Unternehmens fremden Partnern, unter anderem

auch Kunden, zu verbessern. Dabei kann zum Beispiel die „effektive und effiziente Kundenberatung im Service der Warenversendung durch externe Speditionen begutachtet und optimiert werden“[CRMe]. Entlang der Wertschöpfungskette können so Prozesskosten gesenkt und die Geschwindigkeit der Prozesse erhöht werden. „Das kollaborative CRM versucht neue Wege für Industrie und Handel, gemeinsam entlang der Wertschöpfungskette Kundengewinnung, Kundenbindung und Kundenentwicklung über das reine Warengruppenmanagement hinaus zu betreiben“[AQD02].

5 Potenziale beim Einsatz von Big Data im CRM

Der Einsatz von Big Data Konzepten lässt verschiedenste Bereiche eines Unternehmens profitieren. Dazu zählt unter anderem auch der Bereich des CRM, in der die Kundenbeziehung einen besonderen Stellenwert darstellt. Konkret soll es möglich sein, die im Kapitel 4.3 genannten Ziele des CRM durch Big Data zu unterstützten.

Big Data bietet die Möglichkeit der Prozessoptimierung und der Effizienzausweitung bestehender Geschäfts-, Produktions- und Entwicklungsprozesse. Dadurch ist es möglich, völlig neuartige Geschäftsmodelle zu entwickeln oder bereits bestehende Angebote zu verbessern. Zudem steigt der Informationsgewinn durch die Einführung eines neuen Geschäftsmodells weiter an, da eine Ausrichtung dahingehend vorgenommen wird.

Durch die neu geschaffene und schnell zur Verfügung gestellte Informationsbasis, können Kundenprofile hinsichtlich ihrer Genauigkeit gestaltet werden. Durch die in Abbildung 4 vorgestellten Konzepte zur analytischen Verarbeitung ist es möglich, Informationen für die mit dem Front Office verbundenen Unternehmensbereiche über Kunden bereitzustellen. „Dem Unternehmen ist es fortan möglich, Kundensegmente mit größerer Granularität zu beobachten und sowohl Produkt- und Serviceangebot besser auf den tatsächlichen Bedarf auszurichten“ [SUS]. In diesem Zusammenhang kann der „Time-to-Market“ beschrieben werden, der das frühzeitige Erkennen einer Marktveränderung auf der Grundlage der größeren Granularität ermöglicht. „Somit wird das Reagieren auf Marktänderungen zunehmend durch das Agieren ersetzt“ [Bun13].

Analysekonzepte im Big Data lässt ein gezielteres Verständnis komplexer Informationen zu und „verbessert die Qualität und Geschwindigkeit der unternehmerischen Entscheidung“ [Bun13]. Besonders Data Mining Techniken, wie die Klassifikation, Segmentierung oder Prognose und der Einsatz der neuen In Memory Technologie, lässt eine schnelle Ana-

lyse zu und stellt strategisch wichtige Informationen oftmals in Echtzeit zur Verfügung.

6 Anwendungsszenarien

„Big Data – neue Technologien bieten die Möglichkeit, Kunden auf Basis großer Datenmengen auf eine ganz neue Art kennen zu lernen – entscheidend dabei ist der richtige Anwendungsfall“[Fab14]. Das gesteigerte Interesse an Daten jeglicher Art dient vor allem dem CRM Bereich als Sprungbrett, um neue Geschäftsbereiche zu erkunden und das Unternehmen in neue Sphären zu führen. Daten können dabei durch die neuen Big Data Konzepte so umgewandelt und analysiert werden, dass sie zu einer Ertragssteigerung in vielen Bereichen des Unternehmens beitragen. Besonders wichtig ist jedoch, dass ein Unternehmen diesen Mehrwert erkennt. Statistische Analysen zeigen, dass oftmals Daten, die bereits generiert wurden, nicht genutzt werden. Die Umsetzung der Big Data Strategie ist somit die erste Herausforderung, die ein Unternehmen zu überstehen hat.[Bun12]

Im Folgenden werden 3 Anwendungsszenarien vorgestellt, bei dem der Einsatz von Big Data Konzepten und CRM in einem Unternehmen möglich und erfolgt ist.

1. **Kommunikation - Callcenter**
   Unternehmen, die Produkte vertreiben, bieten oftmals für Kunden die Möglichkeit der Kontaktaufnahme mittels eines herkömmlichen Callcenters. Hier werden in den häufigsten Fällen Produkte verkauft, reklamiert oder allgemeine Informationen vermittelt. Es werden somit wertvolle Fakten generiert, die beispielsweise Kunden davon abhalten können, dass Unternehmen zu verlassen. Ein sogenannter Call Center Agent interagiert per Telefon mit dem Kunden und versucht Kundenwünsche zu erfüllen. „Dabei bleibt für eine analytische Fragestellung keine Zeit“ [Fab14]. Durch das sogenannte Mitschneiden von Gesprächen erfolgt nach einer Einverständniserklärung des Kunden eine analytische Verarbeitung. „Beispielsweise durch die Analyse und Verschlagwortung legitimiert aufgezeichneter Gespräche“[Fab14] können Informationen und somit Daten beschafft werden. Mit der Weiterentwicklung des herkömmlichen Callcenters zu einem CIC und der Anbindung weiterer Kommunikationskanäle aus dem Unternehmen ist es möglich, die im Rahmen des Telefonats gesammelten Erkenntnisse schnell zu verarbeiten und zur Verfügung zu stellen. Dabei wird das Profil des Kunden laufend optimiert, sodass der Call Center Agent immer die aktuellsten Fakten abrufen kann. Für den Bereich des Big Data, sind unter anderem die Merk-

male Velocity und Volume betroffen, da die Menge an Daten stetig zunimmt und schnell verarbeitet wird. Die Variety wird im CIC ebenfalls angesprochen, da hier auch unterschiedliche oftmals unstrukturierte Datenströme zusammentreffen, welche mehrere Formate aufweisen.

2. **Soziale Netzwerke - Social CRM**
   Durch die steigende Bedeutung des Internets für Unternehmen und der damit verbunden Repräsentation im Netz, können vor allem Analysen in sozialen Netzwerken sinnvoll erscheinen. Laut einer Studie [Nut] nutzten alleine Im Jahr 2014 monatlich 1,35 Milliarden Menschen Facebook und generierten dabei mehrere Milliarden Datenpakete pro Sekunde. Die einfachen CRM Konzepte, wie sie in Abbildung 3 dargestellt sind, reichen nicht aus, um die Verarbeitung und Analyse dieser Datenmengen zu ermöglichen. Deshalb erscheint der Einsatz einer Big Data Anwendung sinnvoll, da hier vor allem die Stärken aus dem Volume Bereich genutzt werden können. Für Unternehmen besteht „die Möglichkeit, „Likes“ und Kommentare zum eigenen Unternehmen oder Produkten in sozialen Netzwerken mit Kundendaten zu verbinden und diese ins CRM zu transportieren, um sie dort zu analysieren“[Fab14].

3. **Kreditkartenhersteller**
   Weltweit gibt es mehrere Milliarden Kreditkarten. Alleine das Unternehmen Visa meldet weltweit den Umlauf von zwei Milliarden Kreditkarten, mit einem Transaktionsvolumen von 14 Milliarden Us-Dollar. Sekündlich werden dabei 24.000 Finanz-Transaktions-Anfragen gestellt und verarbeitet. [Vis] Eine geeignete Verarbeitung der Informationen ist durch den Einsatz einer Big Data Anwendung gewährleistet. Besonders die gestellten Anforderungen an einer schnellen und geeigneten Datenverarbeitung können durch die Merkmale des Velocity und Volume abgedeckt werden. Hersteller von Kreditkarten versuchen mit Hilfe der neuen In-Memory Technologie die Fülle an Transaktionsdaten von scheidenden Kunden schnellst möglich zu analysieren. „So wurde analytisch aus einer sehr großen Masse an Kundentransaktionen ermittelt, dass diese Kunden plötzlich vermehrt Einrichtungsgegenstände, wie Möbel, Handtücher und Bettwäsche kaufen sowie vermehrt auf Diätprodukte setzen und mehr Alkohol trinken“[Fab14].

Dies sind einige Anwendungsszenarien, die durch das Zusammenspiel von Big Data Konzepten und CRM realisiert werden konnten. Ein wichtiger Aspekt wurde jedoch noch nicht betrachtet. Der Datenschutz spielt gerade im Bereich von Kundendaten eine immens wichtige Rolle. Die rechtlichen Rahmenbedingungen müssen klar festgelegt und Grenzen

definiert werden. „Das Recht auf informationelle Selbstbestimmung verleiht dem Einzelnen die Befugnis, grundsätzlich selbst zu bestimmen, wann und im welchem Umfang er persönliche Lebenssachverhalte preisgeben möchte. Im allgemeinen Persönlichkeitsrecht (vgl. Art.2 Abs.1 GG i.V.m Art.1 Abs.1 GG) ist dies als besondere Ausprägung niedergeschrieben“ [inf]. Damit einhergehend legt das BDSG im § 3 Abs. 9 fest, wie ein exakter Umgang mit personenbezogenen Daten zu erfolgen hat. Der viel genutzte Begriff „gläserner Kunde“ zeigt eine offenbar problematische Entwicklung in diesem Bereich, denn vor allem Big Data Anwendungen versuchen einen Personenbezug zu negieren um möglichst viele Daten nutzen zu können [Thi13]. Es ist wichtig, rechtliche Grauzonen durch entsprechende Gesetze zu vermeiden und die allgemeine Aufklärung gerade im Web 2.0 für Anwender im Bereich der Datenzustimmung (Einwilligung) voranzutreiben.

7 Zusammenspiel von Big Data und CRM am Beispiel

Wie bereits in Kapitel 3.3 beschrieben, bietet Big Data einen großen Baukasten verschiedenster Technologien an. Der Einsatz von Big Data in CRM ist sinnvoll, wenn die verwendete Technologiebasis beispielsweise dem Data Warehouse an ihre Grenzen stößt. In Abbildung 6 wurde eine beispielhafte Architektur einer Big Data Anwendung visualisiert.

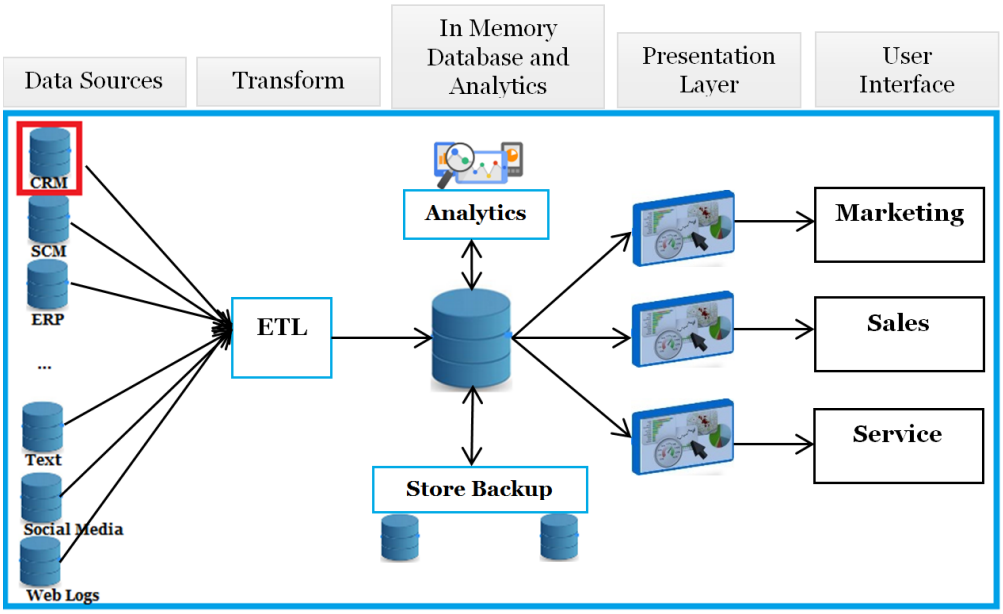


Abbildung 6: Beispiel - Architektur einer Big Data Anwendung

Dabei werden Bereiche festgelegt, die einzelne Verarbeitungsschritte vom Bereich der Rohdaten bis hin zu gemachten Erkenntnissen verdeutlichen und mit Hilfe des ETL Pro-

zesses eine Eingliederung der Daten in die Unternehmensstruktur gewährleistet. Als Datenbank wird die neue In Memory Technologie für ein OLAP System verwendet. Diese garantiert im Bereich der Analyse eine hohe und schnelle Verarbeitung von Daten und der Präsentation von Ergebnissen in Echtzeit. Im folgenden werden die einzelnen Bereiche Data Sources, Transform, In Memory Database and Analytics, Presentation Layer sowie User Interface näher beschrieben. Das Beispiel soll verdeutlichen, wie eine schnelle Analyse von Unternehmensdaten mit Personenbezug einen Mehrwert erzielen kann und die genannten Potenziale aus Kapitel 5 erreicht werden können.

7.1 Data Sources

Als Data Sources für die Big Data Anwendung dienen eine Vielzahl von unterschiedlichen aus beispielsweise OLTP basierten Systemen als Quellen. Zum einen können Unternehmensinterne Ressourcen eingebunden werden. Hierbei sind vor allem Datenbestände aus dem SCM, ERP Systemen und dem bisherigen CRM zu nennen. Die vorhandene Datenbasis kann verknüpft und in die In Memory Datenbank übertragen werden. Die nun vorhandenen Kundenprofile und Beziehungen liegen somit in der In Memory Datenbank vor und können ergänzt werden. Beispielsweise können Web Logs oder Daten aus sozialen Netzwerken verwendet werden, um das Kundenprofil aussagekräftiger zu gestalten. Soziale Daten können vor allem sinnvoll sein, um individuelle Ergebnisse zu gestalten und die Kundenbindung zu erhöhen. Das Zusammenspiel der einzelnen Big Data Merkmale „Volume“, „Variety“ und „Velocity“ ist für diesen Bereich wichtig, da große Datenmengen mit unterschiedlichen Formaten anfallen die möglichst in Echtzeit verarbeitet werden müssen.

7.2 Transform

Der Bereich der Transformation hat das Ziel, Daten aus unterschiedlichen Quellen in eine Big Data Anwendung zu importieren. Rohdaten müssen normalisiert, validiert und mit einer Struktur versehen werden, die im nachfolgenden Datenbankschema der In Memory Datenbank verwendet wird. Die Umwandlung der Daten erfolgt dabei in drei Schritten: Extract, Transform, Load. Um die In Memory Datenbank (OLAP System) zu befüllen, werden die bereits angesprochenen Data Sources (u.a OLTP Systeme) verwendet und transformiert. Dieser Schritt ist wichtig, um eine einheitliche Sicht über das Quellsystem zu erhalten, Berechnungen und Datenbereinigungen durchzuführen und eine schlussendliche Überführung in das Zielformat zu gewährleisten. Durch die Verwendung der In Memory Technologie fällt die klassische Trennung zwischen OLTP und OLAP Systemen weg. Dadruch vereinfacht sich der Aufwand beim Erstellen eines ETL Prozesses enorm und der

Echtzeitaspekt von Analysen kann gewährleistet werden. Konkret wird dadurch der viel zitierte Flaschenhals beim Laden vermieden und Daten die im Anschluss des Load Prozesses im OLAP System vorliegen bleiben aktuell. Auf der Grundlage dieser Daten können nun Analysen durchgeführt und in weiteren Anwendungen verwendet werden. Für das Beispiel ist es wichtig, Kundendaten innerhalb des ETL Prozesses so zu strukturieren, dass Folgedaten optimal zugewiesen werden und das Zielformat der Datenbank aufweisen. Zudem erweitert sich das Datenvolumen kontinuierlich. Um Redundanzen zu vermeiden, ist während des Transformationsprozesses die Datenreinigung wichtig. Durch ein sogenanntes Tagging können zusätzliche Metadaten aus anderen Sources für die Anreicherung von Informationen verwendet werden. Mit dem abschließenden Laden der Kundendaten in die Datenbank stehen dem einsetzenden Unternehmen fortan umfangreichere Kundenprofile zur Verfügung und die Reaktionsfähigkeit im Bereich von individuellen Kundenangeboten steigt.

7.3 In Memory Database and Analytics

Für das Architekturbeispiel wird eine In Memory Datenbank eingesetzt. „Die In Memory Technologie macht sich die vielfach besseren Zugriffszeiten auf den Arbeitsspeicher zu Nutze. Statt die Daten auf Festplatten zu speichern, wird der Arbeitsspeicher das Medium der Datenhaltung. Die Antwortzeiten für Datenbefragung speziell im Hinblick auf Big Data lassen sich signifikant verkürzen. Insbesondere aus Sicht eines effizienten Reportings ist die Problematik Performance ein lösbares Thema“[RP14]. In Memory Datenbanken wie beispielsweise SAP HANA ermöglichen sowohl die zeilen- als auch spaltenorientierte Datenhaltung. Die folgende Abbildung 7 zeigt die Ablagereihenfolge einer gewöhnlichen Tabelle.

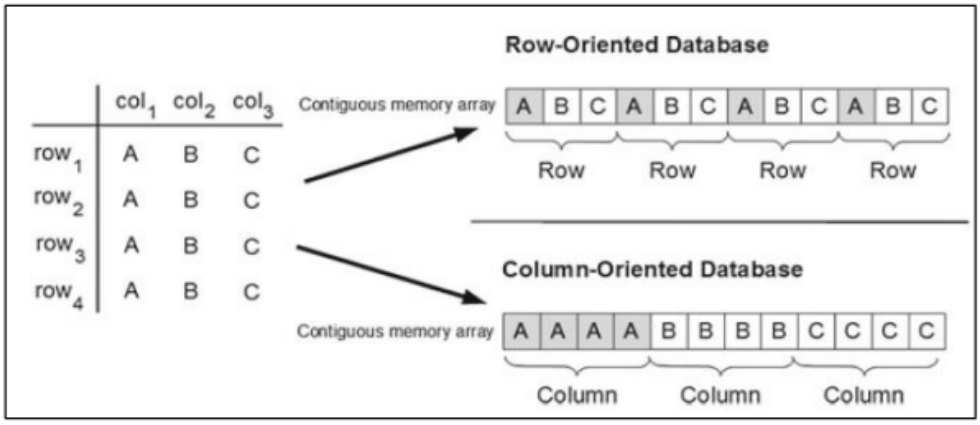


Abbildung 7: Zeilen- und spaltenorientierte Datenhaltung [bi2]

Die spaltenbasierte Arbeitsweise ist besonders für die Erstellung von einfachen Umsatz- oder Absatzberechnungen innerhalb von Reports sinnvoll. So kann beispielsweise der Bereich Sales genaue Umsatzzahlen einsehen. Die zeilenorientierte Arbeitsweise hingegen bietet Vorteile bei der Durchführung von verschiedensten Prozessen oder einer Folge von Schritten. Hierbei ist die Aktualisierung von Kundenstammdaten zu nennen welche in allen Bereichen in der neuesten Form vorliegen sollten. [RP14]
Ein weiterer wichtiger Aspekt ist die Sicherung von Daten. Im Architekturbeispiel ist an dieser Stelle ein Storage für Backup Daten vorhanden, der die in der Datenbank vorliegenden Daten in regelmäßigen Zeitabständen sichert.
Durch den kombinierten Einsatz der In Memory Technologie mit In Memory Analytics wird eine maximale End-to-End Beschleunigung „von der real-time Datenverfügbarkeit [...] über die analytische Anwendung bis hin zur Nutzung der analytischen Ergebnisse [...] in den produktiven Geschäftsanwendungen“ [Ana] (vgl. User Interfaces) erreicht.

7.4 Presentation Layer

Der Presentation Layer beeinhaltet die Präsentationslogik für erstellte Informationen und Analysen aus dem Bereich der Analytics der In Memory Datenbank. Zudem wird hier die Kommunikation mit dem jeweiligen Clientprogramm oder der grafischen Benutzeroberfläche des Users garantiert. Innerhalb des Architekturbeispiels wird die Kommunikation mit den Bereichen Marketing, Sales und Service ermöglicht.

7.5 User Interface

Das User Interface bietet eine Zugriffsmöglichkeit auf Berichte, Prognosen o.ä. für die Bereiche des Marketings, Sales und Services. Über den Presentation Layer wird die Kommunikation mit dem Analytics Bereich ermöglicht, sodass die bereits angemerkten Bereiche optimal versorgt werden können. Als User Interface kann zum einen, eine individuell gestaltete, auf einem Rollenkonzept basierte, grafische Benutzeroberfläche verwendet werden. Mit Hilfe des Rollenkonzeptes wird ermöglicht, dass u.a. der Marketingbereich nur relevanten Berichte oder Prognosen einsehen kann. Als weiterer Zugriffspunkt könnte ein handelsüblicher Browser fungieren, der die Inhalte der grafischen Benutzeroberfläche als HTML darstellt. Denkbar ist hierbei die Verwendung des bekannten Portalkonzepts wie es bereits von SAP eingesetzt wird. Der Mehrwert für Unternehmensbereiche die im engen Kundenkontakt stehen liegen auf der Hand. Durch die optimale Informationsversorgung beispielsweise des Sales Bereiches, können individuelle Kundenangebote basierend auf dem bisherigen Kaufverhalten eines Kunden erstellt werden.

8 Ausblick

Innerhalb dieser Seminarausarbeitung wurden Potenziale für den Einsatz des Trendthemas Big Data im Bereich des CRM betrachtet. Dazu konnten neben erarbeiteten Grundlagen Potenziale genannt und mehrere Anwendungsszenarien beschrieben werden, in dem Big Data und CRM bereits miteinander agieren. Zum Abschluss wurde mit Hilfe eines Beispiels eine Architektur vorgestellt, um den Einsatz einer Big Data Anwendung zu verdeutlichen.

Zukünftig erfordert die Auswertung von nichttextuellen Informationen wie etwa Bild- oder Videodaten bessere Algorithmen um Eckdaten innerhalb von Bildern zu analysieren und zuzuweisen. Damit einhergehend ist die Gesichts- und Spracherkennung ein weiterer Aspekt, der innerhalb des Web 2.0 an Bedeutung gewinnt und das Kundenprofil im Bereich des CRM detaillierter gestaltet. Der Kunde bekommt ein Gesicht sodass seine persönliche Bindung zum Unternehmen steigt. Im Hinblick auf die Ausweitung des Informationsgrades und der Informationsfülle werden vor allem die Big Data Bereiche Volume und Variety hinsichtlich des zusätzlichen Speicherbedarfs und der Zusammenführung neuer Daten gefordert. Weiter an Relevanz gewinnen zudem die Standortinformationen eines Kunden sowie seine Produktnutzungsdaten. Die Generierung von Bewegungsprofilen ermöglicht es Unternehmen Kundenvorlieben zu bestimmen und beispielsweise Stoßzeiten zu ermitteln in dem das Kaufverhalten steigt. Es ist denkbar dem Kunden bereits beim Betreten eines Supermarktes mit entsprechenden Angeboten zu versorgen. Besonders hier ist das Big Data Konzept der Velocity gefordert. Damit verbunden kann eine Analyse der Produktnutzungsdaten erfolgen, um die Nutzungsweise von Bestandteile zu verbessern oder attraktiver zu gestalten.

Im Rahmen der Projektgruppe RAPID wird der Big Data Gedanke allgegenwärtig sein. Eine Vielzahl von beispielsweise Sensorendaten, Wetterdaten oder auch Geodaten werden in strukturierter und unstrukturierter Form vorliegen. Besondere Anforderungen hinsichtlich der Speicherung, Geschwindigkeit, Integration und Verarbeitung dieser Daten sind unabdingbar. Die vorgestellte Beispielarchitektur kann im Rahmen des Erstellungsprozesses einer intelligenten Plattform für Mobilitätsdaten als Referenzarchitektur angesehen werden und mit weiteren Bestandteilen aus dem Technologiebaukasten aus Abbildung 4 ausgeschmückt und erweitert werden. Angedacht ist ebenfalls die Verwendung einer In Memory Datenbank und deren gesteigerte Analysemöglichkeiten. So ist aus dem Bereich der analytischen Verarbeitung der Einsatz der Geospatial und Predictive Analysis sinnvoll um Vorhersagen zu erstellen und diese mit GPS Punkten zu verbinden.

Literatur

[Ana] Morgen, Christoph (Hrsg.): *In Memory Analytics2 mit SAP Hana und SAS High.Performance Analytics.* http://ksfe-ev.de/2015/wp-content/uploads/2015/01/In-Memory-Analytics2-mit-SAP-HANA-und-SAS-High.Performance-Analytics.pdf, Abruf: 01.03.2015

[AQD02] Alexander, Kracklauer (Hrsg.) ; Quinn, Mills (Hrsg.) ; Dirk, Seifert (Hrsg.): *Kooperatives Kundenmanagement. Wertschöpfungspartnerschaften als Basis erfolgreicher Kundenbindung.*. Bd. 1. 2002

[bi2] Andreas, Solsbach (Hrsg.): *Vorlesung Business Intelligence 2 - In-Memory Technologien und SAP HANA.* https://elearning.uni-oldenburg.de/sendfile.php?type=0&file_id=7645ac5481e3b8363f52c5d482906e70&file_name=In-Memory_Computing_und_SAP_HANA.pdf, Abruf: 01.03.2015

[Bun12] Bundesverband Informationswirtschaft, Telekommunikation und neue Medien e. V. (Hrsg.): *Big-Data-Technologien- Wissen für Entscheider Leitfaden.* Bd. 1. 2012

[Bun13] Bundesverband Informationswirtschaft Telekommunikation und neue Medien e. V. (Hrsg.): *Big Data im Praxiseinsatz – Szenarien, Beispiele, Effekte.* Bd. 1. 2013

[CRMa] CRM Centrum für Reisemedizin GmbH (Hrsg.): *Analytisches CRM.* http://www.crm.de/crm/analytisches-crm.html, Abruf: 01.03.2015

[CRMb] Oldenbourg Wissenschaftsverlag (Hrsg.): *Customer Relationship Management (CRM).* http://www.enzyklopaedie-der-wirtschaftsinformatik.de/wi-enzyklopaedie/lexikon/informationssysteme/crm-scm-und-electronic-business/Customer-Relationship-Management, Abruf: 01.03.2015

[CRMc] CRM Centrum für Reisemedizin GmbH (Hrsg.): *Kommunikatives CRM.* http://www.crm.de/crm/kommunikatives-crm.html, Abruf: 01.03.2015

[CRMd] CRM Centrum für Reisemedizin GmbH (Hrsg.): *Operatives CRM.* http://www.crm.de/crm/operatives-crm.html, Abruf: 01.03.2015

[CRMe] Geiger, Daniel (Hrsg.): *Unterschiedliche CRM-Typen und ihre Schwerpunkte.* http://www.crm-special.info/crm-artikel/unterschiedliche-crm-typen-und-ihre-schwerpunkte, Abruf: 01.03.2015

[Fab14] Faber, Alexander: *Big Data Anwendungsbeispiele im CRM.* Bd. 1. 2014

[Gab] Gabler Wirtschaftslexikon (Hrsg.): *Customer Relationship Management(CRM).* http://wirtschaftslexikon.gabler.de/Archiv/5072/customer-relationship-management-crm-v10.html, Abruf: 01.03.2015

[Har09] Harald, Löbig ; Verlag, Grin (Hrsg.): *Einführung eines CRM-Systems - Dargestellt an einem Automobilzulieferer.* Bd. 1. 2009

[inf] Bundesministerium des Inneren (Hrsg.): *Artikel: Der Schutz des Rechts auf informationelle Selbstbestimmung.* http://www.bmi.bund.de/DE/Themen/Gesellschaft-Verfassung/Datenschutz/Informationelle-Selbstbestimmung/informationelle-selbstbestimmung_node.html, Abruf: 01.03.2015

[Nut] statista - Das Statistik Portal (Hrsg.): *Nutzerzahlen Facebook 3 Quartal 2014.* http://www.statista.com/statistik/daten/studie/37545/umfrage/anzahl-der-aktiven-nutzer-von-facebook/, Abruf: 01.03.2015

[Rig] Riggert Wolfgang Prof. Dr. ; Flensburg, FH (Hrsg.): *Customer Relationship Management.* http://www2.wi.fh-flensburg.de/wi/riggert/veranstaltungen/AKAD/4-Administrationssysteme-CRM.pdf, Abruf: 01.03.2015

[RP14] Reinhard, Bär ; Philippe, Burtschert: *Lean Reporting.* Bd. 1. 2014

[Sta00] Stauss, Bernd (Hrsg.): *Perspektivenwandel: Vom Produkt-Lebenszyklus zum Kundenbeziehungs-Lebenszyklus.* Bd. 1. 2000

[SUS] Schmitz-Urban, Arno ; Siegers, Jan ; GITO mbH Verlag für Industrielle Informationstechnik und Organisation (Hrsg.): *Evolution des CRM durch Big Data.* http://www.erp-management.de/node/181, Abruf: 01.03.2015

[Thi13] Thilo, Weichert: *Unabhängiges Landeszentrum für Datenschutz Schleswig Holstein: Big Data und Datenschutz.* Bd. 1. 2013

[Vis] Netz-Trends (Hrsg.): *Kreditkartenhersteller - Weltmarktführer Visa.* http://www.statista.com/statistik/daten/studie/37545/umfrage/anzahl-der-aktiven-nutzer-von-facebook/, Abruf: 01.03.2015

[wer] Smart CRM (Hrsg.): *Kundenwert.* http://www.smatcrm.de/definition/items/kundenwert.html, Abruf: 01.03.2015

Abschließende Erklärung

Ich versichere hiermit, dass ich meine Seminarausarbeitung im Rahmen der Projektgruppe RAPID selbständig und ohne fremde Hilfe angefertigt habe und dass ich alle von anderen Autoren wörtlich übernommenen Stellen wie auch die sich an die Gedankengänge anderer Autoren eng anlegenden Ausführungen meiner Arbeit besonders gekennzeichnet und die Quellen zitiert habe.

Oldenburg, den 15. April 2015

Christian Janßen

Datenanreicherung/-ergänzung durch Open Data Sources

Seminararbeit im Rahmen der Projektgruppe RAPID

Themensteller:	Prof. Dr.-Ing. Jorge Marx Gómez
Betreuer:	M. Sc. Alexander Sandau
Vorgelegt von:	Jannes Spekker, B.A. Artillerieweg 44 26129 Oldenburg 0151/12348006 jannes.spekker@uni-oldenburg.de
Abgabetermin:	09.03.2015

Inhaltsverzeichnis

Abbildungsverzeichnis

Abkürzungen

API Application Programming Interface

BDSG Bundesdatenschutzgesetz

CC Creative Commons

CSV Comma-separated values

DWD Deutscher Wetterdienst

ETL Extract, transform, load

EU Europäische Union

FOKUS Fraunhofer-Institut für offene Kommunikationssysteme

FTP File Transfer Protocol

GeoZG Geodatenzugangsgesetz

GML Geography Markup Language

HTML Hypertext Markup Language

IFG Informationsfreiheitsgesetz

INSPIRE Infrastructure for Spatial Information in the European Community

IWG Informationsweiterverwendungsgesetz

JSON JavaScript Object Notation

KML Keyhole Markup Language

ODS OpenDocument Spreadsheet

ODT OpenDocument Text

OKFN Open Knowledge Foundation

RAPID Regional Analysis and Prediction Platform by In-Memory Data

RDF Resource Description Framework

RSS Really Simple Syndication

SatDSiG Satellitendatensicherheitsgesetz

SVG Scalable Vector Graphics

TXT Text

UIG Umweltinformationsgesetz

UrhG Urheberrechtsgesetz

URI Uniform Resource Identifier

USP Unique Selling Point

VIG Verbraucherinformationsgesetz

XML Extensible Markup Language

1 Einleitung

Der Bedarf komplexer Analysen nimmt immer mehr zu – ob in Wirtschaft, Wissenschaft, der öffentlichen Verwaltung oder anderen Bereichen. Um diesem Bedarf gerecht zu werden müssen Daten und Informationen gesammelt, aufbereitet und korrekt verwendet werden. Bei der Sammlung nutzbarer Daten müssen jedoch einige Hindernisse beseitigt werden. Viele Daten liegen vor und können theoretisch kostenfrei aufgerufen werden, doch scheitert es häufig an der Unvollständigkeit oder der mangelnden Aktualität vorhandener Daten, sowie an fehlendem Wissen bei der Beschaffung und Verwendung auf Seiten des Nutzers. Mit der weiter steigenden Netzabdeckung, dem Reifegrad der Internet-Technologien sowie den niedrigen Speicherkosten sind die Voraussetzungen geschaffen, Unmengen an Daten bereitzustellen. Zudem ist es möglich, sämtliche anfallenden Daten zu speichern. Bestehende Hindernisse wirken den geschaffenen Voraussetzungen entgegen. Viele Interessengruppen mit unterschiedlichen Zielen sind im Besitz der Daten. Teilweise sichern Daten die Existenzen der Gruppen. Einige Daten stellen, öffentlich bereitgestellt, ein hohes Sicherheitsrisiko für Personen, Unternehmen, Städte oder Staaten dar. Neben der Technologie müssen vor allem auch politische, rechtliche und persönliche Aspekte einbezogen werden. Das Thema hat in Wissenschaft und öffentlicher Verwaltung bereits einen hohen Stellenwert. Viele Plattformen für offene Daten sind bereits jetzt aufrufbar, meistens in der Hand öffentlicher Institutionen, die eine Vielzahl an Themen abdecken und Datensätze in offener Form zum Download anbieten. Zudem umfasst die auf dem Markt verfügbare Software in den meisten Fällen Schnittstellen für den Import solcher Datensätze.
Ziel der Projektgruppe „Regional Analysis and Prediction Plattform by In-Memory Data“ an der Carl von Ossietzky Universität Oldenburg ist die Schaffung einer Umgebung zur Erfassung, Verarbeitung und Bereitstellung von Mobilitätsdaten mit regionalem Bezug. Die Anreicherung der von Verkehrsbetrieben und branchennahen Unternehmen bereitgestellten Daten durch offene Datenquellen ermöglicht die Berücksichtigung zusätzlicher Variablen und unterstützt somit die Generierung realistischer Analysen und Vorhersagen. Ziel der Arbeit ist, das Konzept von Open Data Sources zu erläutern und den Nutzen der Datenanreicherung durch offene Daten für das Projekt RAPID darzustellen. In dieser Seminararbeit wird zunächst das Konzept von Open Data aufgezeigt. Dabei werden relevante Begriffe erläutert, Voraussetzungen dargestellt und verwandte Bereiche abgegrenzt. Daraufhin werden die Potentiale beleuchtet, indem auf Kriterien und Chancen eingegangen wird. Im Anschluss daran wird ein Auswahlprozess relevanter Datensätze skizziert. Weiter werden Anbieter und Beispieldatensätze vorgestellt.

2 Open Data

2.1 Definition

Der Begriff Open Data gilt als nicht allgemeingültig definiert, allerdings wurde von der Open Knowledge Foundation (OKFN) eine Definition entwickelt, die sich inzwischen durchgesetzt hat und von vielen einflussreichen Staaten übernommen wurde.
„*Offene Daten sind Daten, die von jedermann frei verwendet, nachgenutzt und verbreitet werden können – maximal eingeschränkt durch Pflichten zur Quellennennung und 'sharealike'.*“ ([Ope15h])
Als Grundlage diente die Definition von Open Source. Die Komplettfassung der so genannten „Open Definition“ enthält 865 Wörter (Stand 15.02.2015). Aufgeführt sind dort neben der Begriffserklärung auch die Voraussetzungen, die Daten erfüllen müssen, um als offen zu gelten. Kernaussagen der Definition betreffen Verfügbarkeit und Zugang, Wiederverwendung und Nachnutzung sowie universelle Beteiligung, also die Aspekte „Access“, „Open Format“ und „Open License“. Der Abschnitt Voraussetzungen führt diese Anforderungen genauer aus. Auch auf die Pflichten zur Quellennennung sowie dem Begriff „sharealike“ wird dort eingegangen (vgl. [Die11c]).

2.2 Open Knowledge Foundation

Die Open Knowledge Foundation mit Sitz in Cambridge, Vereintes Königreich, ist eine gemeinnützige Organisation mit dem Ziel, offenes Wissen zu fördern. Als global agierende Organisation haben sich Ableger in vielen Staaten gebildet, wie etwa dem gleichnamigen eingetragenen Verein in Deutschland. Die OKFN unterstützt und initiiert zahlreiche Projekte auf nationaler und internationaler Ebene. Zudem tritt die Organisation in beratender Tätigkeit mit Regierungen in Verbindung (vgl. [Ope15c]). Ein Projekt, welches international große Beachtung gefunden hat ist die Entwicklung der Open-Source Lösung CKAN, einem frei verfügbaren System zur Erstellung von Open Data Portalen. Nutzer des Systems sind beispielsweise die amerikanische sowie die britische Regierung, deren Portale auf CKAN basieren (vgl. [Ope15a]). Ein weiteres Projekt der britischen OKFN ist die Plattform „Where Does My Money Go“, welche auf übersichtliche Weise darstellt, für welche Zwecke die Steuergelder in Großbritannien ausgegeben werden (vgl. [Ope15d]).
In Deutschland veröffentlichte Projekte sind die Seite „Offener Haushalt“, die den Haushaltsplan des Bundes darstellt und so, ähnlich der britischen Plattform, Transparenz in die Haushaltspolitik bringen soll (vgl. [Ope15f]), sowie der Wettbewerb „Apps für Deutschland“, bei dem die besten mobilen Apps prämiert wurden, die auf offenen Daten basieren

(vgl. [Ope15e, Ope15j]).

2.3 Voraussetzungen

Die Voraussetzungen für Daten, um nach der Open Definition als offene Daten deklariert werden zu können, sind dreigeteilt und unterteilen sich in die Kategorien Access, Open Format und Open License (vgl. [Ope15b]).

2.3.1 Formate

Offene Formate haben grob drei Voraussetzungen zu erfüllen. Zunächst muss ein maschinenlesbares Format verwendet werden. Dadurch wird gewährleistet, dass die bereitgestellten Daten von Anwendungen importiert und verarbeitet werden können. Außerdem ist eine Spezifikation zur Verfügung zu stellen, diese enthält die Dokumentation zu den Daten und stellt sicher, dass es sich um strukturierte Daten handelt. Darüber hinaus muss ein offenes Format verwendet werden. Es handelt sich dann um ein offenes Format, wenn der Datensatz von mindestens einer Open-Source-Software gelesen werden kann (vgl. [Ope15b]).
Daraus ergibt sich eine Liste von Formaten, die als offene Formate im Sinne der OKFN gelten. Zu nennen sind Textformate (.txt, .odt), Tabellenformate (.ods, .csv), Formate für Auszeichnungssprachen (.html, .xml, .rdf), .rss für Newsfeeds, .svg als auf XML basierendes Vektordatenformat, .json zum Datenaustausch und Formate zur Beschreibung von Geodaten (.gml, .kml) (vgl. [Gei10]).

2.3.2 Zugang

Der Aspekt Access beschreibt Voraussetzungen für den Zugang zu offenen Daten. Auch hier werden drei Punkte aufgeführt. Bereitgestellte Datensätze müssen als Ganzes zur Verfügung gestellt werden. Sie sollten idealerweise als Download und maximal zu den einmaligen Reproduktionskosten abrufbar sein (vgl. [Ope15b]).

2.3.3 Lizenzen

Der größte Bereich der Open Definition ist dem Aspekt Open License gewidmet. Daten werden deshalb mit Lizenzen versehen, um kenntlich zu machen, welche gesetzlichen Regelungen für die Nutzung beachtet werden müssen beziehungsweise wie die Daten unter Beachtung von Gesetzen genutzt werden dürfen. Diese rechtlichen Grundlagen sollen zunächst vorgestellt werden, bevor auf die einzelnen Voraussetzungen für offene Lizenzen eingegangen wird.

Besonders Daten der öffentlichen Verwaltung unterliegen gesetzlichen Bestimmungen bezüglich des Besitzanspruchs und der Veröffentlichung. Das Informationsfreiheitsgesetz (IFG) schreibt den uneingeschränkten Zugang zu amtlichen Dokumenten für alle Interessengruppen vor. Das Gesetz über die Weiterverwendung von Informationen öffentlicher Stellen (IWG) wiederum behandelt die Weiterverarbeitung von Daten der öffentlichen Verwaltung. Darüber hinaus werden Eigentumsrechte von Daten im Gesetz über Urheberrecht und verwandte Schutzrechte (UrhG) geregelt. Das Bundesdatenschutzgesetz (BDSG) und die Landesdatenschutzgesetze setzen sich mit dem Datenschutz und Persönlichkeitsrechten auseinander. Es liegen weitere Gesetze zu Daten unterschiedlicher Art vor, etwa das Umweltinformationsgesetz (UIG), das Verbraucherinformationsgesetz (VIG), das Geodatenzugangsgesetz (GeoZG) oder das Satellitendatensicherheitsgesetz (SatDSiG). Je nach Geltungsbereich gelten genannte Gesetze entweder auf Bundes-, auf Länder- oder auf kommunaler Ebene (vgl. [Die11b]).
Beispielhaft soll eine Richtlinie vorgestellt werden, die die Öffnung von Datensätzen gewährleisten soll. Geodaten spielen eine wesentliche Rolle für das Mobilitätsprojekt RAPID, da die Analyse räumlicher Daten, beziehungsweise die Betrachtung von Variablen in einem räumlichen Kontext im Vordergrund der Auswertung steht. Das europäische Parlament hat 2007 die INSPIRE Richtlinie veröffentlicht. Ziel ist die Erleichterung der grenzübergreifenden Nutzung von Daten in Europa. Umgesetzt werden soll dieses Ziel durch webbasierte Online-Dienste für die Suche, Visualisierung und den Download der Daten. Die fachlichen und technischen Einzelheiten für die Umsetzung sind in der Richtlinie jedoch nicht vorgegeben, diese müssen von den Mitgliedsstaaten selbst erarbeitet werden. Die Richtlinie bezieht sich ausschließlich auf Geodaten, wobei die einzelnen Themenfelder der Geodaten begrenzt sind. Sie sind in drei Kategorien unterteilt, die als Priorisierung für die Bereitstellung anzusehen sind. Zu den Themen zählen beispielsweise geographische Namen, Adressen, Verkehrsnetze oder Bodenbedeckungen. Die Beschreibung der Daten durch Metadaten wird vorgeschrieben. Zur Bereitstellung wird zudem vorgegeben, dass die Daten dezentral von den fachlichen Stellen bereitgestellt werden sollen, die technische Umsetzung erfolgt jedoch über ein zentral verwaltetes Netzwerk (vgl. [INS]). Aus dieser Richtlinie folgt in Deutschland das Geodatenzugangsgesetz, welches den uneingeschränkten Zugriff auf amtliche Geodaten vorgibt. Die Richtlinie enthält einen Zeitplan zur Bereitstellung sämtlicher Geodaten. Demnach muss der Prozess der Bereitstellung bis zum 15.05.2019 abgeschlossen sein, 12 Jahre nach dem Inkrafttreten der Richtlinie (vgl. [Lan12]).
Allgemein zählen Daten und Informationen zu immateriellen Gütern. Solche Güter sind im angelsächsischen Raum durch das Copyright, im kontinentaleuropäischen Raum durch das

Urheberrecht geschützt. Die dort geregelten Eigentumsrechte für Immaterialgüter werden verdeutlicht, indem die Güter mit Lizenzen versehen werden.
Für offene Lizenzen ist die OKFN maßgeblich. Demnach sind Lizenzen dann offen, wenn sie konform zur Open Definition sind. Im Abschnitt „Open License“ der Open Definition wird zwischen notwendigen und zulässigen Bedingungen unterschieden. Während die notwendigen Bedingungen zwingend einzuhalten sind und durch eine offene Lizenz niemals eingeschränkt werden dürfen, handelt es sich bei zulässigen Bedingungen um solche, die die Rechte im Umgang mit den Daten einschränken dürfen, wenn aufgrund von Gesetzen keine andere Möglichkeit besteht.
Inhalt notwendiger Bedingungen ist, dass die Daten frei genutzt und weiterverbreitet werden dürfen, auch wenn eine kommerzielle Nutzung beabsichtigt wird. Eine Modifikation der Daten kann genauso vorgenommen werden, wie die Trennung des gesamten Datensatzes in einzelne, benötigte Abschnitte. Die zulässigen Bedingungen bewirken eine Einschränkung in der Nutzung der Daten. Bereits in der Definition zu Open Data werden Pflichten zur Quellennennung und „sharealike“ genannt. Diese beiden Begrifflichkeiten sind Teil der zulässigen Bedingungen. Die Quellennennung, auch Namensnennung oder im englischen Attribution genannt, schreibt vor, den Autor beziehungsweise Bereitsteller der Daten anzugeben, wenn der Datensatz, in jeglicher Form, verwendet wird. Der Begriff „sharealike“ ist auch unter dem Begriff Copyleft bekannt. Bei Nutzung und Weiterverbreitung der Daten ist die bestehende Lizenz in gleicher Form beizubehalten. Weiter könnten zulässige Bedingungen die Integrität und die Quelloffenheit betreffen (vgl. [Ope15b]).

2.4 Potentiale

2.4.1 Chancen

An der Politik wird oft die Kritik laut, die Informationsweitergabe sei nicht ausreichend. Dieser kann durch Veröffentlichung entsprechender Informationen entgegengewirkt werden, so wird das Regierungshandeln transparent. Wichtig dabei ist jedoch, dass nicht nur der Zugang zu Daten ermöglicht wird, sondern durch die Wahl eines geeigneten Formats und die Offenheit der Daten eine Verarbeitung und Nutzung der Daten sichergestellt wird. Die kommerzielle Nutzung offener Daten führt zu Gründung innovativer Unternehmen, wodurch ein wirtschaftlicher und gesellschaftlicher Mehrwert geschaffen wird. Dies führt zur Freisetzung von Werten. Darüber hinaus fühlen sich die Bürger durch die geschaffene Transparenz in die Entscheidungsfindung einbezogen, sie haben die Möglichkeit sich am Regierungsprozess zu beteiligen, etwa durch Demonstrationen oder Petitionen, dies stärkt die Partizipation und die Zusammenarbeit (vgl. [Ope15h, Die11a, Her10]).

2.4.2 Risiken

Die Offenheit sämtlicher Daten ist weder möglich noch erwünscht, gerade sensible Daten erfordern höchste Aufmerksamkeit und können bei falscher Handhabung weitreichende Folgen haben.
Unternehmen leben vom Wettbewerb, einige Daten sorgen für einen Mehrwert für das Unternehmen, sind teilweise der USP. Um Patente und Innovationen zu schützen ist es dringend erforderlich Daten unter Verschluss zu halten. In diesem Rahmen ist auch das Wettbewerbsrecht zu nennen, welches eine Konkurrenzsituation von kommerziellem und durch Steuergelder finanziertem Angebot verbietet. Das oben genannte Datenschutzgesetz schränkt die Veröffentlichung weiter ein. Erfasste Daten über Individuen sind unter Umständen nicht hinreichend anonymisiert, wodurch von den Daten auf bestimmte Personen, Gruppen oder Unternehmen geschlossen werden kann. Ein weiterer Punkt der Nutzung ist die Gefahr des Missbrauchs offener Daten. Gelangen Daten über Sicherheitslücken, Waffen- oder Genforschung in die Hände von Personen oder Vereinigungen mit kriminellen Absichten kann dies schwerwiegende Folgen haben. Darüber hinaus besteht die Gefahr vor absichtlicher oder unabsichtlicher Verfälschung der Daten (vgl. [Ope15h, Die11a, Gru14]).

2.5 Datenkategorien

Die OKFN stellt eine Kategorisierung offener Daten auf ihrer Webseite bereit. Diese Kategorien können als Anhaltspunkt für die Suche relevanter Daten für das Projekt dienen. Geodaten als digitale Informationen über reale Objekte mit Raumbezug dienen der Erstellung von Karten, sie können Informationen über Straßen, Gebäude, Topographien oder Grenzen enthalten. Die Kategorie Kultur speichert Informationen über kulturelle Werke, aber auch Daten, die von Kultureinrichtungen wie Gallerien, Bibliotheken, Archiven oder Museen gesammelt werden. Daten aus der Wissenschaft entstehen in wissenschaftlichen Forschungen. Das Finanzwesen speichert Haushaltsdaten und Informationen zu Finanzmärkten. Statistische Ämter erheben Daten im Bereich Statistik, etwa bei Volkszählungen oder Erhebungen in sozioökonomischen Untersuchungen. Wetterdaten werden von Klimastationen oder durch Vorhersagen gesammelt, außerdem fallen sie in der Klimaforschung an. Des Weiteren werden Umweltdaten genannt, die die natürliche Umwelt beschreiben und beispielsweise Aussagen zu Gewässern, deren Qualität und Schadstoffbelastungen machen. Als letzte Kategorie werden Transportdaten beschrieben, also Informationen die den Verkehr und die Logistik betreffen. Zu nennen sind Fahrpläne und Fahrstrecken (vgl. [Ope15h]).

2.6 Verwandte Bereiche

Der Begriff Open Government Data wird häufig synonym zu Open Data verwendet. Tatsächlich handelt es sich dabei jedoch um offene Daten der öffentlichen Verwaltung und schließt somit, im Vergleich zu Open Data, die Bereiche Wissenschaft und Forschung aus. Ein weiterer Begriff ist Linked Open Data, dabei handelt es sich um ein von Tim Berners-Lee entwickeltes Konzept über ein Netzwerk von Daten. Die einzelnen Datensätze werden durch ein URI eindeutig identifiziert und adressiert, zudem verweisen sie auf andere Daten. Durch dieses Vorgehen entstand ein Netz aus ca. 25 Milliarden Daten mit mehr als 400 Millionen Verbindungen. Durch die Verbindungen können Beziehungen aufgedeckt werden, die bislang nicht betrachtet wurden (vgl. [Gei10, BL06]).
Open Content beinhaltet den Bereich Open Data, wird aber weiter gefasst. Es sind neben Daten auch Medien, also Filme, Musik und Bilder, Software und Technologien unter Open Content zusammengefasst. Ein weiterer Bereich ist Open Access, die öffentliche Bereitstellung von wissenschaftlichen Publikationen. Bekannt ist der Begriff Open Source, hierbei handelt es sich um quelloffene, kostenlos verfügbare Software. Durch Bereitstellung des Quellcodes kann die Anwendung von allen, die über die nötigen Fähigkeiten verfügen, weiterentwickelt werden (vgl. [Bä07]).

2.7 Datenbereitstellung

Es ist jedem möglich seine Daten als offene Daten der Öffentlichkeit zur Verfügung zu stellen. Hierzu müssen vier Schritte durchlaufen werden. Im ersten Schritt müssen die entsprechenden Daten ausgewählt werden. Es muss darauf geachtet werden, dass keine Gesetze durch die Veröffentlichung verletzt werden. Der zweite Schritt sieht, unter Berücksichtigung gesetzlicher Regelungen, die Vergabe einer öffentlichen Lizenz vor. Die notwendigen Bedingungen der Open Definition müssen gewährleistet werden, bei den zulässigen Bedingungen ist zu beachten, dass dem Nutzer ein möglichst großer Spielraum gelassen wird. Im dritten Schritt werden die Daten verfügbar gemacht. Es ist ein geeignetes, offenes Format zu wählen. Zuletzt muss in Schritt vier der Zugang sichergestellt werden. Denkbar ist die Bereitstellung per Download oder als API. Abschließend müssen die Daten auffindbar gemacht werden. Genutzt werden kann ein zentraler Datenkatalog, etwa das Open Data Portal des Bundes. Liegen eine Vielzahl an Datensätzen zur Veröffentlichung vor kann über ein eigenes Portal nachgedacht werden. Das bereits vorgestellte Portal CKAN der OKFN kann kostenlos heruntergeladen und auf einem Webserver installiert werden (vgl. [Ope15h]).
Darüber hinaus sollten drei Hauptregeln bei der Öffnung von Daten befolgt werden, die

kurz vorgestellt werden. Beginne klein, einfach und schnell meint, dass zunächst der Prozess der Öffnung an sich durch Öffnung kleiner Datensätze geübt werden kann, die so gewonnene Erfahrung kann bei der Bereitstellung weiterer Datensätze angewandt werden. Als zweite Empfehlung wird ausgesprochen, sich früh und oft mit anderen Nutzern in Verbindung zu setzen und so die Daten bekannt zu machen. Dieses Prinzip beruht auf der Annahme, dass die häufigste Form der Datenverbreitung die über Vermittler, also Mundpropaganda von Nutzern gegenüber weiteren Nutzern, ist. Die dritte Handlungsempfehlung ist die Beseitigung von Missverständnissen und Ängsten. Dies meint, dass gerade bei Regierungsdaten eine Hemmschwelle vor der Bereitstellung der Daten besteht, da nicht klar ist, ob ein Recht zur Veröffentlichung bestimmter Datensätze besteht. Dies wird behoben, indem sich der Datenbereitsteller mit der Fragestellung auseinandersetzt und sich über Rechte und Pflichten informiert (vgl. [Ope15h]).

3 Datenauswahl

Zur Analyse von Mobilitätsdaten und der Vorhersage von Verkehrsflüssen müssen möglichst viele verkehrsbeeinflussende Faktoren berücksichtigt werden. So wird eine möglichst realistische Berechnung ermöglicht. Die Nutzung bereits erhobener und öffentlich bereitgestellter Daten sorgt für Kosten- und Zeiteinsparungen. Damit solche Daten systematisch auffindbar gemacht werden können wird im Folgenden ein eigenständig entwickeltes Vorgehensmodell vorgestellt.

Abbildung 1: Datenauswahlprozess

Es werden fünf Phasen sequentiell durchlaufen. Zunächst stellt sich die Frage, welche Daten für das Projekt relevant sind. Anschließend muss auf Grundlage der eingesetzten Systeme analysiert werden, in welcher Form die Daten benötigt werden. Darauf hin folgt

die Datensuche. Werden potentiell relevante Daten gefunden müssen diese auf Nutzbarkeit (Form) und Validität (Lizenz) überprüft werden.

3.1 Phase 1: Relevante Daten

Bevor es zu einer wahllosen Suche nach nützlichen Daten kommt sollte das Thema genauer betrachtet werden. Die Frage lautet hier, welche Daten für das Projekt tatsächlich relevant sind. Dabei sollten nicht nur Bereiche untersucht werden, die in direktem Zusammenhang mit dem Straßen- und Nahverkehr stehen. Auch entfernte Bereiche können Faktoren für die Verkehrsbeeinflussung enthalten. Denkbare relevante Themenfelder sind neben Infrastruktur, Verkehr allgemein und Nahverkehr auch Wetter und Kultur. Hat man die Kategorisierung vorgenommen kann innerhalb dieser Themen nach speziellen relevanten Elementen gesucht werden. Da für die Mobilitätsanalyse des Projekts lediglich der Stadtbereich festgelegt worden ist kann nicht der Verkehrsbereich generell als relevant betrachtet werden. Zur Abbildung und Berechnung von Verkehrssituationen sind Daten einer Straßenkarte erforderlich. Um zudem auch den Nahverkehr zu berücksichtigen müssen auch Fahrplanauskünfte, Haltestellen und Liniennetzpläne im System vorliegen. Als wesentliche verkehrsbeeinflussende Faktoren müssen Informationen zu Unfällen, Staus und Baustellen schnell, möglichst in Echtzeit, abrufbar sein. Aber auch indirekte Einflüsse wie die steigende Anzahl an KFZ-Zulassungen sind hilfreich. Als weitere Einflüsse sollen möglichst aktuelle Wetterdaten, sowie wiederkehrende Großveranstaltungen, etwa Heimspiele der EWE Baskets Oldenburg berücksichtigt werden.

3.2 Phase 2: Form

In der zweiten Phase werden die nutzbaren Formate und vorliegenden Schnittstellen berücksichtigt. Wichtig hierbei ist, dass auf proprietäre Formate größtenteils verzichtet werden sollte. SAP Hana ermöglicht den Import vieler Standardformate, darunter auch das Microsoft Excel Format XLSX. Als offene Formate werden unter anderem CSV und XML unterstützt. Bezüglich der Schnittstellen gelten ähnliche Regeln. Die Hana Appliance bietet vorhandene Schnittstellen, darüber hinaus können weitere Schnittstellen programmiert werden. Zu beachten ist, dass ein automatischer Import der Daten ermöglicht werden muss, da ansonsten keine Echtzeit gewährleistet werden kann.

3.3 Phase 3: Datenakquise

Unter dem Schlagwort Datenakquise wird die eigentliche Suche nach relevanten Daten zusammengefasst. Diese Phase ist unterteilt in zwei Bereiche. Zum einen muss der Umstand

berücksichtigt werden, dass noch keine Suche stattgefunden hat und auf keine Basis an brauchbaren Portalen oder Anbietern zurückgegriffen werden kann, zum anderen könnte eben diese Erfahrung bei der Suche vorliegen. Erster Schritt jeder Suche wird die Verwendung einer Suchmaschine sein. Über Keywords zu relevanten Daten oder Bereichen werden die ersten Ergebnisse gefunden. Diese müssen jedoch kritisch untersucht werden. Eine weitere Möglichkeit ist die Anfrage bei Institutionen, die Daten interessanter Bereiche vorhalten könnten. Hier sind Behörden, Unternehmen, Verkehrsbetriebe, Vereine oder Bildungseinrichtungen zu nennen.
Abschließend soll beschrieben werden, welche Arten von Anbietern generell denkbar sind. Häufigste Form sind Datenportale. Diese werden meist von öffentlichen Einrichtungen betrieben, beispielsweise dem Bund, den Ländern oder Kommunen. Datenportale können sowohl fachliche als auch räumliche Bereiche abdecken. Eine weitere Möglichkeit ist bereits im Abschnitt Datenbereitstellung genannt, die APIs. Das sind Programmierschnittstellen die genutzt werden können, indem sie an das eigene System angebunden werden. Anschließend sind die bereitgestellten Daten oder Services erreichbar, ohne das eigene System verlassen zu müssen. Letzte, umständlichere Methode, ist die Kontaktaufnahme mit thematisch ähnlichen Projekten. Diese standen zum Anfang ihrer Arbeit vor den gleichen Problemen und erlauben eventuell den Zugriff auf gesammelte Daten. Nachteil ist dabei eventuell die fehlende Aktualität der Daten.

3.4 Phase 4: Nutzbarkeit (Form)

Als Ergebnis der Datenakquise liegen Rohdaten oder APIs vor, die in das System integriert werden müssen. Dazu muss beachtet werden, dass die Daten in einem geeigneten Format vorliegen. Geeignet bedeutet in diesem Falle zum einen, dass es sich um ein offenes Format handelt, zum anderen, dass es kompatibel zum eigenen System ist. Um gefundene Datensätze überhaupt nutzen zu können, muss dieser entsprechend zugänglich gemacht werden. Der Zugang per Download, der zudem kostenlos ist, ist die einfachste Form.

3.5 Phase 5: Validität (Lizenz)

Als letzter Schritt der Datenbeschaffung muss die Validität überprüft werden. Im Fokus steht hierbei die Lizenz. Wie bereits beschrieben wurde können die Nutzungsrechte eingeschränkt werden. Zur Verarbeitung der Daten müssen diese zum Beispiel getrennt und weiterverarbeitet werden, diese zulässigen Einschränkungen dürfen daher nicht vorliegen. Um auch eine kommerzielle Nutzung zu gewährleisten bedarf es einer offenen Lizenz.
Sind diese fünf Phasen abgeschlossen liegt ein geeigneter Datensatz vor, der für die Ana-

lyse im Mobilitätsumfeld relevant ist. In einem weiteren Prozess muss je nach Typ der Bereitstellung (z.B. periodische Bereitstellung der Daten), Priorität für Betrachtung et cetera eine Strategie für den Import entwickelt werden.

4 Anbieter

4.1 Deutschland

Aufgrund von Gesetzen und Verordnungen, sowie Forderungen von gemeinnützigen Organisationen wird am Open Data Portal govdata.de gearbeitet. Das Portal, welches vom Bundesministerium des Innern betrieben wird, befindet sich derzeit in einer Beta Phase. Das Fraunhofer-Institut für offene Kommunikationssysteme (FOKUS) ist zuständig für das Hosting des Portals und die technische Umsetzung. Angeboten werden offene Daten, Dokumente und Apps von Bund, Ländern und Kommunen. Über eine Suchfunktion kann nach Keywords in den Metainformationen gesucht werden. Es kann außerdem nach verschiedenen Kategorien gefiltert werden. Grundsätzlich werden die Daten des Herausgebers geschützt durch die Lizenz Creative Commons Namensnennung 3.0 Deutschland (CC BY 3.0), jedoch ist es Anbietern gestattet bei der Bereitstellung der Daten eigene Nutzungsbestimmungen festzulegen. Die Daten werden von den Anbietern dezentral vorgehalten (vgl. [Ges15]). Dieses Lizenzmodell sorgt für große Kritik von Seiten der gemeinnützigen Organisationen. Fehlende Standardisierung und veraltete Daten sind Hauptaspekte der Kritik. Durch die individuell festzulegenden Nutzungsbeschränkungen werden die Rechte außerdem weiter eingeschränkt. Es wurde daher eine Petition mit dem Namen „not your GOVDATA“ erstellt, auf der jeder Bürger oder jede Organisation seine Stimme für eine grundsätzliche Überarbeitung des Datenportals abgeben kann (vgl. [Ope15g, Klo10]).
Das Portal offenedaten.de wird im Gegensatz zu govdata.de nicht von einer öffentlichen Stelle, sondern von einer Community betrieben. Das dort vorliegende Datenangebot soll nicht in Konkurrenz zu amtlichen Datenportalen treten, sondern für eine Ergänzung des Angebots sorgen. Die Pflege der Datensätze geschieht durch die Community, es ist demnach jedem Nutzer möglich, eigene Datensätze hochzuladen und auffindbar zu machen (vgl. [Ope15i]).
Als Beispiel für Datenkataloge der Länder können die Portale der Städte Hamburg und Berlin genannt werden. Hier werden speziell Datensätze für die entsprechende Region zur Verfügung gestellt. Das Portal transparenz.hamburg.de wird von der Stadt Hamburg betrieben, daten.berlin.de als Berliner Portal wird von der Senatsverwaltung für Wirtschaft, Technologie und Forschung herausgegeben. Vorteil von Datenportalen für kleinere Regio-

nen ist die höhere Pflegebereitschaft, da die Verantwortung klar auf eine Stelle fokussiert ist. Dadurch wird das Interesse gestärkt, Datensätze schneller und skalierter anzubieten. Auch die kürzeren Dienstwege in kleineren Organisationen verstärken diesen Effekt. Im Gegensatz dazu sorgt eine Vielzahl an Datenportalen öffentlicher Stellen für einen höheren Verwaltungsaufwand. Daten müssen auf verschiedenen Portalen bereitgestellt werden, um mögliche Richtlinien zur Bereitstellung zu erfüllen (vgl. [Fre15, Sen15]).
Lange nicht alle Daten sind zentral abrufbar. Gerade in spezifischen Bereichen, bei denen das öffentliche Interesse gering ist, oder wirtschaftliche Potentiale dafür sorgen, dass Unternehmen die Bereitstellung zu unterbinden versuchen, werden Daten nur auf Druck der Regierung auffindbar gemacht. Hier kann ein Ansatz sein, Anfragen bei Behörden oder Unternehmen zu stellen, die relevante Datensätze vorhalten. Ein Beispiel sind Wetterdaten, die vom Deutschen Wetterdienst in Wetterstationen erhoben werden. Der DWD ist eine teilrechtsfähige Anstalt des öffentlichen Rechts und untersteht dem Bundesministerium für Verkehr und öffentliche Infrastruktur. Auf der Internetseite des DWD wird mit einem kostenfreien Zugang zu Wetterinformationen geworben. Es werden mehrere Methoden für den Zugriff bereitgestellt. Eine Einschränkung ist jedoch, dass die Daten nur in Form von Vergangenheitserhebungen für den vorangegangenen Tag bereitgestellt werden. Werden aktuellere Daten benötigt, muss eine Anfrage beim DWD gestellt werden, anschließend wird ein Angebot für die entgeltliche Bereitstellung gemacht (vgl. [Bun15]).

4.2 International

Die Vereinten Nationen, speziell die United Nations Statistics Division, betreibt ein Datenportal, auf dem statistische Informationen, die von der UN vorgehalten werden, gesucht, angezeigt und heruntergeladen werden können. Das Portal data.un.org gibt eine Anzahl von 60 Millionen Datensätzen an, die über das Portal zu beschaffen sind. Hier besteht jedoch das Problem, dass es sich lediglich um Daten, die von der UN generiert wurden, handelt. Dies bedeutet, dass es sich um keine zuständige Behörde bezüglich relevanter Bereiche für das Projekt handelt (vgl. [Uni15]).
Das INSPIRE Geoportal der EU ist Teil der Umsetzung der INSPIRE Richtlinie, es soll als zentrales Netzwerk dienen und die dezentral vorgehaltenen Daten der EU-Staaten bereitstellen. In der derzeitigen Version ist lediglich die Abfrage und Anzeige von Geodaten möglich, beispielsweise durch die Anzeige von Landkarten, Auswahl von Bereichen und anschließender Anzeige der vorgehaltenen Daten in diesen Regionen. Durch die fehlende Downloadfunktion wird die Beschaffung der Daten erschwert, was einen Nutzen für RAPID unterbindet.

Die Europäische Union stellt ein Portal bereit, bei dem ebenfalls über Schlagworte oder Kategorienfilter eine Suche durchgeführt werden kann. Informationen betreffen von EU Institutionen generierte Datensätze und Anwendungen. Die Anzahl verfügbarer Datensätze liegt bei 8.116 (Stand 06.03.2015). Der Zugang zu Daten wird per Download in einem offenen Dateiformat, beispielsweise RDF, CSV oder HTML ermöglicht. Zudem wird die Möglichkeit geboten, weitere Datensätze über ein Formular anzufragen.

4.3 Vergleich

Im internationalen Vergleich findet sich Deutschland im hinteren Bereich wieder und liegt damit hinter den eigenen Ansprüchen zurück. Die Diskussion um das Portal govdata.de stärkt diesen Eindruck (vgl. [Ope15g]). Als Vorreiter gelten die USA und Großbritannien. In den USA wurde das Datenportal von Präsident Barack Obama vorangetrieben, in Großbritannien gilt Tim Berners-Lee als treibende Kraft (vgl. [SH11]). Ein Blick auf die Anzahl der bereitgestellten Datensätze stärkt diese Aussage. 9.811 Datensätze sind auf der Plattform govdata.de zu finden, davon liegt bei lediglich 7.305 Datensätzen eine freie Lizenz vor (vgl. [Ges15]). Spitzenreiter sind die USA mit 139.597 Datensätzen (vgl. [U.S15]), stärkste europäische Open Data Plattform ist die britische Plattform data.gov.uk, hier werden 23.159 Datensätze zur Verfügung gestellt (vgl. [Dat15]). Frankreich liegt mit 13.987 Datensätzen knapp vor Deutschland. Auch wenn weitere Faktoren wie die Einwohnerzahl, die Fläche oder die Anzahl an Verwaltungsmitarbeiter in die Betrachtung einbezogen werden muss, zeigt diese Gegenüberstellung den aktuellen Stand der Entwicklung (vgl. [Eta15]). (Stand 04.02.2015)

5 Relevante Datensätze

Im letzten Teil der Arbeit werden relevante Datensätze für die anstehende Projektarbeit vorgestellt. Dabei wurde nach dem Schema der Datenauswahl vorgegangen und sowohl auf bekannte Plattformen wie govdata.de sowie auf die Websuche zurückgegriffen.
Die Suche über govdata.de veranschaulicht die im vorigen Kapitel angesprochene Kritik an der Plattform. Werden interessante Datensätze gefunden, sind diese veraltet oder aufgrund ihrer Skalierung unbrauchbar. Ein gefundener Datensatz stellt den Kraftfahrzeugbestand zum Stichtag 01.01.2014 dar, die auf Landkreise skaliert wurden, und somit auch den Stadtbereich Oldenburg enthalten. Die Tabelle schlüsselt zudem nach Fahrzeugtyp auf. Die Beschaffung kann per Download als CSV oder XLSX Datei stattfinden, sowie über den Link zur CKAN API (vgl. [Ges15]).

Im Bereich Mobilität fallen häufig Geodaten an, also Daten mit räumlichem Bezug. Die oben beschriebene INSPIRE Richtlinie des europäischen Parlaments soll die Bereitstellung von Geodaten sichern. Das zentrale Portal zu INSPIRE ist jedoch zum gegenwärtigen Zeitpunkt nicht für den Export von Daten geeignet, weshalb andere Datenquellen betrachtet werden müssen (vgl. [Eur15]). Zur Darstellung von Geodaten auf einer Landkarte hat sich Google Maps durchgesetzt. Es besteht beispielsweise die Möglichkeit, Layer auf verschiedenen Kartentypen einzuzeichnen oder kürzeste Wege zu berechnen. Diese Optionen machen Google Maps zu einer mächtigen API für Geodaten. Bis zu einem gewissen Punkt kann die API kostenlos genutzt werden. Aufgrund von Zugriffsbeschränkungen wird jedoch nur eine bestimmte Anzahl an Zugriffen pro Tag zugelassen. Weitere Einschränkung ist, dass der Nutzer nicht auf die Datensätze an sich zugreifen kann, sondern lediglich die Services nutzen darf. Abschließend kann gesagt werden, dass Google Maps als API für die Darstellung im Web nützlich sein kann, dass jedoch die Berechnung in Echtzeit ausgeschlossen wird, ebenso wie die Offline Nutzung (vgl. [Goo15]).
Als Alternative wird Openstreetmap vorgestellt. Es handelt sich hierbei um ein Projekt für eine offene Weltkarte. Die Daten werden von den Nutzern gesammelt, indem sie mit einem Tracker ausgestattet Wege abfahren oder laufen, die georteten Daten werden dann in die Datenbank aufgenommen. Vorteil von Openstreetmap gegenüber Google Maps ist, dass die Daten als offene Daten im XML Dateiformat heruntergeladen werden können, um diese in das eigene System zu integrieren. Dabei kann ein Export für einen definierten Bereich oder für den gesamten Datenbestand durchgeführt werden. Nutzungsbeschränkungen liegen nicht vor. Durch die Integration der Daten in das eigene System sind Berechnungen in der Hana durchführbar, was zur Steigerung der Performanz führen kann. Auch die Verknüpfung mit eigenen gesammelten Daten und die Nutzung außerhalb des Internets werden so ermöglicht (vgl. [FOS15]).
Eine weitere Vorgabe ist die Berücksichtigung von Wetterdaten. Auf der Seite des Deutschen Wetterdienstes wird ein öffentlicher FTP Zugang angegeben, der den kostenlosen und direkten Zugriff auf Wetterdaten ermöglicht. Die Daten können für sämtliche Wetterstationen in Deutschland abgerufen werden und sind in stündlichem, täglichem und wöchentlichem Format verfügbar. Neben Angaben zur Luft- und Bodentemperatur sind weitere Daten bezüglich der Bewölkung, dem Luftdruck, Sonnen- und Niederschlagswerten sowie Windstärken verfügbar. Die Wetterstation Oldenburg ist im Jahr 2012 geschlossen worden, stattdessen wurde sie in den Ort Friesoythe-Altenoythe verlegt, was eine Entfernung von ca. 20km zur Universität Oldenburg bedeutet. Weitere Einschränkung ist der zeitliche Verzug in der Bereitstellung. Täglich gegen ca. 10:15 Uhr werden die Daten für den Vortag auf den FTP-Server hochgeladen (vgl. [Bun15]).

Für unregelmäßig oder einmalig anfallende Daten empfiehlt sich die Form der manuellen Datenerhebung. Für die Berücksichtigung von Nahverkehrsdaten kann beispielsweise der Liniennetzplan der VWG manuell erfasst und in die Datenbank integriert werden (vgl. [Ver15]). Regelmäßige Veranstaltungen sind über die Seiten der Veranstalter zu beschaffen. Die Stadt Oldenburg bietet auf der Internetseite der Stadt einen Veranstaltungskalender an (vgl. [Sta15]), die EWE Baskets Oldenburg stellen die Spielpläne ebenfalls auf der eigenen Webseite zur Verfügung, der Spielplan kann hier sogar als XML-Datei exportiert werden (vgl. [Bas15]). Eine regelmäßige Beschaffung von Daten durch manuelle Erhebung ist nicht empfehlenswert, da die Gefahr besteht, dass die Daten veralten und nicht alle vorgesehenen Faktoren in die Berechnungen einfließen können.

6 Fazit

Die Arbeit stellt das Prinzip von Open Data dar. Dabei wurde der Begriff zunächst definiert, im Anschluss daran sind Voraussetzungen dargestellt worden. Dabei sind auch gesetzliche Regelungen genannt worden. Nach Betrachtung der Potentiale sind verwandte Bereiche abgegrenzt worden. Der Teil wurde durch Betrachtung von Regeln und einer Vorgehensweise zur Datenbereitstellung abgeschlossen. Darüber hinaus wird ein Prozess zur Datenauswahl konzipiert. Dieser soll bei der Vorauswahl relevanter Datensätze helfen und so die Informationsqualität späterer Analyseergebnisse erhöhen. Nach einer Vorstellung potentieller Anbieter wurden relevante Datensätze auf ihre Relevanz für das Projekt betrachtet. Es wurde untersucht in wie weit offene Datenquellen für das Projekt RAPID einen Mehrwert haben. Allgemein lässt sich sagen, dass offene Daten eine Aufwand- und Kostenminimierung mit sich bringen, da Daten nicht eigenständig erhoben werden müssen. Zur Datenanreicherung und Datenergänzung eignen sich offene Daten außerdem, da weitere Faktoren betrachtet werden können. Gerade Wetter- und Kartendaten haben einen großen Nutzen, wenn sie korrekt eingesetzt werden. Die Open Data Situation in Deutschland führt jedoch zu einigen Problemen. Gerade die Aktualität der Daten spielt eine wichtige Rolle, diese ist jedoch oft nicht gegeben. Ein weiteres Problem ist die Skalierung der Daten, gerade in Open Data Portalen. Datensätze sind aggregiert und besitzen somit wenig Aussagekraft. Dies äußert sich in der Kritik am Open Data Portal des Bundes govdata.de, auf dem Nutzungsrechte an Daten von Bereitstellern eingeschränkt werden können, zudem ist die vom Portal vorab festgelegte Lizenz nicht standardisiert, sondern für das Portal geschaffen worden. Es wird daher diskutiert, in wie weit das Geodatenzugangsgesetz oder das Recht an amtlichen Informationen, welches im Informationsfreiheitsgesetz geregelt wird, durch das Portal Beachtung findet. Dennoch sind Daten vorhanden, die einen Mehrwert liefern können, wenn sie korrekt eingesetzt werden. Zu nennen sind hier vor allem Geodaten der Seite Openstreetmap, sowie die Wetterdaten des DWD. Darüber hinaus müssen Daten teilweise manuell erfasst werden, etwa Daten zu Großveranstaltungen. Auch Informationen aus dem Bereich Nahverkehr müssen in das System aufgenommen werden, allerdings liegen diese zumeist als unstrukturierte Daten vor. Es müssen also weitere Prozesse, wie ein ETL Prozess, definiert werden, die durch Auslesen, Anpassen und Laden solcher Datensätze vom System genutzt werden können. Eine breite Datenvielfalt kann mithilfe von Open Data Sources erwirkt werden, es ist jedoch eine aufwändige Datenakquise notwendig. Diese Arbeit kann bei der Durchführung des Projekts unterstützen, indem es das Prinzip und die Relevanz zur Datenanreicherung mithilfe externer Quellen darstellt und ein mögliches Vorgehensmodell vorstellt.

Literatur

[Bä07] Bärwolff, B. Lutterbeck; M.: *Open Source Jahrbuch 2007*. •, 2007

[Bas15] Baskets Oldenburg GmbH & Co. KG: *Spielplan*. http://ewe-baskets.de/heimspiel/spielplan. Version: 2015, Abruf: 07.03.2015

[BL06] Berners-Lee, T.: *Linked Data – Design Issues*. http://www.w3.org/DesignIssues/LinkedData.html. Version: 2006, Abruf: 07.03.2015

[Bun15] Bundesministerium für Verkehr und digitale Infrastruktur: *Wetter und Klima – Deutscher Wetterdienst – Klimadaten*. http://www.dwd.de/bvbw/appmanager/bvbw/dwdwwwDesktop?_nfpb=true&_pageLabel=_dwdwww_klima_umwelt_klimadaten_deutschland&T82002gsbDocumentPath=Navigation%2FOeffentlichkeit%2FKlima_Umwelt%2FKlimadaten%2Fkldaten__kostenfrei%2Fabrufsysteme__ftp_home__node.html%3F__nnn%3Dtrue. Version: 2015, Abruf: 07.03.2015

[Dat15] Data.gov.uk: *Data.gov.uk*. http://www.data.gov.uk/. Version: 2015, Abruf: 04.02.2015

[Die11a] Dietrich, D. ; Bundeszentrale für politische Bildung (Hrsg.): *Bpb.de – Open Data – Nutzen offener Daten*. http://www.bpb.de/gesellschaft/medien/opendata/64058/nutzen-offener-daten?p=1. Version: 2011, Abruf: 07.03.2015

[Die11b] Dietrich, D. ; Bundeszentrale für politische Bildung (Hrsg.): *Bpb.de – Open Data – Offene Daten in Deutschland*. http://www.bpb.de/gesellschaft/medien/opendata/64061/offene-daten-in-deutschland?p=1. Version: 2011, Abruf: 07.03.2015

[Die11c] Dietrich, D. ; Bundeszentrale für Politische Bildung (Hrsg.): *Open Data*. http://www.bpb.de/gesellschaft/medien/opendata/64055/was-sind-offene-daten. Version: 2011, Abruf: 07.03.2015

[Eta15] Etalab: *Accueil – Data.gouv.fr*. https://www.data.gouv.fr/fr/. Version: 2015, Abruf: 04.02.2015

[Eur15] European Union: *INSPIRE Geoportal*. http://inspire-geoportal.ec.europa.eu/. Version: 2015, Abruf: 07.03.2015

[FOS15] FOSSGIS e.V.: *OpenStreetMap*. http://www.openstreetmap.org/export#map=12/53.1445/8.2137. Version: 2015, Abruf: 07.03.2015

[Fre15] Freie und Hansestadt Hamburg: *Transparenzportal Hamburg*. http://transparenz.hamburg.de/. Version: 2015, Abruf: 07.03.2015

[Gei10] Geiger, C. P.; Lucke, J. v.: Open Government Data – Frei verfügbare Daten des öffentlichen Sektors. In: *Deutsche Telekom Institute for Connected Cities* (2010)

[Ges15] Geschäfts- und Koordinierungsstelle GovData ; Bundesministerium des Innern (Hrsg.): *GovData - Datenportal für Deutschland*. http://govdata.de. Version: 2015, Abruf: 07.03.2015

[Goo15] Google Inc.: *Google Maps API*. https://developers.google.com/maps/?hl=de. Version: 2015, Abruf: 07.03.2015

[Gru14] Gruebler, M. ; Stadt Zurich (Hrsg.): *Open Data der Stadt Zurich*. http://de.slideshare.net/Opendatazurich/open-data-der-stadt-zrich-was-bringt-es. Version: 2014, Abruf: 07.03.2015

[Her10] Herb, U.: *Open Initiatives: Offenheit in der digitalen Welt und Wissenschaft*. universaar, 2010

[INS] Bundesamt für Kartographie und Geodäsie (Hrsg.): *Was ist INSPIRE*. https://www.geoportal.nrw.de/application-informationen/inspire/images/flyer_inspire.pdf, Abruf: 07.03.2015

[Klo10] Klopp, T.: Open Data: Mit alten Daten neues Wissen schaffen. In: *Zeit Online* (2010)

[Lan12] Landesamt für Geoinformation und Landentwicklung Niedersachsen: *Die INSPIRE-Richtlinie - Aufbau einer europäischen Geodateninfrastruktur*. http://www.geodaten.niedersachsen.de/download/26328/INSPIRE-_kompakt_zusammengefasste_INSPIRE-Artikel.pdf. Version: 2012, Abruf: 07.03.2015

[Ope15a] Open Knowledge Foundation: *ckan – The open source data portal software*. http://ckan.org/. Version: 2015, Abruf: 07.03.2015

[Ope15b] Open Knowledge Foundation: *Open Definition*. http://opendefinition.org/od/. Version: 2015, Abruf: 07.03.2015

[Ope15c] Open Knowledge Foundation: *Open Knowledge: About*. https://okfn.org/about/. Version: 2015, Abruf: 07.03.2015

[Ope15d] Open Knowledge Foundation: *Where Does My Money Go?* http://wheredoesmymoneygo.org/. Version: 2015, Abruf: 07.03.2015

[Ope15e] Open Knowledge Foundation Deutschland e.V.: *APPS FÜR DEUTSCHLAND*. http://apps4deutschland.de/. Version: 2015, Abruf: 07.03.2015

[Ope15f] Open Knowledge Foundation Deutschland e.V.: *Haushalte von Bund, Ländern und Kommunen – OffenerHaushalt*. http://offenerhaushalt.de/. Version: 2015

[Ope15g] Open Knowledge Foundation Deutschland e.V.: *Not your GovData*. http://not-your-govdata.de/. Version: 2015, Abruf: 07.03.2015

[Ope15h] Open Knowledge Foundation Deutschland e.V.: *Offene Daten*. http://okfn.de/opendata/. Version: 2015, Abruf: 07.03.2015

[Ope15i] Open Knowledge Foundation Deutschland e.V.: *OffeneDaten.de*. http://www.offenedaten.de. Version: 2015, Abruf: 07.03.2015

[Ope15j] Open Knowledge Foundation Deutschland e.V.: *Projekte — Open Knowledge Foundation Deutschland*. http://okfn.de/projects/. Version: 2015, Abruf: 07.03.2015

[Sen15] Senatsverwaltung für Wirtschaft, Technologie und Forschung: *Offene Daten Berlin*. http://daten.berlin.de/. Version: 2015, Abruf: 07.03.2015

[SH11] Schulzki-Haddouti, C. ; Bundeszentrale für politische Bildung (Hrsg.): *Die globale Bewegung für offene Daten*. http://www.bpb.de/gesellschaft/medien/opendata/64063/globale-entwicklung. Version: 2011, Abruf: 07.03.2015

[Sta15] Stadt Oldenburg (Oldb): *Veranstaltungen – Stadt Oldenburg*. http://www.oldenburg.de/sonderseiten/veranstaltungen.html. Version: 2015, Abruf: 07.03.2015

[Uni15] United Nations Statistics Division: *UNdata*. `http://data.un.org/Default.aspx`. Version: 2015, Abruf: 07.03.2015

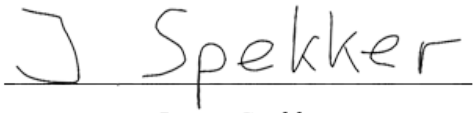
[U.S15] U.S. General Services Administration: *Data.gov*. `http://www.data.gov/`. Version: 2015, Abruf: 04.02.2015

[Ver15] Verkehr und Wasser GmbH: *VWG – Verkehr und Wasser GmbH Oldenburg – Fahrplan*. `https://www.vwg.de/Fahrplan.html`. Version: 2015, Abruf: 07.03.2015

Abschließende Erklärung

Ich versichere hiermit, dass ich meine Seminarausarbeitung im Rahmen der Projektgruppe RAPID selbständig und ohne fremde Hilfe angefertigt habe und dass ich alle von anderen Autoren wörtlich übernommenen Stellen wie auch die sich an die Gedankengänge anderer Autoren eng anlegenden Ausführungen meiner Arbeit besonders gekennzeichnet und die Quellen zitiert habe.

Westoverledingen, den 8. März 2015



Jannes Spekker

VERY LARGE
BUSINESS APPLICATIONS
Carl von Ossietzky Universität Oldenburg

Entwicklungsumgebungen und Frameworks um SAP HANA

Hausarbeit
im Rahmen der Projektgruppe „Regional Analysis and Prediction Platform by In-memory Data (RAPID)“

Themensteller:	Prof. Dr.-Ing. Jorge Marx Gómez
Betreuer:	M.Sc Daniel Stamer

Vorgelegt von:	Nils Worzyk Otto-Suhr-Straße 22 26131 Oldenburg 0177 5901596 nils.worzyk@uni-oldenburg.de

Abgabetermin:	99. Januar 9999

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einleitung

Das grundsätzliche Ziel der Projektgruppe „Regional Analysis and Prediction Platform by In-memory Data (RAPID)“ ist es, Daten in einer In-Memory Datenbank zu bearbeiten. Als zu verwendende Datenbank wird dafür ein SAP HANA System zur Verfügung gestellt, welches in Abschnitt 2 kurz eingeführt wird.

Ein besonderes Augenmerk wird dann in Abschnitt 3 auf die Architektur der SAP HANA gelegt werden und darauf, wie von außerhalb auf die Funktionalität der HANA zugegriffen werden kann. In diesem Zusammenhang gibt es zwei unterschiedliche Vorgehensweisen, die in Abschnitt 3.1 diskutiert werden. Außerdem wird in Abschnitt 3.2 darauf eingegangen, wie intern in der HANA die Funktionalität gewährleistet wird und in Abschnitt 3.3 wird eine sehr wichtige Erweiterung, die sogenannten „Extended Application Services“, kurz XS oder XS Engine, erklärt.

In Abschnitt 4 wird dann das SAP HANA Studio, beziehungsweise die korrespondierenden Eclipse Plug-Ins vorgestellt, welches als Entwicklungsumgebung von der Projektgruppe genutzt werden kann um die Anwendungsentwicklung mit der HANA durchzuführen. Dafür werden zwei unterschiedliche „Views“, die das Studio zur Verfügung stellt, kurz erläutert. Zum einen wird kurz auf die „Modeler“-View, in Abschnitt 4.1 und zum anderen etwas ausführlicher auf die „Development“-View, in Abschnitt 4.2 eingegangen.

Abschließend wird in Abschnitt 5 eine kurze Zusammenfassung der Hausarbeit gegeben und ein Fazit gezogen, wie die Projektgruppe am besten mit der SAP HANA arbeiten kann um das gegebene Ziel zu erreichen.

2 SAP HANA

SAP HANA ist eine von SAP entwickelte und vertriebene Lösung, die durch eine Kombination aus Hard- und Software Echtzeitanalysen oder -transaktionen unterstützen soll. Auf der Softwareseite stellt die SAP HANA, wie auch schon frühere Systeme, zeilen- und spalten-, aber auch objekt-orientierte Möglichkeiten zur Speicherung von Daten zur Verfügung.[8] Die bedeutendere Verbesserung, beziehungsweise Neuerung erfolgt auf der Hardwareseite. Denn um eine Geschwindigkeit bei den Analysen oder Transaktionen zu erreichen, bei der von Echtzeit gesprochen werden kann, nutzt das System In-Memory Technologie.[8]

Um In-Memory Technologie zu erklären, wird zunächst ein Schritt in die „Vergangenheit“ getan. Bei herkömmlichen Datenbanken werden die Daten auf einer ausreichend großen Festplatte gespeichert. Wenn eine Berechnung durchgeführt werden soll, werden

die benötigten Daten in den Hauptspeicher geladen, um von dort für die Berechnung herangezogen zu werden. Wenn die Berechnung mehr Daten benötigt müssen alte Daten aus dem Hauptspeicher wieder auf die Festplatte geschrieben werden und die neuen Daten geladen werden. All diese Lese- und Schreiboperationen kosten Zeit, vor allem deswegen, weil die Zugriffszeiten auf die Festplatte um ein vielfaches höher sind, als die Zugriffszeiten auf den Hauptspeicher.[3] Daraus hat sich die Idee entwickelt den Hauptspeicher so groß zu gestalten, dass es nicht mehr nötig ist die Daten zwischen Hauptspeicher und Festplatte hin und her zu schreiben. Dieses Nutzen des Hauptspeichers als „quasi Festplatte“ wird In-Memory Technologie genannt und führt zu einer generellen Verbesserung der Performance.

Durch diese Verwendung von In-Memory Technologie muss auf der Hardwareseite die entsprechende Infrastruktur zur Verfügung gestellt werden. Dem Nutzer können je nach Kostenaufwand 2 oder 4 Terabyte Hauptspeicher zur Verfügung gestellt werden. Wenn es der finanzielle Aufwand zulässt können allerdings auch Terabyte im Zehnerbereich oder mehr genutzt werden.[8] Alleine der größere Hauptspeicher reicht allerdings nicht aus um ein optimales System zu liefern. Deswegen werden SAP HANA Systemen mit 2 - 8 Rechenkernen und 128GB - 2TB Hauptspeicher pro Server ausgeliefert, welche wiederum in Cluster zusammengeschaltet werden. 2013 lag der Weltrekord bei 12.1 PetaByte Speicherkapazität für ein Datenbanksystem.[10]

Die SAP HANA Technologie ist weiterhin eine noch relativ junge Anwendung. Das zeigt sich auch dadurch, dass erst Ende November 2010 [6] die erste SAP HANA Anwendung ausgeliefert wurde. In den folgenden Jahren wurden dann immer weitere Features herausgebracht, beispielsweise wurde im September 2011 der HANA support für SAP NetWeaver Business Warehouse angekündigt [1] oder als weiteres Beispiel hat SAP im Oktober 2012 ein neues Angebot angekündigt, mit welchem die HANA in der Cloud über beispielsweise die Amazon Web Services stundenweise genutzt werden kann.[2]

3 SAP HANA Architektur und Schnittstellen

Nach der Einführung und zeitlichen Einordnung der HANA in Abschnitt 2 wird in diesem Abschnitt genauer auf die Architektur der HANA eingegangen. Eine Darstellung der Gesamtarchitektur der HANA ist in Abbildung 1 zu sehen.

Wie in der Abbildung zu sehen ist, kann die Arbeit mit der HANA in zwei große Bereiche aufgeteilt werden. Zum einen gibt es die Clientseite, welche in Abschnitt 3.1 näher betrachtete wird. Dabei werden verschiedene Schnittstellen vorgestellt, wie ein Anwender mit der HANA kommunizieren kann.

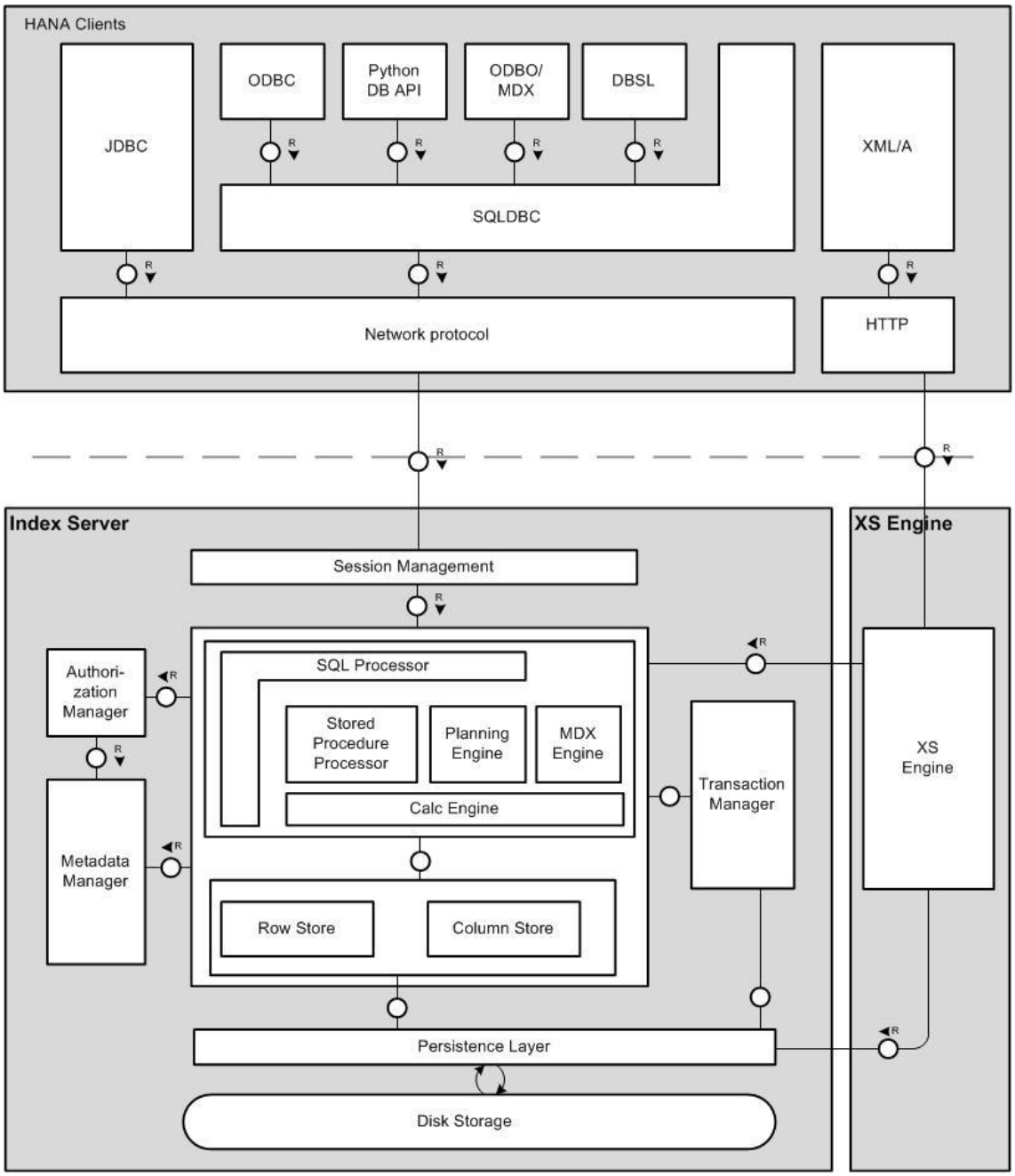


Abbildung 1: Architektur der HANA und der Zugriffsmöglichkeiten auf die HANA, Quelle: nach [4]

Der andere große Bereich ist die Serverseite, wo die „eigentliche“ Arbeit mit den Daten stattfindet. Dieser serverseitige Bereich wiederum lässt sich ebenfalls in zwei Bereiche unterteilen. Zum einen den Indexserver, welcher für die Speicherung und Verarbeitung der Daten zuständig ist und in Abschnitt 3.2 etwas genauer erklärt wird. Zum anderen findet sich auf der Serverseite die XS Engine („Extended Application Services“) welche in Abschnitt 3.3 näher beleuchtet wird.

3.1 HANA Clients

Auf der Clientseite gibt es zwei Möglichkeiten der Anwendungsentwicklung und der damit verbundenen Kommunikation mit der HANA. Zum einen gibt es die „Nicht-native Anwendungsentwicklung“ die in Abschnitt 3.1.1 näher beleuchtet werden soll. Sie ist vergleichbar mit der traditionellen Kommunikation zwischen Anwendung und Datenbank. Die zweite Möglichkeit ist die „Native Anwendungsentwicklung“ auf welche in Abschnitt 3.1.2 näher eingegangen wird.

3.1.1 Nicht-native Anwendungsentwicklung

Wie bereits in der Einleitung angedeutet, ist die nicht-native Anwendungsentwicklung vergleichbar mit der traditionellen Kommunikation zwischen einer Anwendung und einer Datenbank. Auf der einen Seite steht dabei die Anwendung, die in Java oder einer anderen Sprache geschrieben und sowohl eine eigenständige, wie auch eine Web-basierte Anwendung sein kann. Wenn innerhalb dieser Anwendung ein Datenbankzugriff benötigt wird, wird über eine Datenbankschnittstelle wie beispielsweise JDBC[1] eine Verbindung zu der Datenbank aufgebaut. Das folgende Codebeispiels soll diesen Prozess am Beispiel von Java verdeutlichen.

```
1  import java.sql.*;
2
3  public void connect() {
4
5      Connection con;
6      Statement stmt;
7
8     String url = "jdbc:mySubprotocol:myDataSource";
9
10     con = DriverManager.getConnection(url, "myLogin", "myPassword");
11
```

[1] Abkürzung für „Java Database Connectivity“.

```
12      stmt = con.createStatement();
13
14      //create the SQL statement you want to execute
15      String sqlStatement = "CREATE COLUMN TABLE RAPID_NILS.RAPID( MEMBERS
            NVARCHAR (20) null)";
16
17      stmt.executeUpdate(sqlStatement);
18
19      stmt.close();
20     con.close();
21  }
```

In Zeile 10 des Codebeispiels wird die Verbindung zu der Datenbank aufgebaut. Dafür wird zunächst die URL der Datenbank benötigt, auf die zugegriffen werden soll. Gefolgt von dem Nutzer, der auf die Datenbank zugreifen möchte und dem zugehörigen Passwort.

In Zeile 12 wird dann das Statement vorbereitet, welches an die Datenbank übermittelt werden soll. In dem angegebenen Codebeispiel soll ein neuer Table mit dem Namen `RAPID` erstellt werden, der eine Spalte mit der Überschrift `MEMBERS` haben soll, in welcher Daten vom Typ `NVARCHAR` mit einer maximalen Länge von 20 Zeichen gespeichert werden sollen.

Dieses SQL-Statement wird dann in Zeile 17 des Codebeispiels über ein Netzwerkprotokoll an die Datenbank gesendet und dort ausgeführt. Zuletzt müssen sowohl das Statement, als auch die Verbindung zur Datenbank wieder getrennt werden, da dies nicht automatisch passiert.

Für die anderen Programmiersprachen gibt es entsprechend eigene Datenbankschnittstellen. Der Unterschied zwischen den Datenbankschnittstellen der anderen Programmiersprachen und JDBC ist lediglich, dass die anderen Schnittstellen auf SQLDBC[2] basieren. SQLDBC selber ist eine Bibliothek, die es Anwendungen ermöglicht SQL-Befehle in der Datenbank auszuführen und dadurch auf die Daten zuzugreifen oder diese zu manipulieren. Im Vergleich zu JDBC soll außerdem der Zugriff auf die Datenbank schneller sein, weil SQLDBC auf einer niedrigeren Abstraktionsebene arbeitet, als JDBC.[13] Da die folgenden vier Datenbankschnittstellen auch in Abbildung 1 vorkommen, sollen sie an dieser Stelle kurz erklärt werden.

1. ODBC („Open Database Connectivity“), wird hauptsächlich in C/C++ basierten Programmen verwendet

2. Python DB API, Schnittstelle für Python basierte Anwendungen

[2] Abkürzung für „SQL Database Connectivity“.

3. ODBO (basiert auf OLE DB for OLAP) und stellt eine Schnittstelle für analytische Anwendungen dar

4. DBSL („Database Shared library“) stellt eine Bibliothek dar, die die Anfragen des Kunden in SAP spezifische Anfragen umwandelt, und umgekehrt

Der programmatische Zugriff auf die Datenbank läuft allerdings auch bei diesen Schnittstellen ähnlich ab, wie in Java. Zunächst muss eine Verbindung aufgebaut werden, dann wird das gewünschte Statement an die Datenbank übermittelt und zuletzt wird die Verbindung wieder geschlossen.

Wichtig hierbei und ausschlaggebend für die Nicht-native Anwendungsentwicklung ist, dass nur die Auswertung des SQL-Statements innerhalb der Datenbank ausgeführt wird. Alles andere passiert beim Client.

3.1.2 Native Anwendungsentwicklung

Die zweite Möglichkeit Anwendungen mit Zugriff zur HANA zu entwickeln ist die native Anwendungsentwicklung. Die Anwendungen die hierbei entwickelt werden greifen per XML/A[3][4] auf die HANA zu und sind deswegen web-basiert. Einer der Vorteile davon ist, dass die Anwendungen weitestgehend plattformunabhängig sind. Weitere Vorteile, die für diese Art des Datenbankzugriffs genannt werden sind, dass die Zeit für Anfragen an den Server minimiert ist und dass der Nutzer, sobald er die angeforderten Daten bekommen hat, automatisch wieder getrennt wird. Dadurch wird eine bessere Skalierbarkeit im Bezug auf die Anzahl von Nutzern gewährleistet.[11]

Diese Vorteile können dadurch erreicht werden, dass die Anwendung nicht mehr, wie bei der nicht-nativen Anwendungsentwicklung, auf Seiten des Clients ausgeführt wird, sondern weitestgehend auf der HANA selber, beziehungsweise innerhalb der XS Engine, die in Abschnitt 3.3 näher erläutert wird. Nur noch das Rendering der Anwendung findet Clientseitig statt. Das die Logik der Anwendung auf der HANA ausgeführt wird hat zur Folge, dass zum einen die starke Rechenleistung der HANA zur schnelleren Berechnung genutzt werden kann, und zum anderen, dass kein großes Datenaufkommen mehr zwischen Client und Server hin und her geschickt werden muss.

Um diese serverseitige Entwicklung der Anwendung zu realisieren, wurden neue Datentypen entwickelt. Ein Beispiel für einen dieser neuen Datentypen ist .xsjs[5]. Dieser Datentyp

[3] „XML for Analysis“.

[4] OData stellt eine weiter Möglichkeit dar, web-basiert auf Daten in der HANA zuzugreifen. Diese Möglichkeit wird in Abbildung 1 zwar nicht aufgeführt, wird aber in dem SAP HANA Developer Guide [11] angegeben.

[5] xs steht für die Extended Application Services und js steht für JavaScript

löst teilweise Programmcode ab, wo in traditionellen web-basierten Anwendungen JavaScript verwendet wurde. Das folgende Beispiel soll verdeutlichen, wie der neue Datentyp auf die Datenbank zugreift.

```
1  /*
2   * By typing $. you have access to the API's objects
3   * The example code shows how to use the SAP HANA XS JavaScript API's response
        object to write HTML
4   */
5  $.response.contentType = "text/html";
6  var output = "Moin RAPID!";
7
8  var conn = $.db.getConnection();
9
10 var pstmt = conn.prepareStatement( " CREATE COLUMN TABLE RAPID_NILS.RAPID(
       MEMBERS NVARCHAR (20) null) " );
11
12 pstmt.execute();
13
14 conn.close();
15
16 $.response.setBody(output + " Ein Neuer Table mit dem Namen RAPID wurde
       angelegt!");
17 }
```

Der Zugriff auf die Datenbank in diesem Beispielcode ist in seiner Struktur sehr ähnlich zu dem von Java. In Zeile 8 wird zunächst die Verbindung zur Datenbank aufgebaut. Was hierbei auffällt ist, dass die Datenbank nicht explizit angegeben werden muss. Das liegt daran, dass das Script serverseitig vorhanden ist und bei der Erstellung des jeweiligen Projektes die zu verwendende Datenbank angegeben wird. In Zeile 10 wird dann, wie auch schon bei dem Beispielcode zu Java, das Statement angegeben, welches an die Datenbank gesendet werden soll. Das Script kann dann über einen Webbrowser ausgeführt werden.

3.2 Index Server

Der Index Server ist die Datenbank des SAP HANA Systems und der Ort wo der Großteil der Magie hinter SAP HANA passiert (frei übersetzt nach [7]). In Abbildung 1 sind für den Index Server 7 große Komponenten zu sehen.

1. Das *Session Management* dafür da, von Clients gesendete Anfragen zu verwalten und die Verbindung zur Datenbank herzustellen. Anfragen können dabei entweder von der SAP HANA authentifiziert werden, per Benutzername und Passwort, oder an externe Dienste delegiert werden.

2. Wenn die Anfrage autorisiert wurde, wird sie an den *SQL Processor* weitergeleitet. Innerhalb des SQL Processor werden die Anfragen, je nach Typ an 4 weitere Engines weitergeleitet

   - Der *Stored Procedure Processor* wertet Anfragen aus, die auf bereits vorher definierte und optimierte Prozeduren zugreift.
   - Die *Planning Engine* ist für Finanzplanung Anwendungen zuständig und erlaubt es diesen einfache Planungsoperationen durchzuführen. Ein Beispiel dafür wäre eine Finanzplanung für das nächste Jahr. Dafür werden in einem einfachen Fall die Daten des alten Jahres kopiert und mit Filtern manipuliert.[9]
   - Multidimensional Expressions (MDX) ist eine Sprache für die Anfrage und Manipulation von multidimensionalen Daten, die in OLAP-Würfeln gespeichert werden. Um Anfragen in dieser Sprache kümmert sich die *MDX Engine*.
   - Die *Calculation Engine* ist dafür da, Anfragen in Calculation Models umzuwandeln und bemüht sich dabei einen hohen Grad an Parallelisierung zu erreichen.

3. Nachdem die Anfragen durch den SQL Processor optimiert wurden, wird auf den Datenspeicher zugegriffen um die benötigten Daten zu bekommen. Die Daten, die hier angefragt werden, können auf zwei Arten gespeichert sein und sind vollständig im Hauptspeicher vorhanden. Um die Vorteile der beiden Arten von Speicherung zu zeigen soll Tabelle 1 als Beispieldatenbank dienen.

Jahr	**ProfitGes**	**ProfitDeu**	**ProfitCH**	**ProfitA**
2010	14	6	5	3
2011	14	6	6	2
2012	16	7	5	4
2013	16	7	5	4
2014	22	10	7	5

Tabelle 1: Beispiel Table mit dem Gewinn eines fiktiven Unternehmens, aufgeteilt nach dem Jahr, dem Gesamtprofit und dem Profit der Länder Deutschland (Deu), Schweiz (CH) und Österreich (A)

- Innerhalb des *Row Store* werden die Daten zeilenweise gespeichert. Das hat beispielsweise dann Vorteile wenn alle Profitwerte für ein bestimmtes Jahr angefordert werden. Die Werte stehen dann alle nebeneinander in einem bestimmten Speicherbereich und können zur weiteren Verarbeitung geladen werden.
- Innerhalb des *Column Store* werden die Daten hingegen Spaltenorientiert abgespeichert. Dies hat dann Vorteile, wenn beispielsweise alle Profitwerte von Deutschland von allen Jahren angefordert werden. Bei der spaltenweisen Speicherung der Daten stehen nun diese Werte alle in einem bestimmten Speicherbereich und können zur weiteren Verarbeitung geladen werden.

4. In Abbildung 1 befinden sich links vom SQL Processor zwei weitere Komponenten. Die eine davon ist der *Authorization Manager*. Diese Komponente wird von anderen SAP HANA Komponenten angefragt um zu prüfen ob der Nutzer, der die initiale Anfrage gestartet hat, die benötigten Rechte hat, um die angeforderte Operation durchzuführen. Diese Rechte können entweder an einzelne Benutzer vergeben werden oder an Rollenprofile und erlauben dem jeweiligen Nutzer bestimmte Operationen (beispielsweise erstellen, updaten, etc.) auf bestimmte Objekte (beispielsweise Tables, Views, etc.)

5. Die zweite Komponente auf der linken Seite des SQL Processor ist der *Metadata Manager*. Diese Komponente enthält Metadaten über eine Vielzahl an Objekten, die in der HANA gespeichert sind. Beispielsweise die Definitionen von Tables oder Views, aber auch die Definitionen von SQLScript Funktionen.

6. Auf der rechten Seite vom SQL Processor befindet sich in Abbildung 1 der *Transaction Manager*. Einzelne SQL Anfragen werden innerhalb der HANA als Transaktionen bezeichnet und diese Komponente kontrolliert und koordiniert nun diese Transaktionen. Dazu gehört es unter anderem relevante Daten an die entsprechenden Engines zu senden und diese darüber zu informieren, dass eine Aktion ausgeführt werden soll.

7. Am unteren Ende des Index Servers befinden sich in Abbildung 1 noch zwei weitere Komponenten. Die erste davon ist der *Persistance Layer*. Diese Komponente ist dann wichtig, wenn die HANA entweder geplant oder ungeplant neu gestartet wird. Deswegen werden hier alle 5 bis 10 Minuten [7] so genannte Save Points erstellt um bei dem Neustart einen relativ aktuellen Stand der HANA wieder herstellen zu können. Außerdem ist eine Kommunikation zwischen dem *Persistance Layer* und dem *Tran-*

saction Manager wichtig, um bei einem Neustart die Atomarität der Datenbank zu gewährleisten.

8. Die letzte Komponente, die in Abbildung 1 dem Index Server zuzuordnen ist, ist der *Disk Storage*. Diese Komponente ist traditioneller Festplattenspeicher, der als Speicher für alte Daten genutzt werden kann, die nicht mehr benötigt werden oder als Bach-up Speicher für den Fall eines Desasters oder anderen Zwischenfalls.

3.3 XS Engine

Neben dem Index Server sind in Abbildung 1 auf der Serverseite noch die *SAP HANA Extended Application Services* zu sehen, welche oftmals mit XS Engine abgekürzt werden. Die Idee hinter der XS Engine ist es, einen voll funktionsfähigen Anwendungsserver, Webserver und eine Entwicklungsumgebung in die SAP HANA Anwendung zu integrieren. Obwohl die Abbildung vermuten lässt, dass die XS Engine lediglich eine Erweiterung der HANA darstellt, ist die XS Engine vollständig in die HANA integriert und hat dadurch auch direkten Zugriff auf einige der Komponenten der HANA Datenbank. Aus dieser Integration des Anwendungsservers direkt in die Datenbank hat sich ein Paradigmenwechsel ergeben, der in Abbildung 2 dargestellt ist.

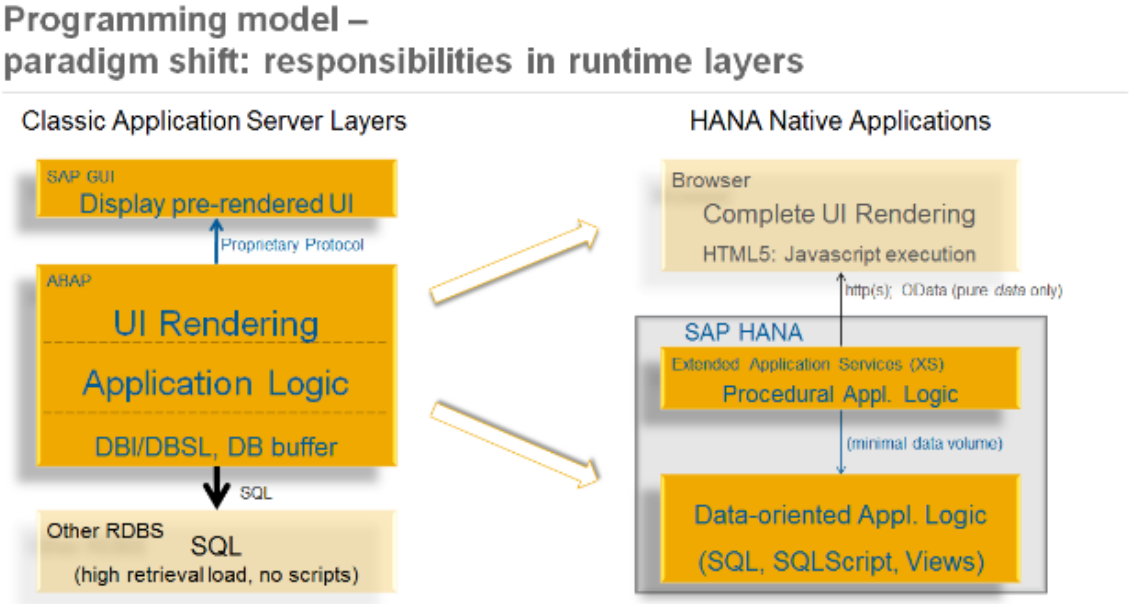


Abbildung 2: Paradigmenwechsel durch die Einführung der XS Engine, Quelle: nach [5]

Bei traditionellen Webanwendungen, in Abbildung 2 auf der linken Seite dargestellt, wurde neben der Datenbank noch ein separater Anwendungsserver benötigt. Dieser Anwendungsserver hat die Logik der Anwendung implementiert und bei Bedarf per SQL Anfragen an die Datenbank getätigt. Durch die Integration des Anwendungsservers in die Datenbank, in der Abbildung auf der rechten Seite dargestellt, fällt dieser zusätzliche Anwendungsserver weg. Auf der Seite des Clients wird nur noch das Rendering der Oberfläche durchgeführt. Die ganze Anwendungslogik, die mit der Datenbank zu tun hat wir dann direkt in der HANA ausgeführt. Daraus entstehen zwei Vorteile: Einfachheit - die Anzahl der Schichten in einer Anwendung wird weniger, wodurch es einfacher und übersichtlicher wird eine Anwendung zu entwickeln und deployen; und Performance - dadurch, dass die XS Engine in die HANA integriert ist, kann sie nicht nur SQL Statements nutzen, sondern auch andere, optimierte Verfahren wie SQLScripte oder Views.[14]

Abbildung 3 zeigt, wie eine Anwendung aufgebaut ist, die die XS Engine nutzt und welche Programmiersprachen an welcher Stelle verwendet werden.

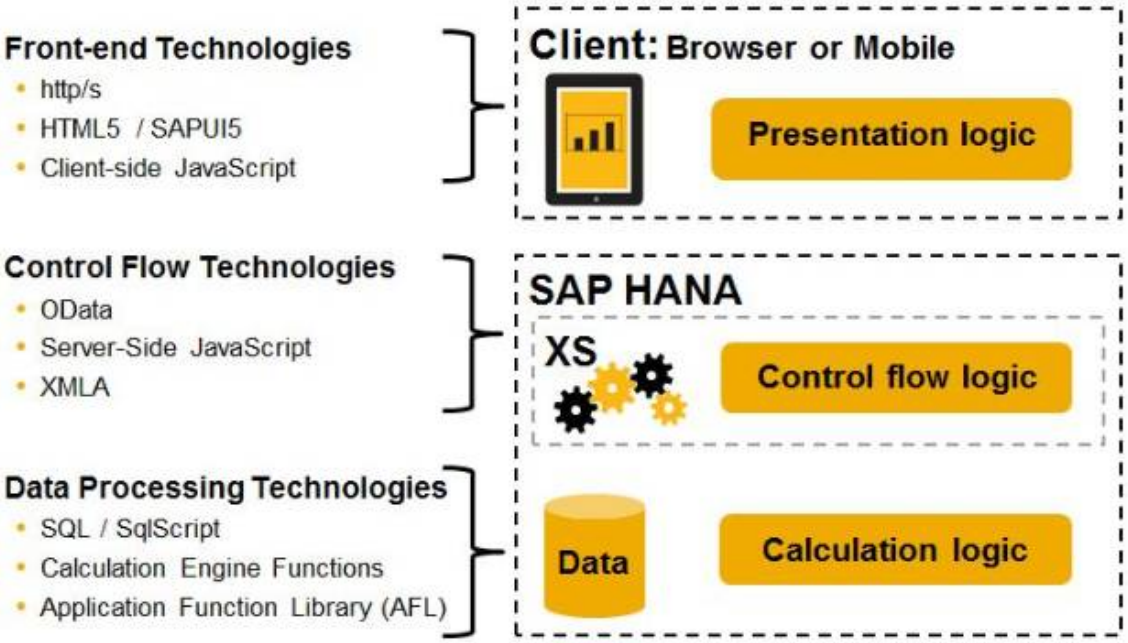


Abbildung 3: Programmiermodell durch die Einführung der XS Engine, Quelle: nach [14]

Das Front-end soll im Browser (oder einem mobilen Gerät) durch HTML5 und clientseitiges JavaScript realisiert werden. Der Großteil der Anwendungslogik und vor allem der Berechnung von Daten oder andere datenintensive Operationen sollen in der Datenbank per SQL, SQLScript oder Funktionen der Calculation Engine direkt in der Datenbank

durchgeführt werden. Der XS Engine bleibt dadurch nur noch die Aufgabe, diese beiden Bereich zu verbinden. Die Technologien, die für diese Aufgabe verwendet werden sind OData, serverseitiges JavaScript und XMLA.

4 SAP HANA Studio als Entwicklungsumgebung

Eng verbunden mit der XS Engine, die in Abschnitt 3.3 vorgestellt wurde, ist das *SAP HANA Studio*; eine auf Eclipse basierende und von SAP vertriebene Entwicklungsumgebung, die entwickelt wurde um einfach alle nötigen Ressourcen für die Anwendungsentwicklung mit der HANA bereitzustellen und zu verwalten.

Dafür stellt das SAP HANA Studio Perspectives für die 4 Bereiche zur Verfügung, die benötigt werden um effektiv mit der HANA arbeiten zu können.[12]

1. Information Modeler
2. Datenbankentwicklung & Datenbeschaffung/-versorgung
3. Administration, Überwachung, Sicherheit
4. Anwendungsentwicklung

Da das SAP HANA Studio auf Eclipse basiert, kann durch das Herunterladen von mehreren Plug-Ins für Eclipse die selbe, aber kostenfreie Funktionalität zur Verfügung gestellt werden. Im folgenden werden zwei „Perspectives“ vorgestellt, die für die Anwendungsentwicklung und somit für die Projektgruppe interessant sind.

4.1 Modeler

Die Modeler-Perspective kann dazu genutzt werden einfache Attribute- oder Analytic Views zu erstellen, aber auch komplexe Calculation Views, die wiederum SQL und SQL-Script basierte Befehle oder Prozeduren verwenden. Die erstellten Views können dann gegebenenfalls auch direkt durch eingebaute Reporting Tools visualisiert werden.

Hierbei ist wichtig zu erwähnen, dass die Modeler Perspective nur auf die HANA zugreift. In dieser Perspektive können keine Java Scripte oder ähnliches geschrieben werden, die die Daten aus der Datenbank nach draußen tragen. Auch die Views werden nur innerhalb der HANA erstellt und dort berechnet. Dabei stehen den Berechnungen die in Abschnitt 2 erwähnten multiplen Rechenkerne zur Verfügung.

4.2 Development

Die zweite wichtige Perspective ist die Development Perspective. Innerhalb dieser Perspective gibt es wiederum 3 Views, die genutzt werden können um eine Anwendung zu entwickeln.

4.2.1 System View

Über die System View wird die Verbindung zu einer HANA Datenbank hergestellt. Nach der erfolgreichen Verbindung und Authentifizierung mit einer HANA stehen dem Nutzer in dieser View die gleichen Möglichkeiten zur Verfügung, wie in der Modeler Perspective. Der Nutzer kann also beispielsweise verschiedene Tabels anlegen oder diese in unterschiedlichen Views manipulieren.

4.2.2 Project Explorer-View

Die zweite View, die in der Development Perspective zur Verfügung gestellt wird, ist die Project Explorer View, welche den lokalen Workspace darstellt. In dieser View kann, wie es von anderen Entwicklungsumgebungen bekannt ist, die Anwendung programmiert und gedebugged werden. Im Grunde genommen könnte hier eine traditionelle Java Anwendung programmiert werden, die per JDBC auf die Datenbank zugreift. Wenn allerdings die Vorteile der XS Engine genutzt werden sollen, stehen dem Entwickler hier neue Projekt- und Dateitypen zur Verfügung, die speziell für die XS Engine entwickelt wurden. So steht einem Entwickler beispielsweise der neue Dateityp .xsjs zur Verfügung, der sich ähnlich wie JavaScript verhält, allerdings *nicht* auf der Clientseite ausgeführt wird, sondern serverseitig. Die Vorteile davon sind, dass das Script näher an den eigentlichen Daten liegt, die Zugriffszeiten also kürzer sind, und dass wieder die hohe Rechenleistung der HANA genutzt werden kann.

4.2.3 Repository View

Die dritte und vielleicht wichtigste View, die die Development Perspective zur Verfügung stellt, ist die Repository View. Das Repository stellt das Bindeglied zwischen den lokal angelegten Projekten und der HANA dar. Innerhalb dieser Ansicht können mehrere *Repository Workspaces* erstellt werden, die sich jeweils auf eine HANA Datenbank beziehen. Innerhalb dieser Workspaces werden dann die Pakete angezeigt, die dem Entwickler für die angegebene HANA zur Verfügung stehen.

Einem Entwickler stehen nun zwei Wege zur Verfügung, wie er sein Projekt mit dem Workspace Repository verbinden kann. Hat der Entwickler bereits ein Projekt angelegt, kann er dieses mit dem Repository „sharen“. Dafür muss er in einem Auswahlmenü die gewünschte HANA auswählen und kann sich direkt mit dieser verbinden. Die andere Möglichkeit ist es, ein Projekt aus dem Repository in den lokalen Workspace zu importieren. Dafür muss der Entwickler in der Repository View das gewünschte Paket auswählen und kann dieses dann per „Check Out and Import Projects“ in den lokalen Workspace importieren.

Wenn der Entwickler dann weiter an dem Projekt arbeitet kann er die veränderten Dateien aktivieren. Durch das aktivieren werden die Daten auf die HANA gespielt und können dort verwendet werden.

Interessant wird das Repository auch dann, wenn mehrere Entwickler gleichzeitig an einer Anwendung arbeiten. Jeder der Entwickler kann sich mit dem Repository verbinden und hat dadurch die gleichen Dateien zur Verfügung. Auch dabei ist das aktivieren von veränderten Dateien wichtig, denn die anderen Entwickler können nur auf aktivierte Dateien, also die, die auch wirklich von der HANA ausführbar sind, zugreifen. Sollten zwei Entwickler gleichzeitig an einer Datei arbeiten stellt das Repository auch Methoden zum mergen von Dateien zur Verfügung. Außerdem soll laut [12] ein „version management“ vorhanden sein, welches zwar in gewisser Weise auch vorhanden ist, aber nicht so, wie es von beispielsweise Git bekannt ist.

5 Fazit

Abschnitt 3 hat gezeigt, dass es im wesentlichen zwei Möglichkeiten gibt, mit der HANA zu kommunizieren. Auf der einen Seite gibt es die traditionelle Kommunikation per Treiber wie JDBC und auf der anderen Seite gibt es die Möglichkeit die in die HANA integrierte XS Engine zu nutzen. Die zweite Option bietet jedoch gegenüber der ersten Performancevorteile, weil unter anderem die Scripte, die Datenbankzugriffe fordern, näher an der Datenbank liegen. Da es das Ziel der Projektgruppe ist in quasi Echtzeit Daten zu analysieren, bietet es sich an, die zweite Möglichkeit und damit auch den vollen Umfang der SAP HANA Technologie zu nutzen.

Als Entwicklungsumgebung bietet es sich an, das SAP HANA Studio, beziehungsweise die korrespondierenden Eclipse Plug-Ins zu nutzen, da diese dafür konzipiert wurden mit der HANA zu kommunizieren. Dadurch ist es sehr einfach und überschaubar Scripte in die HANA zu laden und diese dort zu nutzen. Auch das eingebaute Repository bietet den Vorteil, dass keine weitere Software installiert werden muss um die Daten zu verwalten, auch wenn das Repository hier und dort noch einige Schwachstellen hat.

References

[1] Courtney Bjorlin. *SAP Begins BW on HANA Ramp-Up, First Big Test for the HANA Database*. 2011. URL: http://www.asugnews.com/article/sap-begins-bw-on-hana-ramp-up-first-big-test-for-the-hana-database.

[2] Doug Henschen. *SAP Launches Cloud Platform Built On Hana*. 2012. URL: http://www.informationweek.com/applications/sap-launches-cloud-platform-built-on-hana/d/d-id/1106889?.

[3] ITWissen. *Zugriffszeit*. 2015. URL: http://www.itwissen.info/definition/lexikon/Zugriffszeit-access-time.html.

[4] Prasad Illapani. *Third Party ETL Tool Certification Program for SAP HANA*. 2012. URL: https://blogs.saphana.com/2012/12/12/third-party-etl-tool-certification-program-for-sap-hana/.

[5] Thomas Jung. *SAP HANA Extended Application Services*. 2012. URL: http://scn.sap.com/community/developer-center/hana/blog/2012/11/29/sap-hana-extended-application-services.

[6] Chris Kanaracus. *The in-memory analytic appliance will compete with next-generation data-processing platforms such as Oracle's Exadata machines*. Dec 1, 2010. URL: http://www.infoworld.com/article/2624847/database/sap-launches-hana-for-in-memory-analytics.html.

[7] Nicholas Chavez Kay Somers and Mike Lampa. *SAP HANA – Core Architecture*. 2012. URL: http://en.community.dell.com/techcenter/b/techcenter/archive/2012/09/28/sap-hana-core-architecture.

[8] Jeff Kelly. *Primer on SAP HANA*. 2013. URL: http://wikibon.org/wiki/v/Primer_on_SAP_HANA.

[9] Saurav Mitra. *SAP HANA Architecture*. 2012. URL: http://www.dwbiconcepts.com/database/28-hana/105-sap-hana-architecture.html.

[10] SAP News. *SAP and Partners Set New Record for World's Largest Data Warehouse*. 2014. URL: http://www.news-sap.com/sap-and-partners-set-new-record-for-worlds-largest-data-warehouse/.

[11] SAP. *SAP HANA Developer Guide*. 2014.

[12] SAP. *SAP HANA Studio - Overview*. 2013.

[13] SAP. *SQL Database Connectivity*. URL: `http://maxdb.sap.com/documentation/sqldbc/SQLDBC_API/`.

[14] Ronald Silberstein. *Introducing...SAP HANA Extended Application Services (XS)*. 2013. URL: `http://scn.sap.com/docs/DOC-60322`.

Abschließende Erklärung

Ich versichere hiermit, dass ich meine Hausarbeit selbständig und ohne fremde Hilfe angefertigt habe, und dass ich alle von anderen Autoren wörtlich übernommenen Stellen wie auch die sich an die Gedankengänge anderer Autoren eng anlegenden Ausführungen meiner Arbeit besonders gekennzeichnet und die Quellen zitiert habe.

Oldenburg, den 9. März 2015 Nils Worzyk

Protokolle

Nach jeder Sitzung, die Montags zwischen 12:00 - 14:00 statt gefunden haben, wurde ein Protokoll erstellt, in dem die wesentlichen Punkte der Sitzung zusammengeführt wurden. Für die Anfertigung der Protokolle waren die Teilnehmer der Projektgruppe zuständig.

Sitzungsprotokoll für KW 43

Sitzungsdatum: Montag, 20.10.2014
Uhrzeit: 12:00 - 14:00
Vorsitzende(r): Anwesende PG Betreuer
Protokollant(in): Christian Janßen
Abwesende(r): Milan Tomovic

Ablauf / Besprechungsergebnisse

- Zunächst Vorstellung der anwesenden PG-Betreuer sowie der anwesenden PG Mitglieder
- Im Anschluss Start der Folienpräsentation :
  - Festlegung der Protokollanten sowie der Vorsitzenden im heutigen Treffen sowie für die Zukunft
  - Dabei wechselt Protokollant und Vorsitzender rythmisch
  - (Protokollant von letzer Woche ist in der darauffolgenden Woche Vorsitzender)
  - Vorstellung der Grundidee, Termine sowie deren Laufzeit und PG Phasen
  - PG Phasen setzen sich zusammen aus Seminarphase, Entwurfsphase und Entwicklungsphase
  - Nahelegung von Social Events um die Gruppendynamik zu stärken
  - Erläuterung der Notenzusammensetzung sowie Vorgaben:
    - Mailvorgaben
    - Ferienregelung (Es wird auch in den Semesterferien gearbeitet, Maximal 3 Wochen Urlaub)
    - PG Raum (Frage wird geklärt)
  - Eläuterung der Projektgruppenidee
  - Interviews:

    - Spezifizierung der PG Ziele
    - Befragung erfolgt in drei Gruppen zu allgmeinen Zielen und folgenden speziellen Zielen (Jede Gruppe wählt 1 spzielles Ziel):
        - Use Cases
        - Plattform
        - Analysen

- Folgende Jobs wurden in der Projektgruppe vergeben:
    - Projektmanager: Kai
    - Stellvertreter: Kamiran
    - Dokumentationsbeauftrager: Olga
    - Finanzbeauftragter: Philipp
    - Webseitenbeauftragter: Jannes
    - Testbeauftragter: Milan
    - Serveradmin: Christian
    - Stelvertreter (2 Admin): Nils

Aufgabenverteilung nächste Woche

- Terminfindung für einen gemeinsamen Termin:
    - Projektmanagement setzt Doodle auf und verschickt Link an PG Mitglieder
    - PG Mitglieder tragen sich ein
- E-Mail-Verteiler muss eingerichtet werden
    - Stellvertreter Serveradmin kümmert sich drum (Arbi Jörg oder Olaf)
- Interviews müssen organisiert werden
    - Gruppenbildung
    - Jedes Mitglied macht sich Gedanken über Ziele und ein Spezialziel
- Serverangelegenheiten
    - Einarbeitung
    - Serveradmin macht sich schlau und kontaktiert Hans Hermann aus der WI Abteilung
    - Server für PM Tool, Webseite werden in absehbarer Zeit benötigt
    - Evtl. weiterer Server für Prototyp
- Finanzbeauftragter (Erstellt Liste mit möglichen Strafen/Abgaben etc.)

- Dokumentenbeauftragte (Macht sich über sinnvolle Vorlagen Gedanken (z.B. Protokoll, Stundenzettel, Sitzungsplanung))
- Webseitenbeauftragter informiert sich über Webauftritte (Evtl Mockup?)
- Projektmanagement erstellt Übergangsdokument mit wichtigen zu klärenden Fragen
  - Allgemeine Übersicht und Themen die wichtig zu besprechen sind, Hochladen in Cloud
  - Mitgleider ergänzen dieses Dokument für bessere Ideenfindung (Zur Erleichterung Vorsitzender)
- Projektmanagement PM Tool, Redmine wurde vorgeschlagen
- Jedes PG Mitglied macht sich in absehbarer Zeit Gedanken über ein Logo
  - Spezielle Kenntnisse in Photoshop sind willkommen
- Jedes PG Mitglied sendet E-Mail mit 3 Seminarthemen (nach Priorität) an PG Betreuer

Sitzungsprotokoll für KW 44

Sitzungsdatum: Montag, 27.10.2014
Uhrzeit: 12:15 - 13:30
Vorsitzende(r): Christian Janßen
Protokollant(in): Janine Haase
Abwesende(r): ———

Ablauf / Besprechungsergebnisse

Seminarthemen:

- PG-Betreuer geben Seminarthemenvergabe bekannt
- Olga und Milan erhalten Seminarthemen
- Thementausch zwischen Kai, Phillipp und Janine
- Vortrag im Januar, schriftliche Abgabe im Februar
- Die Themen sollen von den jeweiligen Verantwortlichen mit den PG-Betreuern durchgesprochen werden, um die Ziele und Vorstellungen des entsprechenden Themas aufzunehmen

Organisatorisches:

- Raum A04-3-319 ist bis Ende Oktober 2015 für die PG montags zwischen 12 und 14 Uhr reserviert
- Beginn der Treffen wird jeweils auf 12:15 festgelegt
- Kommunikation über Mail ist teilweise umständlich, die Gruppe soll auch untereinander kommunizieren und Absprachen treffen können, ohne dass die Betreuer alle ünwichtigenMMails hierzu erhalten, daher wird eine weitere Verteileradresse eingerichtet, in der Hauptverteileradresse bleibt Prof. Dr. Marx Goméz als Adressat enthalten.

- ftp-Server soll eingerichtet werden
- Urlaub ist in den Projektplan einzutragen, dazu sind die PG-Betreuer zu informieren, hier unbedingt auf den richtigen Mailvetreiler achten!
- Abwesenheiten: Wer komplett nicht an einer PG-Sitzung teilnehmen kann, muss sich von der Gruppe und den Betreuern abmelden, auch hier auf den richtigen Mailverteiler achten!
- Stundenzettel: 20 Wochenstunden sind durchschnitt und können nicht gleichmäßig auf die Wochen verteilt werden. Daher werden keine Stundenzettel verwendet.
- Janine Haase wird Kommunikationsbeauftragte.
- Die Wahl des Vorgehensmodells wird bis nach den Interviews zurückgestellt. Als Projektmanagementtool wird Redmine verwendet.
- Themenvorschläge für die Montagssitzungen müssen bis Donnerstagabend an den Vorsitzenden gesendet werden.

Finanzen:

- Die folgenden Strafgelder werden festgelegt:
  - Strafe für bis zu 15 Minuten zu spät erscheinen: 2 Euro
  - Strafe für über 15 Minuten zu spät erscheinen: 5 Euro
  - Strafe für unentschuldigtes Fehlen: 10 Euro/Flasche Havanna/Kiste Bier nach Wahl.
- Bei vorheriger Abmeldung wird kein Strafgeld fällig.

Website:

- eine eigene Website unabhängig von der Website der Uni OL ist leichter händelbar, daher wird eine eigene Website erstellt

Logo:

- Entwürfe für Logos werden vorgestellt
- der Entwurf von Jannis wird in die Cloud geladen, damit sich jeder noch einmal Gedanken dazu machen kann

Interviews:

- Timing sollte kurzfristig erfolgen, da darauf das Projekt aufbauen wird
- Bis Mittwoch, 29.10.2015 sollen Interviewtermine feststehen
- alle Interviews sollen innerhalb der nächsten 2 Wochen durchgeführt worden sein, in der übernächsten Woche sollen die Ergebnisse oder Teilergebnisse vorgebracht werden. Jede I-Gruppe wird einen eigenen Termin suchen und die Termine mit den PG-Betreuern eigenständig vereinbaren und durchführen.
- Der Fragenkatalog soll allgemeine und spezielle Fragen enthalten. Daher werden 3 Dokumente in die Cloud geladen, von allen eingesehen entsprechend ergänzt oder überarbeitet

Dokumente:

- sinnvolle Dokumente, z. B. eine Protokollvorlage werden in der Cloud zur Verfügung gestellt
- Dokument für Vorschläge zu Themen der Montagssitzungen wird eingestellt.

Seminarbeitsthemen:

- Christian Janßen: Potentiale von Big-Data
- Jannes Spekker: Datenanreicherung/ -ergänzung
- Kamiran Tizyani: Data Mining
- Milan Tomovic: Standardisierte Datenmodelle für Verkehrsdaten/ Ontologien
- Nils Worzyk: Entwicklungsumgebungen und Frameworks
- Olga Schwarz: Mobilitätsrelevante Sensorik im Verkehr
- Kai Hänig: Agiles Projektmanagement
- Phillipp Schuhmacher: Visual Analytics
- Janine Haase: Bedeutung von In-memory

Aufgabenverteilung nächste Woche

Kommunikation:

- Christian wird sich mit XMPP bezüglich einer verschlüsselten Nachrichten- und Dokumentenübermittlung beschäftigten.
- Nils wird einen zweiten Mailverteiler ohne Betreuer einrichten lassen für Einzelabsprachen der PG

Interviews:

- Kai wird Interviewgruppen einteilen und per Mail bekanntgeben.
- Interviewgruppen erstellen Fragenkataloge, organisieren einen Termin und führen die Interviews durch
- alle prüfen die Interviewfragen aller Gruppen und überarbeiten bzw. ergänzen diese gegebenenfalls

Serverangelegenheiten und Webpräsenz:

- Nils und Christian kümmern sich um die Organisation eines Servers
- Jannes wird sich um die Webpräsenz der PG auf einem eigenen Server kümmern
- Kai wird sich um den Redmine-Server kümmern

Dokumente:

- Olga wird sich um eine Vorlage für ein "Vorschlagsdokument" zur Montagssitzung kümmern.

Sonstiges:

- Jedes Mitglied wird sich über das Logo weitere Gedanken machen (Jannes-Vorlage Cloud)

Sitzungsprotokoll für KW 45

Sitzungsdatum: Montag, 03.11.2014
Uhrzeit: 12:15 - 12:50
Vorsitzende(r): Christian Janßen
Protokollant(in): Jannes Spekker
Abwesende(r): Galina Janusauskiene (unentschuldigt)

Ablauf / Besprechungsergebnisse

Neues Teammitglied:

- Galina nicht anwesend, auf KW 46 vertagt

Server I:

- Server läuft seit Ende KW 44
- Namensauflösung fehlt noch
- User werden bis Mitte der Woche angelegt
- Christian erstellt Präsentation für Server Handhabung zum nächsten Meeting
- Redmine Installation kann vorgenommen werden, Installationspakete vorhanden

Interviews:

- Vorgehen für Interviews mit Betreuern wird besprochen
- Termine wurden festgelegt
- es wird angemerkt, dass die einzelnen Fragebögen (Use-Case, Plattform, Analyse) unnötige doppelte Fragen enthalten
- Leitfäden für Interviews werden im Anschluss an Meeting erstellt
- als Raum steht A04 3-319 bereit

Logo:

- keine weiteren Vorschläge wurden eingereicht
- Vorschlag von Jannes wird angenommen

Dokumente:

- Themenvorschlag-Vorlage wurde noch nicht erstellt, soll zur nächsten Sitzung vorliegen

Webseite:

- Jannes stellt erstelltes Design und CMS für Webseite vor
- Diskussion über mögliche Veränderungen: Logo wirkt zu groß
- Vorschlag 1: Text unter Logo horizontal ausrichten, dadurch Platzersparnis
- Vorschlag 2: Symbole aus Navigation entfernen, Logo in Navigationsleiste schieben
- WordPress wird als CMS genutzt

Server II:

- Christian stellt ein paar grundlegende Server Funktionen / Clients vor
- Jabber als Kommunikationsplattform
- Vorschlag: Client "jitsi", Erläuterung von Registrierung, Client-Installation, GUI
- SSH Client wird benötigt
- Vorschlag: Client "PuTTY", Erläuterung von Login und Konsole
- FTP-Server wurde aufgesetzt
- Vorschlag: Client "WinFTP", läuft nur unter Windows, Erläuterung von Login und GUI
- Vorschlag 2: Client "FileZilla", läuft unter Windows und Mac OS

Schlussbemerkungen:

- Termin für teaminternes Meeting wird noch nicht angesetzt
- Nachfrage zu Leitfaden: kurze Erklärung von leitfadengestützten Interviews durch Betreuer

Aufgabenverteilung nächste Woche

Interviews:

- Durchführung der Interviews
- Zusammenfassung der Interviews

Server:

- Anlegen von Usern auf Server

Projektmanagement:

- Installation von Redmine auf Server

Dokumente:

- Erstellung von Themenvorschlags-Vorlage

Webseite:

- Überarbeitung der Webseite nach Vorschlägen aus Meeting

Sitzungsprotokoll für KW 46

Sitzungsdatum: Montag, 10.11.2014
Uhrzeit: 12:15 - 13:00
Vorsitzende(r): Jannes Spekker
Protokollant(in): Kai Hänig
Abwesende(r): Benjamin Wagner vom Berg, Daniel Stamer, Galina Janusauskiene (unentschuldigt)

Ablauf / Besprechungsergebnisse

Anwesenheit

- Olga und Milan kommen ca. 2 Minuten zu spät

Neues Teammitglied

- nicht erschienen, eine Teilnahme an der Projektgruppe ist nun für Galina nicht mehr möglich

Interviews:

- Die durchgeführten Interviews der Vorwoche werden von den einzelnen Gruppen vorgestellt.
- Es wird ein einheitliches Dokument erstellt um die Zusammenfassung der Interviewbögen zu erstellen. Dies wird von Janine erstellt

Fallstudie:

- SAP Hana Fallstudie - Es wird direkt im Anschluss an die Sitzung ein Termin festgelegt. und den Betreuern mitgeteilt um den Raum zu reservieren.
- Nachtrag: Der Termin wurde auf Dienstag, den 18.11.14 von 14-18 Uhr gelegt.

Administration:

- Hinweis auf die Erklärung zur Installation von Putty und Filezilla wurde per Mail an alle Teilnehmer geschickt. Eine zweite Mail enthält Passwörter und Nutzernamen.
- Von nun an soll nur noch der FTP-Server genutzt werden
- Redmine wurde durch Admins aufgesetzt, die Konkretisierung einzelner Gruppen und Aufgaben sowie eine Einarbeitung aller Teammitglieder erfolgt jedoch erst, wenn wir in die Ausarbeitungsphase des Projektes übergehen.

Dokumentation

- Neue Themenvorschlagsvorlage soll erstellt werden, sodass der Sitzungsleiter die Themen vorab rumschicken kann. Olga wird dies dann auf den Server legen.
- Jannes lädt das Logo ebenfalls auf den Server
- Neue Protokollvorlage wird in Word erstellt.

Webseite:

- Website, CMS nicht aufsetzbar, da keine Root Rechnte. Leichte Anpassungen der Website durch Jannes. Die Seite soll dann mit der Zeit wachsen und angepasst werden.

Nächsten Schritte:

- Labor muss mit der zweiten PG Gruppe abgesprochen werden. Janine als Kommunikationsbeauftragte soll sich darum kümmern. (Raum A2-2-241)
- Termine festlegen mit den Betreuern, besonders bei Manuel, sollten diese Woche noch Termine gemacht werden um Seminarthemen abzustimmen und mit der Bearbeitung zu beginnen.
- Agiles Projektmanagement betreut jetzt Daniel

Zusätzliches:

- Hasso-Plattner-Institut Anmeldung um Zugriff auf deren SAP Hana zu bekommen. Bis April. Projektskizze erstellen und einreichen. Als Alternative zu unserer eigenen Hana.

Aufgabenverteilung nächste Woche

Alle:

- Einrichten von Putty (Windows) oder mit Hilfe des Terminals (Mac)
- Filezilla downloaden und einrichten
- Anmeldung auf Redmine durchführen
- Termine festlegen mit Betreuern der einzelnen Seminarthemen.

Dokumentationsverantwortliche:

- Erstellung der Themenvorschlagsvorlage
- Erstellung der Protokollvorlge

Kommunikationsbeauftragte:

- Abstimmung der Nutzung des Labors (Raum A2-2-241) mit der zweiten PG am Lehrstuhl.

Sitzungsprotokoll für KW 47

Sitzungsdatum: Montag, 17.11.2014
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Jannes Spekker
Protokollant(in): Kamiran Tizyani
Abwesende(r): Kai Hänig (entschuldigt) und Milan Tomovic (entschuldigt)

Ablauf / Besprechungsergebnisse

Jannes übernimmt die Moderation für Kai

Team:

- Galina ist neu in die PG-Gruppe eingetreten.
- PG-Betreuer und die anwesenden PG-Mitglieder stellen sich vor.
- Galina muss auf dem neusten Stand gebracht werden.
- Seminarthema erhält sie von den PG-Betreuern.
- Jannes erläutert kurz den aktuellen Stand für Galina:
  - Server
  - Interviews
  - Themen Abgabe und Vorstellung

Fallstudie

- Hinweis auf den Termin für die Erarbeitung der Fallstudie (18.11.14 von 14-18 Uhr; Ort: A4-3-321)
- Dient der Einarbeitung in SAP-Hana
- Gruppeneinteilung erfolgt in der Übung

Systeme / Server:

- Kai hatte Probleme mit Putty auf seinen McBook.
- Nils hat bei der Lösung des Problems unterstützt.
- PG-Betreuer bekommen auch einen Zugang, für den Server.
- Jannes bekommt zusätzliche Rechte für den Server, um ein CMS zu installieren.

Redmine:

- Milan, Olga, Phillipp, Kamiran wurden auf die Anmeldung hingewiesen.
- Betreuer erhalten ebenfalls einen Zugang.

Olga hat eine neue Vorlage für das Protokoll auf dem Server zur Verfügung gestellt. In der Zukunft achten, dass die Emails nicht doppelt gesendet werden. Gruppenmitglieder wurden hingewiesen, Termine für die Seminare festzulegen. Abstimmung der Labornutzung mit anderer Projektgruppe.

- Janine hat Kontakt mit der Parallel laufenden Gruppe aufgenommen.
- Die Räume stehen uns an folgenden Tagen zur Verfügung;
  - Donnerstag von 08 – 16 Uhr
  - Freitag steht uns der Raum ganztägig zur Verfügung

Janine hat die Interviews in einen Dokument zusammengefasst und stellt es auf dem Server zur Verfügung.

Sitzungsprotokoll für KW 48

Sitzungsdatum: Montag, 24.11.2014
Uhrzeit: 12:15 - 13:45
Vorsitzende(r): Kamiran Tizyani
Protokollant(in): Milan Tomovic
Abwesende(r): Janine Haase (entschudigt)

Ablauf / Besprechungsergebnisse

Team:

- Galina: Frage, ob sie sich mit dem Sever und den Zugriffsrechten vertraut gemacht hat. Ist in Bearbeitung.
- Neue Aufgabe für Galina: Eventbeauftragte (Cebit Auftritt oder öffentlich wirksame Auftritte)
- Wenn man zukünftig krank sein sollte: Mail an Betreuer durch den PG-Verteiler.

Server / Cloud

- Beim Server nicht angemeldet: Galina, Milan und Olga.
- Server Adresse: Rapid.informatik.uni-oldenburg.de/redminie
- Cloud Ordner gibt es einen neuen Link. Soll nur der Server genutzt werden oder auch die Cloud?
- Abstimmung: Server, Cloud oder beides? - Beides wird beibehalten.

Seminarthemen

- Haben sich alle bei den Betreuern gemeldet bezüglich den Seminarthemen?
- Galina hat noch kein Thema gewählt.

- Interessante Literatur auch für andere Gruppenmitglieder oder evtl. interessant für alle Mitglieder sollten der Gruppe mitgeteilt werden

Formatvorlage bzgl. Präsentation

- Die End-Dokumentation sollte von der Uni Formatvorlage genutzt werden.
- Für interne Präsentationen ist kein bestimmtes Format notwendig.
- Zitierweise: Genormte Zitierung nutzen. Sollte mit dem Dokumentenbeauftragten besprochen werden.
- Latex sollte für die Dokumentation genutzt werden.

Daten

- Rückmeldung von der VBG, keine Ressourcen. Die VBG Daten können genutzt werden, wurden aber nicht zur Verfügung gestellt. Vielleicht VBN Daten.
- Bis Weihnachten sollte das Okay da sein, jedoch werden keine Daten da sein.

Hana

- Wann können wir ins SAP Labor und wie können wir da was testen?
  - Zugänge können jeder Zeit bereitgestellt werden, sodass man auch von zuhause arbeiten kann.
  - Eclipse Software gibt es auch als Plug-In für die Hana.

Aufgabenverteilung nächste Woche

- Aufgabe für alle:
  - Seminarthemen bearbeiten, Quellen finden etc.
  - Aktuellen Stand vorstellen des jeweiligen Seminarthemas. (muss abgestimmt werden, nicht verpflichtend – wie sinnvoll?)
  - SAP Hana Dokumentation ansehen – liegt auf dem Server.
- Vorträge:
  - GIT Vorstellung (Nils KW 49)
  - Vorstellung der Einstellung des Eclipse Plug-in's. (Jannes KW 49)
  - Einführung in Redmine – Live-Vorstellung (Kai KW 49)
  - R Vorstellung in Zusammenhang mit SAP (Philipp KW 51)
  - Latex Präsentation und Formatvorlage (Olga KW 50)

- Dokumentation Quberunner sollte gesichtet werden (KW 51).
- Christian gibt Themen vor und teilt die folgenden Personen ein:
- Milan, Galina, Kamiran und Janine werden dies bearbeiten.

Sitzungsprotokoll für KW 49

Sitzungsdatum: Montag, 01.12.2014
Uhrzeit: 12:15 - 13:00
Vorsitzende(r): Milan Tomovic
Protokollant(in): Nils Worzyk
Abwesende(r): Galina Janusauskiene war 4 min zu spät

Ablauf / Besprechungsergebnisse

Anmeldung bei Redmine

Es wurde festgestellt, dass sich Olga bei Redmine angemeldet hat und Milan noch fehlt

Aufgabenverteilung zu dem Vorgängerprojekt "Cuberunner"

Es wurde festgestellt, dass die Aufgabenverteilung zu den Abschnitten des Vorgängerprojekts "Cuberunner" fertig gestellt wurde.

Vorstellung von Redmine durch Kai

Kai hat das Programm "Redmine" anhand einer Live-Demonstration vorgestellt. Dabei zeigt die Startseite des Programms für jeden Benutzer an, welche Tickets er derzeitig zu bearbeiten und welcher er selber erstellt hat. Als nächstes wurde der Programmreiter "Meine Seite" vorgestellt, welcher beispielsweise durch einen Kalender oder weitere Widgets personalisiert werden kann. Vor allem für die spätere Projektorganisation wurde dann das Ticketsystem von Redmine vorgestellt. Der Projektreiter "RAPID" stellt das Oberprojekt dar, das gesamte Projekt, und soll hauptsächlich globale Themen behandeln. Innerhalb dieses Oberprojektes sollen dann Unterprojekte erstellt werden, welche in einzelnen Reitern dargestellt werden. Für ein jedes solches Unterprojekt wird von dem Programm protokolliert, wer was wann gemacht hat und welche Tickets für dieses Unterprojekt existieren. Tickets wiederum stellen die Aufgaben dar. Dabei muss beispielsweise eingestellt werden, welchen Typ das Ticket hat und wie die Beschreibung für das Ticket aussieht. Es wurde hervorgehoben, dass diese Beschreibung ganz genau sein muss und nicht zweideutig sein

darf! Als letztes wurde in dem Vortrag der Reiter ”Gantt-Diagramm” erläutert, durch welchen das gesamte Projekt als Gantt-Diagramm ausgegeben werden kann und der Reiter ”Wiki” wurde vorgestellt, welches es zunächst noch einzurichten gilt und im späteren Verlauf dazu genutzt werden soll um How-To’s, allgemeine Infos oder andere Informationen an die Gruppe weiterzuleiten. Die Folien für den Vortrag sind auf dem Server einsehbar.

Vorstellung von dem Programm Git durch Nils

Die Folien für die Präsentation sind auf dem Server verfügbar.

Weihnachtsmarkt

Es wurde besprochen, dass ein generelles Interesse, von sowohl der Gruppe als auch den Betreuern vorhanden ist, ein gemeinsames Treffen auf dem Weihnachtsmarkt zu organisieren. Um einen Termin zu finden soll ein Doodle erstellt werden, welches für die Woche ab dem 16. gilt, da ein Großteil der Betreuer in der Woche davor keine Zeit hat.

Aufgabenverteilung nächste Woche

- Nils: Wird eine Doodle-Umfrage erstellen, mit dem Ziel einen Termin für das Treffen auf dem Weihnachtsmarkt zu finden.
- Olga: Wird in der kommenden Woche einen Vortrag über die Benutzung von Latex halten.
- Kamiran: Wird einen Teil der Dokumentation des Projektes ”Cuberunner” vorstellen.
- Janine: Wird einen Teil der Dokumentation des Projektes „Cuberunner“ vorstellen.a

Sitzungsprotokoll für KW 50

Sitzungsdatum: Montag, 08.12.2014
Uhrzeit: 12:15 - 13:45
Vorsitzende(r): Nils Worzyk
Protokollant(in): Olga Schwarz
Abwesende(r): Milan Tomovic (unentschuldigt), Galina Janusauskiene (unentschuldigt)

Ablauf / Besprechungsergebnisse

Kurze Einführung von Nils:

- Wiederholung der letzten Sitzung
- Vorträge auf den Server; Ordner verfügbar

Vortrag Olga – Einführung in Latex

- Inhalt des Vortrags
  - Bilder einbinden /verweisen
  - Tabellen erstellen
  - Mathematische Formeln
  - Quellcode
  - Literaturverzeichnis und zitieren
- Einigung auf backslash
- ” usepackagesubfiles” einbinden, um nur die jeweiligen Kapitel zu kompilieren. Nicht unbedingt das ganze Hauptdokument
- Jeder schreibt seine Ausarbeitung in Latex und benutzt das Template von der Homepage (http://vlba.wi-ol.de/22199.html), um ein einheitliches (corporate) Design einzuhalten.

- Für die Präsentation wird empfohlen ebenfalls das Template der Abteilung zu nutzen.

Vortrag Kamiran - Cuperunner: Zusammenfassung

- Inhalt: CeWe Fachkonzept; Data mining

Vortrag Janine - Cuperunner: Zusammenfassung

- Inhalt des Vortrags: Seminarthemen der letzten PG zusammengefasst
- Wichtig für Rapid könnte sein:
  - Knowledge Database Fayyad
  - Semantische Schicht
  - In Memory Vergleich (Janines Vortagsthema)
- Regelmäßige Tests sind sehr wichtig auch für unsere PG (Rückschluss aus der Cuperunnerdoku)

Aufgabenverteilung nächste Woche

- Weitere Vorträge über die PG ‚Cuperunner' : Milan, Christian, Galina
- Der Vortrag von Phillip (R-Language) fällt aus!
- Olga: Rapid Latextemplate bearbeiten, an das Template der Abteilung anpassen!
- Vorträge von Kai und Jannes werden zur Weihnachtszeit auf den Server geladen und für alle verfügbar gestellt.
- Protokoll von Milan fehlt! Muss noch online gestellt werden

Sitzungsprotokoll für KW 51

Sitzungsdatum: Montag, 15.12.2014
Uhrzeit: 12:15 - 13:45
Vorsitzende(r): Olga Schwarz
Protokollant(in): Philipp Schumacher
Abwesende(r): ———

Ablauf / Besprechungsergebnisse

Anwesenheit

- Galina fehlt: Benjamin will noch mal mit Galina reden
- Olga zu spät: 2Euro in die PG Kasse

Christians Vortrag zu Smart Wind Control (Auflistung der relevanten Abschnitte)

- Kennzahlen
- Wahrscheinlichkeiten
- Analytische/nicht analytische Anforderungen
- Architektur
  - u.a. Kettel,
  - R Language
  - (Philipp schaut sich die Einbindung von R in die HANA an (R-Integration))
  - Dazu insbesondere Kapitel 4.1 und 4.2 relevant

Anforderungsanalyse

- Anforderungsanalyse nach Marcel Siewerti anschauen

Sonstiges

- Termine für Seminare sollen vor Weihnachten feststehen
- Kassenstand wurde auf den Server geladen
- Weihnachtsmarkt: Treffen am Julius Mosel Platz
- Jänig Oldenburg: Betreuer sprechen morgen (16ter) mit Jänig

Sitzungsprotokoll für KW 02

Sitzungsdatum: Montag, 05.01.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Philipp Schumacher
Protokollant(in): Christian Janßen
Abwesende(r): Galina Janusauskiene (unentschuldigt), Milan Tomovic (entschuldigt)

Ablauf / Besprechungsergebnisse

Organisatorisches zu Beginn

Das Problem Galina wurde erneut angesprochen. Voraussichtlich ist sie zur nächsten Woche kein Bestandteil mehr dieser Gruppe. Zudem hat Philipp die Strafenliste aktualisiert und den Betrag angepasst. Von dem vorhandenen Geld wird eine Domäne finanziert.

Rückblick auf die vergangene Sitzung

Es wurde ein kleines Resümee der letzten Sitzung gezogen und ein kleiner Überblick über noch anliegende und fehlende Präsentationen gegeben. Einwände von Seiten der Gruppen gab es keine.

Installation

Zunächst wurde gefragt wie der aktuelle Stand mit der Installation der bereits vorgestellten Komponenten bspw. dem SAP Modeler ist. Da aber hier erst einige wenige sich mit dem Thema beschäftigt haben und die Seminararbeiten im Vordergrund stehen, konnten dazu keine weiteren Angaben und Ergebnisse erzielt werden. Bei einer Testweisen Installation von Philipp traten zudem Connect Probleme auf. Dies wurde auf die nächste Woche verschoben mit dem Verweis der Findung eines zusätzlichen Termins in der kommenden Woche. Des Weiteren wurde der Einsatz von kettle von Philipp vorgebracht. Allerdings sollten zuvor erst konkretere Anwendungsfälle erstellt werden bevor eine Softwareauswahl erfolgt.

Vorträge

n einem Vortrag stellte Philipp die R Language vor. Zudem behandelte sein Vortrag die Frage, ob der Einsatz im weiteren Verlauf des Projektes sinnvoll ist. Zudem wurde eine Architektur besprochen, die im Anschluss des Weihnachtsmarktes entworfen wurde. Dabei wurden folgenden Informationen ergänzt bzw. als konstruktive Kritik vorgetragen:

- Daten können direkt in die Hana DB geladen werden.
- Im Anschluss werden die Daten zusammengefasst und in der Hana gefiltert.
- Das Speichern der zusammengefassten Daten auf der Hana dient einem Vergleich mit Echtzeitdaten
- Überlaufende Daten, Daten die nicht mehr den Echtzeitaspekt erfüllen und den historischen Grand überspannt haben, werden in der Architektur, einer weiteren zusätzlichen DB gespeichert.

Planung

Der Vortrag von Milan wurde krankheitsbedingt in die kommende Woche verschoben. Zudem bieten die Betreuer an, bereits erstellte Seminarausarbeitungen und Präsentationen durchzuschauen. Im Rahmen der Datenbeschaffung warten wir weiter auf Informationen von Jähnig, da das Gespräch kurz vor Weihnachten stattgefunden hat und noch keine Antwort erfolgt ist.

Aufgabenverteilung nächste Woche

- Milan: Stellt seinen ausstehenden Vortrag über einen Teil der Cuberunner Dokumentation vor.
- Gruppe: Jeder macht sich weitere Gedanken über Anwendungsfälle. Die bereits besprochenen Anwendungsfälle vom 05.01.15 dienen dabei als Denkanstoß.
- Janine: Kopiert bereits besprochene Anwendungsfälle in ein separates Dokument.

Sitzungsprotokoll für KW 03

Sitzungsdatum: Montag, 12.01.2015
Uhrzeit: 12:15 - 12:35
Vorsitzende(r): Christian Janßen
Protokollant(in): Janine Haase
Abwesende(r): Galina Janusauskiene, Milan Tomovic

Ablauf / Besprechungsergebnisse

Anwesenheit

- Milan hat sich rechtzeitig krank gemeldet.
- Nils ist wie vorher angekündigt verspätet erschienen.
- Galina war nicht anwesend. Sie wird nach Rücksprache mit den Betreuern aus der Gruppe aussteigen und zu einem anderen Zeitpunkt eine andere PG besuchen.

Rückblick auf die vergangene Sitzung

Zum Protokoll von letzter Woche gab es keine weiteren Anmerkungen. Bislang steht die Antwort von Jähnig noch aus, wird laut Aussage der Betreuer jedoch kurzfristig folgen.

Vortrag Teilbereich Cuberunner von Milan

Aufgrund der Abwesenheit von Milan wird der Vortrag auf nächste Woche verschoben.

Planung

- Christian wird sich mit dem Plugin von Eclipse beschäftigen, wenn möglich, sollen sich alle anderen auch damit beschäftigen.
- Kai wird demnächst eine kleine Ausarbeitung bzw. Präsentation zu Redmine hochladen (wurde bereits gehalten).

- Die Besprechung von Anwendungsfällen, die nach der Sitzung in der letzten Woche begonnen wurde, soll nach der Sitzung fortgesetzt werden. In der Sitzung der nächsten Woche sollen diese kurz den Betreuern vorgestellt werden.
- Die Termine für die Präsentationen der Seminararbeiten liegen für einige aus der Projektgruppe an Klausurtagen. Aus dem Grund soll eine Mail an die Betreuer gesendet werden, in der die Termine, an denen die PG-Mitglieder keine Vorträge halten können, verzeichnet sind, damit die Termine ggf. angepasst werden können.
- Längere Abwesenheiten wegen Krankheit bedürfen einer Arbeitsunfähigkeitsbescheinigung. Diese müssen den Betreuern vorgelegt werden.

Aufgabenverteilung nächste Woche

- Milan: Milan wird seinen ausstehenden Vortrag über einen Teil der Cuberunner Dokumentation vorstellen.
- Christian: Wird sich in das Eclipse-Plugin weiter einarbeiten.
- Gruppe: Wenn möglich, sollte sich jeder aus der Gruppe mit dem Eclipse-Plugin beschäftigen.
- Janine: Wird die kurze Vorstellung der Anwendungsfälle in der nächsten PG-Sitzung vornehmen.
- Kai: Wird die Redmine-Informationen hochladen. Dazu wird Kai eine Info-Mail an die Betreuer senden, die Termine enthält, an denen die PG-Mitglieder keine Vorträge halten können.

Die aktuellen Informationen zur Abwesenheiten im Vortragszeitraum 04./05.02. lauten:
Janine: komplett abwesend Jannes: 04.02. Klausur - 14-16 Uhr
Christian: 05.02. Klausur - 10-12 Uhr
Phillip: 05.02. Klausur - 10-12 Uhr
Olga: kann immer
Milan: 05.02. Klausur - 10-12 Uhr
Nils: kann immer
Kamiran: 05.02. Klausur - 10-12 Uhr
Kai: kann immer

Sitzungsprotokoll für KW 04

Sitzungsdatum: Montag, 19.01.2015
Uhrzeit: 12:15 - 13:00
Vorsitzende(r): Janine Haase
Protokollant(in): Jannes Spekker
Abwesende(r): ——

Ablauf / Besprechungsergebnisse

Rückblick letzte Sitzung

- Kai hat Redmine Guide hochgeladen
- Jähnig: keine neuen Erkenntnisse
- Eclipse Plugin SAP HANA: Probleme beim Login, bei SAP HANA Modeller wird keine Verbindung hergestellt. Passwort verfällt nach sieben Tagen, wird neu angelegt.

Vorträge

- Anwendungsfälle: Vortrag durch Janine, Dokument wird von Janine bereitgestellt
- Vortrag durch Milan zu Jinengo (Teilbereich Cuberunner Projekt), Dokument wird auf Server bereitgestellt

Planung

- Handbuch zu Logins, Plugins, Software etc. soll angelegt werden
- Wird in Redmine-Wiki angelegt

Sitzungsprotokoll für KW 05

Sitzungsdatum: Montag, 26.01.2015
Uhrzeit: 13:15 - 13:35
Vorsitzende(r): Jannes Spekker
Protokollant(in): Kai Hänig
Abwesende(r):

Ablauf / Besprechungsergebnisse

Anwesenheit

- Olga ist wenige Minuten zu spät erschienen

Rückblick auf die vergangene Sitzung

- Zum Protokoll der vorherigen Woche gab es keine Anmerkungen der Gruppe

Beschluss der PG

- Es wurde sich auf eine Projektdomain geeinigt: www.rapid-ol.de
- Vorerst werden keine Gruppenaufteilungen der Teilnehmer vorgenommen. Bei Bedarf wird dies nachträglich durchgeführt.
- Einrichtung von Jabber / Jitsi (Mac OS X) für eine verbesserte Kommunikation zwischen den Projektteilnehmern während der Programmierphase. Telefonkonferenzen sind ebenfalls möglich.
- Urlaub wird wie folgt gehandhabt:
  - Ticketerstellung in Redmine. Titel = ‚urlaub' , Beginn und Ende Eintragen
  - Urlaub bei betreuern einreichen
  - Mind. 2 Wochen im vorraus
  - Brückentage = Arbeitstage

– Pro Arbeitstag werden 4 Arbeitsstunden veranschlagt.

Sitzungsprotokoll für KW 06

Sitzungsdatum: Montag, 02.02.2015
Uhrzeit: 12:45 - 13:15
Vorsitzende(r): Kai Hänig
Protokollant(in): Kamiran Tizyani
Abwesende(r): Janine hat Urlaub

Ablauf / Besprechungsergebnisse

Anwesenheit

- Jannes kam wenige Minuten später

System/Software:

- Alle Mitglieder des Projekts(Janine!) sind bei Hana Angemeldet
- Kai hat ein Backlog in Rapidminer eingerichtet
- Die Projektmitglieder haben entschieden Skype statt Jitsi, als Kommunikationsmittel zu nutzen, da es einfacher ist.

Uni-Veranstaltungen:

- Projektmitglieder die an Blockseminare teilnehmen, haben Sonderurlaub
  - Betriebliche Umweltinformationssysteme (Kai und Jannes)
  - Mobile Commerce(Christian)

Sitzungsprotokoll für KW 07

Sitzungsdatum: Montag, 09.02.2015
Uhrzeit: 12:15 - 13:00
Vorsitzende(r): Kamiran Tizyani
Protokollant(in): Milan Tomovic
Abwesende(r): Nils (Skype zugeschaltet)

Ablauf / Besprechungsergebnisse

- Anwesenheit
- Bei Fragen zu den Vorträgen direkt an die zuständige Person wenden
- Präsentationen in die Cloud hochladen
- Einheitliche Zitierweise nutzen. Evtl. die vom Fachbereich nutzen.
- Raum kann auch öfter genutzt werden, gegebenenfalls vorher anmelden.
- Präsentation von Janine (OLAP OLTP) für Details, die Präsentation verfügbar

Aufgabenverteilung nächste Woche

- Roadmap erstellen nächste Woche (Anforderungen)
- Erster Termin Mittwoch 15:00 Uhr BIB Gruppenraum 3.2
- Zweiter Termin Donnerstag 10:00 Uhr A04 3 319 ?

Sitzungsprotokoll für KW 08

Sitzungsdatum: Montag, 16.02.2015
Uhrzeit: 13:15 - 13:45
Vorsitzende(r): Milan Tomovic
Protokollant(in): Nils Worzyk
Abwesende(r): Olga Schwarz (entschuldigt), Christian Janßen (entschuldigt)

Ablauf / Besprechungsergebnisse

Gruppenaufteilung

- Wetterdatengruppe: Jannes hat eine PHP-Lösung entwickelt, mit Hilfe welcher die Wetterdaten vom Deutschen Wetterdienst automatisiert heruntergeladen und entpackt werden können. Nils möchte sich im Laufe der Woche um das automatisierte einladen von Daten in die SAP HANA kümmern.
- Allgemeine Datenrecherchegruppe: Bestehend aus Milan, Kamiran, Janine und Olga. Milan hat seine Ergebnisse der Recherche bei MDM (Mobilitäts Daten Marktplatz) vorgestellt.

Gruppenraum

Janine hatte sich mit der anderen Projektgruppe verständigt, dass uns der Gruppenraum am Montag und Donnerstag zur Verfügung steht. Der Schlüssel muss bei der Sekretärin abgeholt werden.

SAP HANA Repository

- Nils hat zwei Lösungen präsentiert, wie man die Versionierung des Projektes durchführen könnte
- Vorteile: Es ist bereits in dem SAP HANA Studio Eclipse Plugin vorhanden und bietet somit auf sehr einfachem Wege an mit der SAP HANA zu interagieren.

- Nachteil: Die Versionierung ist mangelhaft, was bedeutet dass es kompliziert ist eine gelöschte oder fehlerhafte Datei zurückzusetzen.
- Eine Git Lösung:
  - In den Quellen die Nils zu dem Thema gefunden hat, wird unter anderem davon ausgegangen, dass mit der Cloud Lösung der SAP HANA gearbeitet wird. Außerdem wird in den Quellen geschrieben, dass nur bestimmte Dateien, wie HTML und JavaScript Files über die Git Lösung versioniert werden können.
  - Vorteile: Wenn eine Datei aus versehen gelöscht oder fehlerhaft bearbeitet wurde, kann sie relativ leicht zurückgesetzt werden.
  - Nachteil: Es können nur bestimmte Dateien versioniert werden und es wird davon ausgegangen, dass mit einer Cloud Lösung gearbeitet wird.
- Nachfolgend gab es eine lange Diskussion darüber, welche der beiden Lösungen die sinnvollere wäre und es wurde überlegt, wie das Problem sonst noch zu lösen sei. Geendet ist die Diskussion damit, dass wir als Gruppe uns nochmal mit der Thematik auseinandersetzen sollten um eine sinnige Lösung zu finden.

Aufgabenverteilung nächste Woche

- Die einzelnen Gruppen arbeiten weiter an ihrem Themen.
- Nils möchte sich nochmal um das Repository kümmern, allerdings möglichst nicht alleine, damit die Sicht nicht einseitig ist und das Wissen verteilter. Ein weiterer, der sich darum kümmert stand zu diesem Zeitpunkt noch nicht fest.

Sitzungsprotokoll für KW 09

Sitzungsdatum: Montag, 23.02.2015
Uhrzeit: 13:15 - 13:45
Vorsitzende(r): Nils Worzyk
Protokollant(in): Olga Schwarz
Abwesende(r): Janine (entschuldigt), Kai (entschuldigt)

Ablauf / Besprechungsergebnisse

Gruppe Openstreetmap

- Anmerkung von den Betreuern: Selektion der Daten
- Frage: Was seht ihr als relevante Daten?
  - Knoten, Wendestellen, Wege
- Anmerkung: Ampeln nicht vergessen
- Aufpassen! Nicht zu stark selektieren
- Erst einmal gucken was nicht gebraucht wird
- Graphenorientierte Datenbank als Alternative
  - Welches Datenmodell (XML oder graphenorientiert) am besten geeignet ist, hängt immer vom Nutzen ab
  - Graphendatenbank wird von den Betreuern als sinnvoll erachtet
  - Semantik auch in der XML-Datei vorhanden

Gruppe Wetterdaten

- Automatisiertes Einlesen von Verkehrsdaten kann bis dato nicht durchgeführt werden

Gruppe Verkehrsdaten

- Datenschema zur weiteren Verarbeitung
- Anmerkung von den Betreuern: Was ist das für ein Datenschema?
  - Datentyp, Geschwindigkeit etc.
  - Tabellenschema als Notfallplan, falls keine reellen Daten vorhanden, werden Daten anhand des Schemas simuliert
- Kurze Vorstellung der prototypischen Tabelle von Kamiran
  - Anmerkung von den Betreuern: Zusätzlich auch Parkhäuser berücksichtigen
  - Gewichtung der Straßen ist wichtig: Dies wird schon in der Gruppe Openstreetmap abgedeckt; Kapazität der Straßen in Literatur ersichtlich
  - Anmerkung: Ziel ist wichtig bevor die Tabelle in CSV abstrahiert wird

Sonstiges

- Nils nimmt einen neuen Job als SAP Hana Admin an
- Website : Betreuer werden als Auftraggeber mit aufgeführt

Aufgabenverteilung nächste Woche

- Aufteilung in Gruppen:
  1. Openstreetmap Gruppe
  2. Wetterdatengruppe
  3. Allgemeine Datenrecherche Gruppe: Fertigen exemplarisch einen Datensatz an; Verkehrsdaten simulieren
- Website mit Inhalt füllen
  - Team auflisten
  - Aufgabe der Dokumentationsbeauftragte

Sitzungsprotokoll für KW 10

Sitzungsdatum: Montag, 02.03.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Olga Schwarz
Protokollant(in): Philipp Schumacher
Abwesende(r): ——-

Ablauf / Besprechungsergebnisse

Recherche

Die Recherchegruppe hat sich die Inhalte von ALISE angeschaut

ADAC-Daten

- Die ADAC-Daten wurden gesichtet (Marken und Verbräuche)
- Die Tabelle soll noch bereinigt werden mittels SPSS (Kamiran)
- Anschließend sollen die Daten mittels Adobe importiert werden (Jannes)

Wetterdaten

- Import der Wetterdaten funktioniert nicht
- Scheitert am Cromejob
- Nils schreibt Dokument mit den notwendigen Rechten

Internetseite

- Inhalt wird erstellt
- Christian hat die Rechte zur Bearbeitung (LogIn-Daten) von Jannes bekommen

Projektplanung:

- Kai arbeitet Roadmap aus

Sitzungsprotokoll für KW 11

Sitzungsdatum: Montag, 09.03.2015
Uhrzeit: 13:00 - 13:20
Vorsitzende(r): Christian Janßen
Protokollant(in): Janine Haase
Abwesende(r): Philipp Schumacher (Urlaub), Milan Tomovic (unentschuldigt)

Ablauf / Besprechungsergebnisse

Anwesenheit

- Philipp hat Urlaub; Christian übernimmt die Tagesordnung für Philipp
- Milan hat per Whats App Bescheid gegeben, aber den Betreuern nicht, damit fehlt er unentschuldigt.

Arbeitspakete der einzelnen Gruppen:

- Philipp richtet Arbeitspakete ein, es werden unter anderem die folgenden Pakete erstellt (nicht vollständig, wird nachgeholt):
- Kamiran und Janine arbeiten mit Bezug auf Datensimulation die Dokumentation der Projektgruppe ALICE durch
- Kamiran und Janine überprüfen die ADAC-Daten und kümmern sich um weitere Datenbeschaffung
- Olga und Milan prüfen die Verwendbarkeit und den Nutzen von Modellen, insbesondere von DatexII
- Philipp, Kai und Jannes überprüfen die Aufnahme von besonderen Wegpunkten (Ampeln) und Wegealgorithmen

Einbindung der ADAC-Daten

- Möglichkeit zum Einbinden der Daten wird durch Kamiran und Janine geprüft.
- Die Daten müssen noch geprüft und überarbeitet (bereinigt) werden, da einige Begriffe in der Datei nicht einheitlich sind.

Besprechung der Ergebnisse der Gruppen:

Datensuche

- Schreiben an das Kraftfahrtbundesamt:
- Verwendung der ADAC-Daten: Wird durch Kamiran und Janine geprüft und vorgenommen.
- Befüllen der Tabellenschemata
- Import der Daten (Milan und Olga): Muss ggf. aufgrund Abwesenheit von Milan anderweitig vergeben werden.

Wetterdaten

- Konnten die technischen Probleme behoben werden?
- Befüllen in Echtzeit aufgrund eines technischen Problems mit dem Zugriff auf ein Skript noch nicht möglich.
- Es werden weiterhin Lösungsansätze durch Nils gesucht.

GPS-Daten

- Anlegen der Tabellen in der HANA
- über Karten sollen die Daten in eine csv-Datei und dann in die HANA geladen werden.
- Korrektur durch Alexander: Besserer Begriff statt GPS sei Längen- und Breitengrade
- Christian wird den Konverter von XML zu csv weiterschreiben.

Weitere Punkte seitens der Betreuer - Milan:

- Alexander und Benjamin wollten sich heute nach der Sitzung mit Milan zusammen setzen
- Aufgrund der Abwesenheit wird hier vorerst abgewartet, Näheres kann noch nicht gesagt werden.

Aufgabenverteilung nächste Woche

- Siehe Oberpunkt Arbeitspakete, weitere Arbeitspakete werden in der Gruppe verteilt. Wenn Milan ausfällt, werden seien Aufgaben intern auf den Rest der Gruppe verteilt.

Sitzungsprotokoll für KW 12

Sitzungsdatum: Montag, 16.03.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Philipp Schumacher
Protokollant(in): Christian Janßen
Abwesende(r): Janine (Krankheitsbedingt Entschuldigt), Milan (Unentschuldigt und damit ausgeschlossen), Nils(Urlaub)

Ablauf / Besprechungsergebnisse

Mobilitätsdaten

Das vorgefertigte Anschreiben von Janine und Kamiran wurde inzwischen abgenommen. Kamiran arbeitet Änderungen ein und schickt es dann an die zuständigen Unternehmen weiter. Die ADAC Daten werden von Janine und Kamiran weiter durchforstet und aktuell breinigt.

Zusätzliche Datensätze

In einem Gespräch mit Frank Köster vom DLR in Braunschweig konnten eine Einigung erzielt werden und zusätzliche Daten können beschafft werden.

- Daten aus Fahrzeugen:
  - Im Speziellen hier aus Sensoren die beispielsweise bestimmte abweichende Fahrverhaltensweisen aufzeichnen.
- Daten aus stationären Sensoren
  - Im Speziellen hier aus Sensoren die stationäre beispielsweise an einer Ampel installiert sind.
- Daten aus einem Taxiunternehmen (Bedarf weiterer Klärung)

Das DLR ist besonders an sicherheitsrelevanten Informationen interessiert. Dabei könnte ein Use Case beispielsweise daraus zusammengesetzt sein, das eine Analyse bestimmter Knotenpunkte erfolgt, wo zum einen zu dicht aufgefahren wird und zum anderen zu schnell gefahren wird.

In einem weiteren Gespräch konnte eine Zusicherung von Daten seitens der Stadt Oldenburg erzielt werden. Hier wird der erste Schritt eine Kontaktaufnahme bezüglich einer Besichtigung der Verkehrsleitzentrale in Oldenburg sein. Die damit korrespondierende Doodle-Umfrage wurde bereits von Kai aufgesetzt.

Als Use Case stellt sich die Stadt Oldenburg vor allem die Reisezeitermittlung auf verschiedene Verkehrsmittel wie z.B. Bus oder Auto vor.

Wetterdaten

Bei der Gruppe Wetterdaten steht immer noch das Problem mit dem Laden der CSV in die Hana Datenbank im Raum. Dieses Problem ist auch analog in der Gruppe GEO Daten vorhanden. Zur Klärung des Problems befassen sich Jannes und Kai damit und setzten sich mit Alexander zusammen.

Geo Daten

Wie bereits angesprochen herrschen auch hier Probleme mit dem Laden einer CSV Datei in die Hana Datenbank. Der XML Parser ist fertig. Der Converter ebenfalls. Es müssen nur noch kleine Restarbeiten vorgenommen werden.

Roadmap

Jannes hat in der letzten Woche eine Roadmap erstellt und diese heute kurz erläutert. Kai hat damit begonnen Tickets auf der Grundlage dieser Roadmap zu erstellen. In der Kalenderwoche 13 wird Jannes die vollendete Roadmap vorstellen. Diese kann dann unter Redmine und auf dem Server abgerufen werden.

SPSS

Kamiran hat die Möglichkeit überprüft SPSS an der Uni zu bekommen. Die Universitätsbibliothek stellt einen aktuellen Key zur Verfügung. Die Verwendung von SPSS wird allerdings mittelfristig nicht angestrebt.

Aufgabenverteilung nächste Woche

- Christian:
  - Feinschliff CSV Converter, Parser;
  - Füllt Internetseite mit Inhalt
  - Vollendet Java, Javascript und HTML Programmierrichtlinien

- Kai, Jannes:
  - Befassen sich weiter mit dem CSC - Hana Problem
  - Jannes vollendet Roadmap
  - Kai erstellt restliche Tickets
- Philipp:
  - Unterstützung Feinschliff CSV Converter, Parser
- Kamiran:
  - Korrektur Anschreiben und abschicken des Anschreibens an weitere Unternehmen
- Gruppe:
  - Trägt sich in die erstellte Doodle Umfrage für die Besichtigung der Verkehrsleitzentrale ein.

Sitzungsprotokoll für KW 13

Sitzungsdatum: Montag, 23.03.2015
Uhrzeit: 12:45 - 13:15
Vorsitzende(r): Janine Haase
Protokollant(in): Jannes Spekker
Abwesende(r): Christian (Krankheitsbedingt, entschuldigt), Nils (entschuldigt)

Ablauf / Besprechungsergebnisse

Mobilitätsdaten

Anschreiben mit Anfrage zu relevanten Daten wurde von Kamiran an Unternehmen verschickt. Teilweise erhielt er bereits eine Antwort.

- ADAC: Weiterverwiesen an anderen Ansprechpartner; Projektziel muss näher erläutert werden;
- Kassel: Telefongespräch für weitere Details angefragt
- BAS: Absage, die vorgehaltenen Daten enthalten Kundeninformationen BMW: Telefongespräch für weitere Details angefragt
- Magdeburg: Weiterverwiesen an die Stadt, da diese Eigentümer der Daten ist

Kamiran fragt, in wie weit auf Antworten der Unternehmen eingegangen werden soll. Antwort Manuel: Anfragen oder Anregungen durch die Unternehmen können als Vorschläge für mögliche Use-Cases geprüft und aufgenommen werden, ansonsten sollen die eigenen Ideen dargestellt und näher erläutert werden.

Wetterdaten

Import der CSV Datei scheitert an Zugang, da die CSV auf der HANA liegen muss. Von den Betreuern wurde ein Ticket eröffnet, um Zugang zum HANA FTP-Server zu erhalten und somit Daten in die HANA zu laden. Daniel wirft ein, dass es möglich ist die CSV über ein Skript auszulesen und die Daten in ein INSERT Statement in die HANA zu laden. Möglichkeit soll geprüft werden.

Geo Daten

Parser der XML mit Geodaten ist geschrieben, es wird auf Import-Skript gewartet.

Zusätzliche Datensätze

- Stadt Oldenburg: Termin mit Stadt Oldenburg steht noch nicht fest, da auf Antwort der Stadt gewartet wird.
- PG OLIMP: Die laufende PG „OLIMP“ hat sich bereits mit dem Import von großen Datenmengen in die HANA beschäftigt und dafür ein Skript geschrieben. Janine nimmt Kontakt zu Benjamin Hemken auf, der an dem Skript mitgearbeitet hat, um den Zugang zum Code zu erfragen.

Roadmap

Kai stellt Roadmap für ersten Sprint vor. Diese stellt die Aufgabenbereiche für die nächsten Wochen dar und beinhaltet die Aspekte Datenerhebung, Datenverwaltung und Import der Daten. Daniel schlägt vor, eine eigene, automatisierte, Import-Schnittstelle zu programmieren. Dies ist von der PG durch eine generische Import-Schnittstelle vorgesehen.

Internetseite

Philipp hat Inhalte für die Webseite geschrieben, diese werden nach der Sitzung in der Gruppe besprochen und sollen dann veröffentlicht werden.

Aufgabenverteilung nächste Woche

- Kamiran/Janine:
  - Schreiben von Unternehmen bearbeiten/beantworten
  - Daten zu PKW-Bestand sichten
- Olga:
  - Dokumentation vorbereiten (LaTeX)
  - Seminararbeiten in Dokumentation einfügen
- Kai:
  - Tickets aus Roadmap erstellen
- Jannes:
  - weitere Möglichkeiten für Import testen

Sitzungsprotokoll für KW 14

Sitzungsdatum: Montag, 30.03.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Jannes Spekker
Protokollant(in): Kai Hänig
Abwesende(r): ———

Ablauf / Besprechungsergebnisse

Mobilitätsdaten

- Unternehmen wurden angeschrieben
  - Kamiran hat Freitag mit einem Unternehmen telefoniert – konkrete Ergebnisse folgen noch
  - Kamiran hat diese Woche noch drei weitere Telefonate - Offene Fragen: Welches Ziel soll verfolgt werden und wie wir die Daten benutzen.
- Kamirans Vorschlag: Unfalldaten direkt bei der Polizei anfragen.
- Es muss noch abgeklärt werden, welche Daten in welcher Form zur Verfügung gestellt werden können von den Unternehmen.

Website

Philipp hat Texte geschrieben, die letzte Woche besprochen wurden und Jannes hat diese live gestellt.

Dokumentation:

Olga hat ein Grundgerüst für die LATEX-Dokumentation aufgesetzt und erste Inhalte eingefügt.

Gruppenaufteilung:

Aufgrund der Tatsache, dass kontinuierliche Gruppenteilnehmer keine Aufgabe haben und wir momentan nicht wie geplant fortfahren können, wurden Gruppen gebildet, um bereits Themen zu bearbeiten, die weiter in der Zukunft liegen.

- Datenerhebungsgruppe / Datenbankschema – Importer und Handling (Nils, Christian, Kai)
- Analysegruppe – Algorithmen (Phillip, Janine, Kamiran)
- Datenausgabe - Darstellung auf der Webseite (Olga, Jannes)

FTP-Server der Hana

Zugriffsrechte um Daten auf die Hana zu laden besteht nach wie vor nicht.Kontaktaufnahme mit anderer PG bereits erfolgt, um deren Vorgehensweise zu sichten

Datenimport:

- Import der csv-Dateien per SQL importieren - hierbei müssen die Gesamtdateien in einzelne Datensätze gestückelt werden.
- Beschluss: Daten die periodisch importiert werden müssen, werden als SQL Statement gestückelt und sequentiell per Schleife in die Hana importiert.

Besichtigungstermin:

Die PG Teilnehmer sollten die grundsätzlichen Vorgaben der PG kennen und einige Use-Cases im Hinterkopf haben. Im Rahmen des Treffens und der Besichtigung sollten dann entsprechende weitere Ziele mit der Stadt zusammen festgelegt werden; Treffen ist ca. 5 Minuten vor dem Termin

Sonstiges:

Urlaub kann von nun an täglich genommen werden.

Sitzungsprotokoll für KW 16

Sitzungsdatum: Montag, 13.04.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Kai Hänig
Protokollant(in): Nils Worzyk
Abwesende(r): Kamiran, entschuldigt (Urlaub)

Ablauf / Besprechungsergebnisse

Anwesenheit

Alle Gruppenmitglieder sind anwesend, außer Kamiran, der wegen Urlaub entschuldigt fehlt.

Berichte

- Mobilitätsdaten: Es wurde eine Mail an Herr Brandt geschickt, in welcher um die Daten gebeten wurde. Da bis zu dem Sitzungsdatum noch keine Antwort seitens Herr Brandt eingegangen ist, wurde beschlossen gegen Ende der Woche eine weitere Mail abzuschicken und gegebenenfalls anzurufen.
- Wetterdaten: Für die Wetterdaten, wie auch für das gesamte HANA System, ist es noch nicht möglich einen Import automatisiert durchzuführen. Der Ansatz per ODBC auf die HANA zuzugreifen wurde zwar bereits diskutiert, allerdings noch nicht erfolgreich abgeschlossen.
- Geodaten: Wie auch bei den Wetterdaten ist ein automatisierter Import noch nicht möglich. Da es sich bei diesen Daten allerdings, mehr oder weniger, nur um einen einmaligen Import handelt hat sich Kai bereit erklärt, die Daten erstmal händisch zu importieren.

Stand

- Kommunikation mit anderer Gruppe (OLIMP): Die Gruppe wurde angeschrieben und von Seiten der anderen Gruppe gibt es die Bereitschaft sich mit uns zu treffen, für einen Informationsaustausch. Ein genauer Termin ist noch nicht abgesprochen, es wurde die Idee geäußert ein Doodle zu erstellen um den Termin zu bestimmen. Allerdings wurde nach einer kurzen Diskussion entschieden ein solches Doodle nicht durchzuführen, da es zum einen einen erhöhten Aufwand darstellt und zum anderen nicht zwingend alle anwesend sein müssen, wenn dieser Erfahrungsaustausch stattfindet.
- Datenimport mit ODBC-Driver: ODBC wurde auf dem Server installiert, allerdings ist es noch nicht möglich eine Verbindung zwischen Server und HANA herzustellen.

Sonstiges:

- Sonstiges seitens der PG: Keine weiteren Anmerkungen seitens der PG
- Sonstiges seitens der Betreuer: Da die PG die Daten bekommen hat, wird es sich dabei wahrscheinlich um einen konstanten Datensatz und keinen Stream oder ähnliches handeln. Deswegen schlägt Daniel vor, die zweite Woche des Datensatzes für eine "Echtzeit"-Simulation zu nutzen, indem die Daten quasi erst kurz nach ihrem Erstellungsdatum (der per Timestamp bestimmt werden kann) dem System zur Verfügung stehen. Außerdem wird von den Betreuern vorgeschlagen KNIME zu nutzen. KNIME ist ein Tool, welches für die Datenanalyse verwendet wird und uns dabei helfen könnte die Daten zu untersuchen und zu analysieren.

Aufgabenverteilung nächste Woche

- Automatisierten Datenimport funktionsfähig bekommen
- Termin mit der anderen Gruppe abklären

Sitzungsprotokoll für KW 17

Sitzungsdatum: Montag, 20.04.2015
Uhrzeit: 12:15 - 13:45
Vorsitzende(r): Nils Worzyk
Protokollant(in): Kamiran
Abwesende(r): ———

Ablauf / Besprechungsergebnisse

Anwesenheit

- Es gab keine Fragen zum letzten Protokoll.
- Der Import von Daten über ODBC wurde Erfolgreich durchgeführt.
- Jannes führte eine kurze Präsentation zum Web-Frontend-System und erläuterte dies kurz. Des Weiteren wurden einige Analysen grafisch dargestellt.
- Jannes zeigte die Wetterdaten in Form von Tabellen und erläutert dies kurz.
- Olga zeigte eine Vorversion des Endberichts in dem einige Seminararbeiten eingetragen wurden.
- Weitere Kriterien für das Endbericht
  - Im Endbericht sollen alle Protokolle, im Anhang eingefügt werden.
  - Die Tagesordnung soll nicht im Endbericht eingefügt werden.

Sitzungsprotokoll für KW 18

Sitzungsdatum: Montag, 27.04.2015
Uhrzeit: 12:15 - 13:45
Vorsitzende(r): Kamiran
Protokollant(in): Olga Schwarz
Abwesende(r): ———

Ablauf / Besprechungsergebnisse

Berichte

- Zustellung der Daten von Herrn Brandt (Verkehrsleitzentrale)
- Fragen an Brandt von letzter Woche wurden schon beantwortet; GPS Daten selber bestimmen, Manuelle Bearbeitung der Daten notwendig!; Referenz zu den Katen herzustellen
- Busdaten fehlen; Legende fehlt
- IDs Erklärung wird nachgereicht
- Nächster Schritt: Daten schnellst möglichst importieren

Geodaten:

- Wurden bereinigt und liegen auf dem Server vor
- Weitere Daten werden von Kamiran aufbereitet, in wieweit diese im weiteren Verlauf verwendet werden können, wird später entschieden

Beitrag seitens der Betreuer (1): Algorithmen überlegen

- Was ist da sinnvoll?!
- Was können wir nutzen?
- Schnittstellen (Zeitverlust)
- Über dir HANA laufen lassen (Software in der Owncloud) damit kann man Analysen durchführen, Modelle werden auf der Hana erstellt
- Berichte auf dem Rechner: Graphische Oberfläche, um die Berichte zu erstellen (PAL)

Beitrag seitens der Betreuer (2): Die Stadt kann keine Autobahndaten liefern

- Fazit: Alle Daten können verarbeitet werden, da das Ergebnis aggregiert ist
- Statistisches Bundesamt könnte ggf. Daten liefern

Aufgabenverteilung nächste Woche

Problem:

- A: PDFs müssen analysiert werden
- B: Im Datensatz (BUS-Tabelle) fehlen Geokoordinaten, dies stellt die PG vor ein Problem. Zuordnung der Meldepunkte von den Bussen an die Geodaten gestaltet sich dementsprechend schwierig.

Aufgabe

- Bustabelle erstellen:
- Auflistung der Buslinien in Oldenburg (Richtung, Nummer etc.)
- Neue Relations generieren

Neuer fester Termin: Mittwoch von 10:00-17:00 Uhr

Sitzungsprotokoll für KW 19

Sitzungsdatum: Montag, 04.05.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Olga Schwarz
Protokollant(in): Philipp Schumacher
Abwesende(r): ———

Ablauf / Besprechungsergebnisse

Daten der Verkehrsleitzentrale

- Zählspuren (mit D und T in den Exceldateien und Lageplänen codiert)
  - Die Exceldateien mit den Zählspuren wurden unter der Gruppe aufgeteilt und werden nach bestimmten Vereinbarungen mit XML modelliert
  - Die Zählspuren werden mit XML dargestellt
  - Alle Zählspuren werden bis zum Ende der Woche modelliert

Dokumentation

Aktuelle Version der Dokumentation wurde vorgestellt

PAL (Predictive Analysis Library)

- Weitere Rechte/Rollen für die Nutzung sind erforderlich
- Magdeburg wurde nach diesen Rechten/Rollen angeschrieben und hat noch nicht geantwortet

Use Cases

Donnerstag bei der Gruppensitzung werden Use Cases auf Grundlage der vorhandenen Datenbasis besprochen

Weitere Datenbeschaffung

- Möglichkeit der Verwendung von Daten des statistischen Bundesamtes
  - Zielstellung der Projektgruppe soll dazu beschrieben werden
  - Beschaffung von Daten würde wahrscheinlich viel Zeit in Anspruch nehmen

Sitzungsprotokoll für KW 20

Sitzungsdatum: Montag, 11.05.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Philipp Schumacher
Protokollant(in): Christian Janßen
Abwesende(r): Kai Hänig (Krankmeldung)

Ablauf / Besprechungsergebnisse

XML Daten zu den Zählschleifen

- Philipp stellt den Aufbau des XML Dokuments für die Zählschleifen vor und erklärt detailliert die Struktur mit allen Tags.
- Diese Zählschleifen Daten von Herrn Brandt wurden von unserer Seite aus mit Koordinaten versehen
- Manuel hat angemerkt bzw. gefragt, ob es sinnvoll ist Daten von Zählschleifen zu aggregieren die hintereinander liegen
  - Hier wurden explizit die T Zählschleifen, welche 80 m vor der Ampel liegen angemerkt.
  - Von Seiten der Gruppe kamen die Anmerkungen, dass dies geprüft wird und vom Use Case abhängig ist.
  - Zunächst werden aber die Zählschleifen D und T einzeln betrachtet.

ADAC Daten

- Hier wurde eine Bereinigung der ADAC Daten angesprochen. Es liegen keine bereinigten Daten vor
- Des Weiteren sind für uns nur Daten aus den ADAC Daten relevant, welche bspw. Die Co2 Belastung, den Verbrauch und andere Emissionsdaten beinhalten relevant

- Manuel spricht an, dass diese Daten mit den Daten vom Kraftfahrtbundesamt, falls relevant, zusammengepackt werden können.

Daten vom DLR in Braunschweig

- Von der Seite der PG wurde angemerkt, dass die Daten aus Braunschweig im ersten Moment nicht relevant erscheinen, da wir Daten nicht mit Daten aus Oldenburg vermischen wollen
- Ben hatte angemerkt, dass in Punkt 1 aus der DLR E-Mail auch Daten vorhanden sind, die ortsunabhängig sind
  - Aus diesen Daten können Thesen erstelle werden
  - Simulation des Fahrflusses
  - Diese Daten können über Oldenburger Daten gelegt werden und sind nicht an spezifische Use Cases von Seiten des DLR gebunden.
  - Kamiran wird Daten anfragen als Entscheidungsfindung für Punkt 1.

Use Cases

- In den vergangenen Wochen wurden grobe Use Cases besprochen.
- Philipp stellte diese kurz vor.
- Im Anschluss der Sitzung setzt die PG sich zusammen und arbeitet diese weiter aus.
- Manuel hat angemerkt, falls noch konkret Interesse an irgendwelchen Daten bestünden, sollte man ihn kontaktieren.
- Er hat einen Kontakt beim statistischen Bundesamt
  - Emissionsdaten
  - Bahndaten
  - Unfalldaten

PAL

Noch keine neuen Erkenntnisse über die Rechtevergabe aus Magdeburg

Aufgabenverteilung nächste Woche

- Christian:
  - Kümmert sich um den Datenimport der HANA
- Kamiran:

  - Kümmert sich um die Bereinigung der relevanten ADAC Daten
  - Kommuniziert mit Manuel über weitere Daten seitens des statistischen Bundesamts

- Olga:
  - Kümmert sich um Veranstaltungsdaten detailliert

- Philipp:
  - Lädt Dokument mit Use Cases hoch und bringt Struktur in eben dieses
  - Setzt sich mit Nils zusammen und berät über PAL

- Nils:
  - Macht sich zu Pal Gedanken. Alternativen und eine konkrete Möglichkeit soll nächste Woche vorliegen

- Kai:
  - Wird gesund
  - Befasst sich mit PM Aufgaben und erstellt Tickets etc...
  - Befasst sich zudem mit Algorithmen

- Jannes:
  - Arbeitet am Portal weiter
  - Busströme/Verkehrsströme
  - Bennent parallel Anforderungen für dieses

Sitzungsprotokoll für KW 21

Sitzungsdatum: Montag, 18.05.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Christian Janßen
Protokollant(in): Jannes Spekker
Abwesende(r): ———

Ablauf / Besprechungsergebnisse

Algorithmen

- Nils stellt die Ergebnisse der letzten Wochen vor
- PAL Ansatz wird aufgrund fehlender Unterstützung von Magdeburg nicht weiter verfolgt
- Betreuer schlagen R als Analysebibliothek vor
- Nils stellt grundsätzliche Server- und Hardwarevoraussetzungen von R vor
- Es wird auf Performancenachteile hingewiesen, die sich aus der Architektur ergeben
- Philipp erklärt Funktionsweise von R
- Beschluss: R wird verwendet, falls alle Voraussetzungen für den Einsatz gewährleistet werden können

Daten

- ADAC Daten
  - Kamiran hat die Daten nach Marke unterteilt
  - Berechnung der Durchschnittswerte relevanter Informationen
- Kontakt statistisches Bundesamt

  - E-Mail mit Datenanfrage von Kamiran an Manuel
  - Manuel lieferte Link, unter dem Datensätze zu finden sind
  - Datensätze liegen in stark aggregierter Form vor
- Kontakt Braunschweig
  - Kai hat per E-Mail Daten bei DLR angefragt
  - Bislang keine Antwort erhalten
- Veranstaltungsdaten
  - Olga hat Daten um Adress- und Koordinatendaten erweitert, außerdem wurden Besucherzahlen ergänzt
- Festlegung von Zählschleifen
  - KFZ-Zählschleifen sind bereinigt worden
  - Herr Brandt (Verkehrsleitzentrale Stadt Oldenburg) hat Fahrrad- Zählschleifen geschickt
  - Buszählschleifen (Telegrammdaten) liegen vor, Problem mit Lokalisierung, hoher manueller Aufwand

Use Cases

- Use Cases wurden in der Gruppe durchgesprochen und festgelegt
- Philipp hat eine Strukturierung vorgenommen
- Aus Use Cases werden im Folgenden Anforderungsdefinitionen erstellt

Portal

- Jannes stellt aktuellen Stand des Portals vor
  - Analysedashboard enthält Kartenansicht, Filter ermöglichen das Ein- und Ausblenden von Layern
  - Wetterdaten werden im Diagramm dargestellt
  - Informationen zu Nahverkehr und Veranstaltungen werden eingeblendet

Einzelgespräche

Termine für Einzelgespräche werden schnellstmöglich festgelegt

Aufgabenverteilung nächste Woche

- Christian:
  - Hana Import der finalen OSM Daten
  - Hana Import der Messspulendaten
- Kamiran:
  - ADAC Daten auswerten Bundesamts
- Olga:
  - Veranstaltungsdaten importieren
- Philipp:
  - In R einarbeiten
  - Anwendungsfälle beschreiben
- Nils:
  - In R einarbeiten
  - R auf Server installieren
- Kai:
  - Prognose-Algorithmus skizzieren
- Jannes:
  - Nahverkehrsdaten erheben und bereinigen
  - Nahverkehrsdaten in Hana importieren

Sitzungsprotokoll für KW 23

Sitzungsdatum: Montag, 01.06.2015
Uhrzeit: 12:15 - 13:45
Vorsitzende(r): Jannes Spekker
Protokollant(in): Kamiran Tizyani
Abwesende(r): Kai Hänig (Urlaub)

Ablauf / Besprechungsergebnisse

Raum Nutzung anderer PGs

- Im Gruppenraum wurde festgestellt, dass der Raum durch unangemeldete Personen genutzt wird.
- Die Forschungsgruppe, die den PG-Raum nutzt wird in eine andere Räumlichkeit verlegt.

Use Cases

- Es wurden Anforderungsdefinitionen erstellt, die zunächst von der PG geprüft werden.

Daten

DLR

- Kai hat Kontakt mit der DLR aufgenommen.
- Es wurden verschiedene Termine über ein Doodle von der DLR vorgeschlagen. Der genaue Termin wird der PG bald mitgeteilt.

Nahverkehr

Jannes erarbeitet die Bus-Daten, um sie darzustellen.

Veranstaltungen

Die Veranstaltungsdaten werden von Olga erarbeitet, die anschließend in die Hana geladen werden.

Verkehrsleitzentrale

Es wurden Daten für 30 Tage erarbeitet. Des Weiteren ist eine Erweiterung in Planung, in der eine Auswertung für 90 Tage erstellt wird.

Prognose

- Es wurde im Zusammenhang auf die Prognose, weitere Recherchen von Kai und Nils durchgeführt.
- Für eine genauere Prognose werden weitere Daten benötigt. Hierzu wird Herr Brandt kontaktiert und gebeten uns weitere Daten zur Verfügung zu stellen.
- Für die Verarbeitung und Darstellung der Daten werden noch weitere Methoden und Ideen entwickelt.

Sonstiges

- Einzelgespräche: Die Termine für die Einzelgespräche wurden noch nicht festgelegt. Diese werden von den Betreuern demnächst festgelegt.

Sitzungsprotokoll für KW 24

Sitzungsdatum: Montag, 08.06.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Kamiran Tizyani
Protokollant(in): Kai Hänig
Abwesende(r): ———

Ablauf / Besprechungsergebnisse

Raumnutzung des PG Raumes:

- Die Gruppe RAPID hat den Projektgruppenraum Montags, Dienstags und Mittwochs.
- Donnerstags und Freitags teilen sich die anderen beiden Gruppen

DLR:

- Es soll der Termin abgewartet werden um zu klären, welche Verpflichtungen mit den akquirierten Daten einher gehen.
- Dies wird alles dann durch die Projektgruppe im Termin vom 19.06.15. geklärt werden.
- Hierbei sollte sich die Projektgruppe genau darauf vorbereiten.
- Wenn die Daten nicht dem Projektgruppenziel dienen, bzw. sich nicht mit den bisherigen Daten kombinieren lassen, besteht die Möglichkeit diese abzulehnen.

Sitzungsprotokoll für KW 25

Sitzungsdatum: Montag, 15.06.2015
Uhrzeit: 12:15 - 13:45
Vorsitzende(r): Kai Hänig
Protokollant(in): Olga Schwarz
Abwesende(r): Nils Worzyk (Urlaub)

Ablauf / Besprechungsergebnisse

Kurzfristige Prognose

- kurze Erläuterung seitens Kai
- Stand: Diagramme, die den Verkehrsfluss vorhersagen(kurzfristig)

Langfristige Prognose

- kurze Erläuterung seitens Philipp
- Kategorisierung wird vorgenommen: Tage werden unterschieden in Wochenende, Werktage, Feiertage,
- Funktionsgraph, beispielsweise Vorhersage für den nächsten Samstag; Funktion wird direkt an den gewünschten Zeitpunkt angewendet. (Wetter und Events werden im nächsten Schritt als Faktoren berücksichtigt)

Dokumentation

- Protokolle werden in Latex überführt

Sonstiges

- DLR-Termin wurde auf den 06.07 verschoben
- Einzelgespräche: Reihenfolge wird festgelegt und an die Betreuer geschickt

Sitzungsprotokoll für KW 26

Sitzungsdatum: Montag, 22.06.2015
Uhrzeit: 12:15 - 13:45
Vorsitzende(r): Olga Schwarz
Protokollant(in): Nils Worzyk
Abwesende(r): ————

Ablauf / Besprechungsergebnisse

Kurzfristige Prognose

- Formel ist vorhanden
- insert/update muss noch ausformuliert werden
- Warten auf die Tabellen von Jannes
- Inputwert ist Name der Spule

Langfristige Prognose

- Werte in Gruppen aufgeteilt
  - Montag-Donnerstag
  - Freitag
  - Samstag
  - Sonntag/Feiertag
- Werte mittels view berechnen und Tabellen erstellen
- Regressionsanalyse
- Frage: googleVis verwenden? Rechtemäßig ist das ok von den Betreuern
- Für jede Gruppe kann ein Scatterplot erstellt werden

- Für jede Gruppe kann eine Regressionsfunktion berechnet werden
- Bemerkung: Können Trends mit berücksichtigt werden? Noch keine Trends, vielleicht später

DLR – Präsentation für das Meeting am 06.07.2015

- Präsentation ist in Vorbereitung, kann kommende Woche vorgestellt werden

ADAC-Daten

- Es ging darum, was mit der Tabelle passiert?
- Relativ häufig das selbe Fahrzeug mit unterschiedlicher Ausstattung
- Tabelle so gut wie möglich in die HANA laden und die Schnelligkeit der HANA ausnutzen um die Daten in der HANA zu aggregieren
- Daten müssen bereinigt werden
- Wie sind die Autoklassen verteilt, in der Literatur nachschlagen
- Ihr müsst nach den Klassen gehen, die von der Stadt gegeben werden (Alexander)
- Beispiel: Golf wird stärker in die Gewichtung eingehen, als andere Autos

Sitzungsprotokoll für KW 27

Sitzungsdatum: Montag, 29.06.2015
Uhrzeit: 12:15 - 13:00
Vorsitzende(r): Nils Worzyk
Protokollant(in): Philipp Schumacher
Abwesende(r): ———

Ablauf / Besprechungsergebnisse

Kurzfristige Prognose

- Soll von der Rapid-Plattform ermöglicht werden
- Noch Probleme beim SQL-Script („store procedures“ im Speziellen)

Langfristige Prognose

- Philipps R-Script wurde fertig geschrieben und im SQL-Code von Christian eingebettet
- Die Daten für die Erstellung der einzelnen Scatterplots und für den Erstellung der Regressionskurven wurden noch nicht in Tabellen gespeichert (Fehler im R- oder SQL-Code)
- Fehler wird voraussichtlich noch heute behoben

Sonstiges

- Es soll in zwei Wochen ein Gruppenfoto gemacht werden
- Manuel wird in 2 Wochen seine Tätigkeit in der VLBA beenden

Sitzungsprotokoll für KW 28

Sitzungsdatum: Montag, 06.07.2015
Uhrzeit: 12:15 - 13:45
Vorsitzende(r): Philipp Schumacher
Protokollant(in): Christian Janßen
Abwesende(r): Olga Schwarz (Urlaub)

Ablauf / Besprechungsergebnisse

Kurzfristige Prognose

- Die kurzfristige Prognose ist lauffähig und es müssen nur noch Kleinigkeiten angepasst werden
- Es funktioniert allerdings
- Problematisch wird nur die Gruppierung in Tagen sein, da in der kurzfristigen Prognose zu wenige Werte für eine adäquate Berechnung zur Verfügung stehen.
- Die Prognosearten sind mit 15,30 und 60 Minuten angegeben

Langfristige Prognose

- Die Probleme mit der Kommunikation mittels RServe konnten in den Griff bekommen werden
- Es können nun die Scatterplot Koordinaten sowie die Koeffizienten für die Polynominal Regression in Tabellenform aufgebaut werden
- Ein Beispiel wie der Scatterplot aussieht, wurde von Philipp in SAP Hana gezeigt
- Wir warten zudem noch auf die restlichen Zählschleifendaten die von Jannes importiert werden

Gespräch mit DLR

- Kamiran stellt geplante Abfolge des Gesprächs vor.
- Wir stellen uns zunächst vor und dann werden die Ziele der Projektgruppe erläutert
- Im Anschluss warten wir auf die Projektvorstellung des DLR
- Das Gespräch startet um 16 Uhr

Sonstiges seitens der Gruppe

- Kai hat angemerkt das die Verschiebung des Termins um einen weiteren Monat nicht gut für den weiteren Verlauf der PG ist
- Wir verständigen uns darauf, erst einmal die Informationen vom DLR zu sammeln und zu sichten und dann zu entscheiden wie diese für das Projekt von Vorteil sein könnten.

Sonstiges seitens der Betreuer

- Alexander wollte wissen was unsere Ziele sind und wie diese mit dem DLR gemerged werden können. Dies wurde als Frage für die Interviewrunde aufgenommen

Aufgabenverteilung nächste Woche

Die Teilgruppen arbeiten weiter an der Prognose sowie dem Frontend.

Olga und Kamiran bearbeiten gemeinsam die gestellte Programmieraufgabe. Dort soll der Verbrauch in Co2 pro Zählschleife berechnet und dargestellt werden.

Sitzungsprotokoll für KW 29

Sitzungsdatum: Montag, 13.07.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Christian Janßen
Protokollant(in): Kai Hänig
Abwesende(r): Jannes Spekker (Urlaub)

Ablauf / Besprechungsergebnisse

Kurzfristige Prognose

- Kurzfristiger Algorithmus funktioniert bei 30 und 60 Minuten, 15 wird diese Woche fertig gestellt.

Langfristige Prognose

- Langfristiger Algorithmus muss noch durch einige Tabellen ergänzt werden.

Gespräch mit DLR

- Kamiran kommuniziert mit dem DLR, dass eine Zusammenarbeit so kurzfristig nicht möglich ist
- Kamiran hat bereits die Präsentationen mit dem DLR ausgetauscht.

Sonstiges

- Olga wird den Zwischenstand der Dokumentation den Betreuern zukommen lassen

Sitzungsprotokoll für KW 30

Sitzungsdatum: Montag, 20.07.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Kai Hänig
Protokollant(in): Jannes Spekker
Abwesende(r):

Ablauf / Besprechungsergebnisse

Kurzfristige Prognose

- 15-minütig (Nils): Fehler wurden behoben, die Methode zur kurzfristigen Prognose (15 Minuten) ist lauffähig.
- 30-/60-minütig (Kai): Die genutzte Methode zur kurzfristigen für 30 und 60 Minuten wurden umgestellt, das Skript ist lauffähig.

Langfristige Prognose

- Tabelle der Durschnittswerte zu den Zählspulen fehlt. Werte werden derzeit berechnet, allerdings sind noch Nullwertde vorhanden, die zu Unterbrechung des Scripts führen. Nullwerte müssen von Jannes aufgefüllt werden.

DLR Nachbereitung

- Kamiran hat eine Nachricht an DLR geschickt, die Daten können aus zeitlichen Gründen nicht mehr berücksichtigt werden

Sonstiges

- Frage der Betreuer nach Vorstellung der Gruppe von Darstellung des Outputs.
- Gruppe: Nutzer noch nicht klar definiert. Visuelle Darstellung anhand einer Karte, die einzelnen Zählspulen sind darauf dargestellt. Per Mausklick werden Prognoseergebnisse anhand von Diagrammen dargestellt.

- Betreuer nennen Verkehrsleitzentrale als nnatürlichenÄdressaten, die Story sollte für diesen Nutzer aufgebaut werden. Breite Masse soll ebenfalls berücksichtigt werden, allerdings wird hier für konkrete Anwendung der Rahmen gesprengt.
- Zudem wird von Betreuern vorgeschlagen, weitere Daten der Verkehrsleitzentrale zu Zählspulen anzufragen. Kai fragt Daten bei VLZ an.

Sitzungsprotokoll für KW 31

Sitzungsdatum: Montag, 27.07.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Jannes Spekker
Protokollant(in): Nils Worzyk
Abwesende(r): Kamiran (Verspätung), Philipp (entschuldigt)

Ablauf / Besprechungsergebnisse

HANA-Performance-Probleme

- das Script zieht alle werte für eine bestimmte zeit für alle spalten heraus und berechnet daraus den Mittelwert
- das Script performanter machen
- schauen, dass das Script in möglichst nur einer Transaktion auf die HANA schreibt
- durch das Script wurde die HANA performancetechnisch eingeschränkt und trotz Abbruch des Scriptes am 22.07. ist die HANA sehr eingeschränkt nutzbar
- Kai geht in direkten Kontakt mit dem Menschen aus Magdeburg

Ausblick: HANA - Upgrade

- SAP graph engine in HANA einbauen
- bessere Darstellung der Kreuzungspunkte und so
- erst mit SPSS11

Sitzungsprotokoll für KW 32

Sitzungsdatum: Montag, 03.08.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Nils Worzyk
Protokollant(in): Kamiran Tizyani
Abwesende(r): Jannes (Urlaub) / Kai (krank abgemeldet)

Ablauf / Besprechungsergebnisse

Mittelwerttabelle

- Die Hana ist am vergangenen Donnerstag wieder abgestürzt
- Für eine Gruppe dauert die Berechnung des Mittelwerts 15 Stunden.
- Die PG wünscht, falls die HANA wieder abstürzt, eine Fehlermeldung zu bekommen.

Daten der Verkehrsleitzentrale

- Es wurde eine Mail an Herrn Brandt bereits gesendet. Auf die Anfrage gibt es noch keine Antwort. Hierdurch ist es noch offen, ob wir weitere Daten bekommen.

Logo der Stadt

- Das Logo der Stadt kann in Dokumentation übernommen werden.

Sitzungsprotokoll für KW 33

Sitzungsdatum: Montag, 10.08.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Kamiran Tizyani
Protokollant(in): Olga Schwarz
Abwesende(r): Jannes Spekker (Urlaub)

Ablauf / Besprechungsergebnisse

Daten von der Stadt Oldenburg

- Brand hat sich zu unserem Anliegen weitere Daten von der Verkehrsleitzentrale zu erhalten wie folgt geäußert: Es muss noch geklärt werden, da es für die Stadt Oldenburg mit hohem Zeitaufwand verbunden ist. Viele Mitarbeiter sind zurzeit im Urlaub, deshalb ist es im Moment nicht möglich uns weitere Daten zur Verfügung zu stellen.
- Einwand von den Betreuern: Wurde bei der Stadt Oldenburg nachgefragt, ob die neuen Daten, die gerade entstehen über eine Schnittstelle direkt in die HANA übermittelt werden können? Stichwort: Echtzeitimport
- Kai: Für unsere PG ist das realisierbar, da dafür nur der Trigger verändert werden müsste.Das Skript läuft weiter, wenn neue Daten reinkommen. Allerdings hat die Verkehrsleitzentrale im Moment keine Möglichkeit weiter Daten zur Verfügung zu stellen.

CO2 - Verbrauch

- Präsentation und Erläuterung des Sourcecodes für die CO2-Belastung in Oldenburg von Olga: Bis lang wurde eine Prozedur erstellt, die für einen beliebigen Zeitpunkt (Übergabe als Parameter) für jede Spule (beliebige Stelle in Oldenburg, wo der Verkehrsfluss über eine Induktionsspule erfasst wird) den Co2 Verbrauch berechnet. Der Verkehrsfluss wird mit einem Durchschnittswert des Co2-Verbrauchs in Oldenburg multipliziert.

- Im weiteren Verlauf der Programmieraufgabe wird die CO2-Belastung für einen beliebigen Zeitpunkt für 15 Minuten, 30 Minuten und 60 Minuten prognostiziert.
- Es soll sich an die Tabelle PREDICTION gehalten werden.
- Beitrag von den Betreuern: Gleiche Prozedur für die Feinstaubbelastung in Oldenburg erstellen. Dabei wurde angemerkt, dass Oldenburg durch einen hohen Feinstaubausstoß belastet wird. Dies ergibt ein neues Use Case.

Darstellung der Website

- Präsentation und Beschreibung der Website von Christian
- Christian hat in die Frontend-Gruppe gewechselt

Langfristige Prognose

- Kurze Erläuterung des aktuellen Stands von Philipp
- Regressionkurve wurde gezeigt

Kurzfristige Prognose

- Die Dauer der Prozedurlaufzeit stellt die PG vor ein Problem.

Sonstiges

- **Beitrag von den Betreuern**: Wie weit seid ihr bei der Visualisierung von den Karten?
- Wurde mit dem Zeigen der Website beantwortet.

Sitzungsprotokoll für KW 34

Sitzungsdatum: Montag, 17.08.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Nils Worzyk
Protokollant(in): Philipp Schumacher
Abwesende(r):

Ablauf / Besprechungsergebnisse

Daten von der Stadt Oldenburg

- Es werden keine weiteren Daten hinzugenommen

Feinstaub

- Kennzahlen
- Wahrscheinlichkeiten

Frontend

- Löschen von Benutzern möglich
- Profil- und Rollenfunktionen angepasst
- Jannes wieder da;...es kann weitergearbeitet werden
- Nächste Woche wird eine Demo gegeben

Langfristige Prognose

- Skripte für AveragerGroup 2 und 3 laufen noch
- Dauert so lange, weil for-Schleife manuell gesteuert wird und nicht automatisch durchlaufen kann

Kurzfristige Prognose

- Funktioniert
- Weitgehend abgeschlossen

Sitzungsprotokoll für KW 35

Sitzungsdatum: Montag, 24.08.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Philipp Schumacher
Protokollant(in): Christian Janßen
Abwesende(r): Kamiran (Urlaub)

Ablauf / Besprechungsergebnisse

Dokumentation

- Bei der kurzfristigen Prognose hat Nils in der vergangenen Woche Verfahren recherchiert und ein paar Sachen dazu aufgeschrieben. Dies wird nächste Woche vorgestellt.
- Bei der Langfristigen Prognose hat Kai seinen Teil abgeschlossen und Olga bereits geschickt
- Kamiran hat sich um den Projektmanagement Teil gekümmert der allerdings nochmal komplett überarbeitet werden muss
- Alex stellt eine Frage bzgl. des Inhaltsverzeichnisses. Daraufhin zeigt Philipp die neue Gliederung der Dokumentation
- Das Kapitel technische Rahmenbedingungen kann als Standardkapitel angesehen werden und sollte nicht ausführlich beschrieben werden
- Kapitel Datenerhebung/Datenbereinigung war für Ben nicht ganz klar
- Nils erklärt den Zusammenhang auch anhand des Modells CRISP
- WICHTIG: Es sollte ein paar Sätze zur Gliederung geschrieben werden warum dieser Ansatz so verfolgt wurde. Die Gedankengänge der Projektgruppe sollte in der Dokumentation visualisiert werden

Frontend

- Zunächst erklärt Jannes den Fortschritt der Zählschleifen und zeigt die Darstellung.
- Die geplanten weiteren Schritte sind
  - Interaktive, Ajax gesteuerte Darstellung der Zählschleifen (Verhindert das Laden der gesamten Seite neu)
  - Widgets
  - Langfristige Prognose
  - Einflüsse wie Wetter oder Veranstaltungen
- Anmerkungen von Ben WICHTIG: Griffige Beispiele für die Visualisierung überlegen und Darstellen. Use Case bezogen etwas ausdenken und sinnvoll verknüpfen. Den Mehrwert für ein Unternehmen aufzeigen dieses Tool zu verwenden
- Anmerkungen von Daniel WICHTIG: Für die Präsentation einen Use Case raussuchen und diesen durch die ganze Präsentation ziehen. Ein Roter Pfaden muss zu erkennen sein
- Christian stellt im Anschluss arbeiten am Backend (Password Recovery) vor

Anmerkungen seitens der PG /Betreuer

- Olga stellt eine Frage bzgl. der von Ben angesprochenen klassischen Gliederung
- Plakat/Poster/Flyer müssen erstellt werden. Dies kann auch im Anschluss der Projektabgabe erfolgen. Kosten werden voraussichtlich von der Abteilung übernommen
- Abgabetermin/Präsentationstermin muss noch besprochen werden. Zudem müssen organisatorische Dinge wie Raumbuchung, Terminabstimmung innerhalb der Gruppe koordiniert werden
- Intern festgehaltenes Abgabedatum ist der 30.09.2015
- Nochmals die Anmerkung von Daniel für die Präsentation griffige Ergebnisse zu zeigen
- Es müssen Gedanken über Use Cases angestellt werden. Als Beispiele sind die Feinstaubproblematiken in der Innenstadt gefallen

Sitzungsprotokoll für KW 36

Sitzungsdatum: Montag, 31.08.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Christian Janßen
Protokollant(in): Jannes Spekker
Abwesende(r): Olga (Urlaub), Kamiran (Urlaub)

Ablauf / Besprechungsergebnisse

Dokumentation

- Kai: Dokumentation zu Projektmanagement abgeschlossen, ER-Diagramm in Bearbeitung
- Nils: Datenbereinigung in Bearbeitung, Recherche abgeschlossen
- Philipp: Einleitung grob abgeschlossen
- OSM-Daten in Bearbeitung (Datenerhebung)

Frontend

- grober Use Case wurde festgelegt (für Präsentation), wird verfeinert und in kommender Sitzung vorgestellt
- Widget-Funktionalität vorbereitet
- Server-Zeit für Echtzeitsimulation auf März umgestellt
- GUI weiterentwickelt
- kürzeste Wege-Funktionalität einfügen

Anmerkungen seitens der PG /Betreuer

- Prüfungstermin wurde noch nicht eingetragen
- Social Event ist in Planung

Sitzungsprotokoll für KW 37

Sitzungsdatum: Montag, 07.09.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Jannes Spekker
Protokollant(in): Kai Hänig
Abwesende(r): Olga (Urlaub), Kamiran (Urlaub)

Ablauf / Besprechungsergebnisse

Dokumentation

- Frontend soll bis Mitte nächster Woche fertig sein, danach Bugfixing und Dokumentation
- Nils macht kurzfristige Algorithmen, Nils unterstützt bei der Datenbereinigung
- Christian: Datenerhebung fertig, Datenbereinigung wird gerade erstellt (OSM, ADAC, Zählspulen)
- Kai: Projektmanagement fertig, Dokumentation der Stored Procedures zur langfristigen Prognose, Use-Cases für die Präsentation erstellen.
- Phillipp: allgemeine Gliederung, langfristige Prognose,
- Kamiran: Projektmanagement, Teil über die Ansprache einzelner Unternehmen.

Frontend

- Jannes kümmert sich um die Bahndaten
- Philipp kümmert sich noch um die langfristige Prognose, Problematik bei der Schnittstelle zum Frontend.

Sitzungsprotokoll für KW 38

Sitzungsdatum: Montag, 14.09.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Kai Hänig
Protokollant(in): Kamiran Tizyani
Abwesende(r): ————-

Ablauf / Besprechungsergebnisse

Use-Case

- Es wurden zwei Use-Case überlegt und definiert.
- Der erste Use Case: Experten-user ist, der Stadt oder die VWG. Der Experten-user hat die Möglichkeit, den aktuellen Stand des Verkehrsflusse zu überprüfen. Er kann einen bestimmten Tag aus der Vergangenheit auswählen, an dem besonders viel Stauaufkommen in Oldenburg gegeben hat. Hierdurch kann der Experten-user die Ursachen für den Stau herausfinden und mögliche gegen Maßnahmen einleiten, um dies in der Zukunft besser in Griff zu bekommen.
- Der zweite Use-Case: In diesem Use-Case kann der User mit einer kurzfristigen und einer langfristigen Prognose arbeiten. Der aktuelle Zeitpunkt ist der 26.03.2015. Der Experten-User möchte eine Prognose Für einerseits 30 Minuten und andererseits für einen längeren Zeitpunkt von über 100 Tagen dargestellt bekommen. Die kurzfristige Prognose dient der Überprüfung der Ampelschaltung. Diese werden auf Basis kurzfristige Änderungen angepasst, um so einen Stau zu vermeiden. Die langfristige Prognose wird von dem User verwendet, um die Planung einer zukünftigen Baustelle durchzuführen.

Frontend

- Jannes ist dabei die Heatmap einzurichten
- Jannes kümmert sich weiterhin um die Darstellung der Zählspulen

- Christian hat die Bahndaten aus Oldenburg komplett digitalisiert. Die Zeiten der Bahn wurden in die Hana verknüpft.
- Christian hat Parallel dazu begonnen, Darstellungsmöglichkeiten für die langfristige Prognose zu selektieren.

Sitzungsprotokoll für KW 39

Sitzungsdatum: Montag, 21.09.2015
Uhrzeit: 12:15 - 12:45
Vorsitzende(r): Kamiran Tizyani
Protokollant(in): Nils Worzyk
Abwesende(r): Jannes, wegen Krankheit entschuldigt

Ablauf / Besprechungsergebnisse

Dokumentation

- Darstellung von Quellcode: Sollen längere Codeabschnitte in den Anhang oder nicht?
- Generell soll Quellcode in den Anhang
- Nur wichtige Abschnitte sollen in den Fließtext aufgenommen werden
- Olga geht die Gliederung durch und stellt zu jedem Unterpunkt den aktuellen Fortschritt vor
- Frage nach der Benennung der Use-Cases
- Anwendungsfallszenario für den übergeordneten Anwendungsfall
- Die Anwendungsfälle für die Webseite sollen weiter Use-Case heißen

Frontend

- Auf die HANA kann zur Zeit nicht in akzeptabler Zeit zugegriffen werden. Die Ursache ist zu diesem Zeitpunkt noch unbekannt
- Dadurch, dass auf die HANA nicht zugegriffen werden kann, ist auch ein Zugriff auf das Frontend nicht möglich, da benötigte Werte nicht abgefragt werden können

Sonstiges

- Christian hat eine Frage zum geplanten Update der HANA auf die neuste Version und äußert Bedenken, falls das Update in die Zeit der Projektgruppe fällt. Die Betreuer versichern, dass das Update nicht mehr für uns relevant ist, weil es erst sehr viel später durchgeführt werden soll
- Kamiran fragt, ob die Sitzungen auch im Oktober weiter fortgeführt werden. Die Betreuer sagen, dass die Sitzungen auch bis zum 15. Oktober weiter fortgeführt werden sollen
- Es wurde die Frage gestellt, wie es mit einem Social Event nach der Projektgruppe aussieht. Bezüglich des Social Events möchte sich Alexander bei der anderen PG erkundigen, wie die zu einem solchen Event stehen und auch Abteilungsintern nach Interessenten fragen
- Die Frage nach dem Termin für die Abschlusspräsentation hat ergeben, dass dieser noch nicht sicher anzugeben ist. Wenn Herr Gomez, welcher im folgenden Semester ein Forschungssemester im Ausland absolviert, per Skype an der Präsentation teilnehmen möchte würde der Termin von ihm abhängen und mehr oder weniger vorgegeben werden. Wenn er nicht teilnehmen möchte kann die Projektgruppe selber entscheiden, wann der Termin der Präsentation ist.

Sitzungsprotokoll für KW 40

Sitzungsdatum: Montag, 28.09.2015
Uhrzeit: 12:15 - 13:00
Vorsitzende(r): Nils Worzyk
Protokollant(in): Olga Schwarz
Abwesende(r): ———

Ablauf / Besprechungsergebnisse

Dokumentation

- Nils stellt den neuen Stand der Dokumentation vor
- Zuordnung der Gruppenmitglieder an die Einzelnen Kapitel.
- Informiert die Betreuer, welche Kapitel noch in Arbeit sind und welche fertiggestellt sind

Front End

- Jannes stellt das Webportal vor
- Allgemeine Rückmeldung der Betreuer zu dem Webportal (Darstellung der Ergebnisse) ist positiv.
- Legende fehlt
- Definition der Farblichen Abgrenzung soll dynamisch an die Daten angepasst werden und nicht statisch sein
- Intensität der Farbe soll berücksichtigt werden.
- Farben verschmelzen für eine einheitliche Darstellung
- Je mehr Autos über die Zählspule fahren, desto größer wird der Kreis.

- Überlagerungen durch die Verteilung der Zählspule
- Datum und Ereignisse (Events) werden oben-rechts dargestellt.
- Es ist möglich in die Vergangenheit zu gehen, um Unterschiede sehen zu können

Sonstiges

- Treffen werden optional weitergeführt.
- Die heutige Sitzung ist somit offiziell die letzte Sitzung.
- Es wird Gruppenfoto für die Dokumentation erstellt.

Abschließende Erklärung

Wir versicheren hiermit, dass wir die Dokumentation im Projekt "RAPID" selbständig und ohne fremde Hilfe angefertigt haben, und dass wir alle von anderen Autoren wörtlich übernommenen Stellen wie auch die sich an die Gedankengänge anderer Autoren eng anlegenden Ausführungen unserer Arbeit besonders gekennzeichnet und die Quellen zitiert haben.

Oldenburg, den 15. Oktober 2015	Kai Hänig
Oldenburg, den 15. Oktober 2015	Christian Janßen, B. Sc.
Oldenburg, den 15. Oktober 2015	Philipp Schumacher, B. Sc
Oldenburg, den 15. Oktober 2015	Olga Schwarz
Oldenburg, den 15. Oktober 2015	Jannes Spekker
Oldenburg, den 15. Oktober 2015	Kamiran Tizyani
Oldenburg, den 15. Oktober 2015	Nils Worzyk