



Hochschule für Angewandte Wissenschaften Hamburg
Fakultät Design, Medien und Information
Department Medientechnik

Bachelor-Thesis zur Erlangung des akademischen Grades B.Sc.

Vergleich Oracle 18c Datenbanksystem in einer Cloud Lösung versus XE hinsichtlich Data-Mining-Verfahren in der Hochschullehre.

Patrick Jarren

Matr.-Nr.: 2339659

Erstprüferin: Prof. Dr.-Ing. Sabine Schumann

Zweitprüferin: Prof. Dr.-Ing. Marina Tropmann-Frick

Hamburg, 03.08.2020

Inhaltsverzeichnis

1	Einleitung	4
1.1	Motivation	4
1.2	Ziel und Struktur der Arbeit	4
1.3	Hinweise zum Betriebssystem	5
2	Grundlagen	6
2.1	Relationale Datenbanken	8
2.1.1	In-Memory	9
2.1.2	Zeilenorientierte Datenbanken	10
2.1.3	Autonome Datenbanken	10
2.2	Cloud	11
2.2.1	Infrastructure as a Service	13
2.2.2	Platform as a Service	13
2.2.3	Software as a Service	13
2.2.4	Datenbanken in der Cloud	14
2.3	Docker	15
2.3.1	Virtualisierung	15
2.3.2	Images	15
2.3.3	Containers	16
2.3.4	Volumes	16
2.3.5	Networks	17
2.3.6	Technologien	17
2.4	Data-Mining	19
3	Entwicklungszyklus der Bachelor-Thesis	22
3.1	Anforderung und Durchführbarkeit	24
3.2	Konzeption	25
3.2.1	Oracle-Datenbank	25
3.3	Prototyping	25
3.3.1	Docker-Desktop Installation	26
3.3.2	Client-Server Integration	29

3.3.3	Cloud Integration	33
3.4	Testszenarien	38
3.4.1	Verbindung zum SQL Developer	38
3.4.2	Datenbankbenutzer anlegen	42
3.5	Data-Mining-Lösung in der Praxis	44
3.5.1	SQL Developer	44
3.5.2	R zu Oracle	46
3.5.3	Oracle Machine Learning	46
4	Evaluation	48
4.1	Express Edition(XE) 18.4.0	48
4.1.1	Vorteile der benannten Lösung	49
4.1.2	Nachteile der benannten Lösung	49
4.2	Oracle Cloud Infrastructure	50
4.2.1	Vorteile der benannten Lösung	51
4.2.2	Nachteile der benannten Lösung	52
4.3	Gegenüberstellung	53
4.4	Bewertung auf der Grundlage der festgelegten Anforderungen	55
4.4.1	Offene Fragen	57
5	Fazit und Ausblick	58
5.1	Zusammenfassung	58
5.2	Aussichten	59
5.3	Schlussfolgerung	59
	Literaturverzeichnis	61
A	Abbildungsverzeichnis	1
B	Tabellenverzeichnis	2

Danksagung

An dieser Stelle möchte ich mich bei all denjenigen bedanken, die mich während der Anfertigung dieser Bachelor-Thesis unterstützt und motiviert haben. Zuerst gebührt mein Dank Prof. Dr.-Ing. Sabine Schumann, die meine Arbeit betreut und begutachtet hat. Für die hilfreichen Anregungen und die konstruktive Kritik bei der Erstellung dieser Arbeit möchte ich mich herzlich bedanken. Ebenfalls möchte ich mich bei meinen Freunden Max Wiebicke, Daniel Nisch, Finn Hansen, Alexandra Twerdochleb, Rouven Bardtke, Torben Schlüter, Daniel von Ahn, Markus Gross und Niklas Müller bedanken, die mir mit viel Geduld, Interesse und Hilfsbereitschaft zur Seite standen. Abschließend möchte ich mich bei meiner Oma bedanken, die mir mein Studium durch ihre Unterstützung ermöglicht hat und stets ein offenes Ohr für mich hatte.

Abstract

The purpose of this thesis is the comparison of different Oracle database implementations for the application of courses at the Hamburg University of Applied Sciences, under consideration of data mining methods with final evaluation. The tools used for the application are examined in detail and applied in practice. In particular, modern cloud architectures are compared with established data center structures. In addition, the characteristics of the various database solutions will be considered. This accompanies the conceptual work up to the prototypical application.

Zusammenfassung

Ziel dieser Arbeit ist der Vergleich verschiedener Oracle-Datenbankinstallationen für die Anwendung von Lehrveranstaltungen an der Hochschule für Angewandte Wissenschaften Hamburg unter Berücksichtigung von Data-Mining-Methoden mit abschließender Auswertung. Die für die Anwendung verwendeten Werkzeuge werden im Detail untersucht und in der Praxis angewendet. Dabei werden insbesondere moderne Cloud-Architekturen mit etablierten Rechenzentrumsstrukturen verglichen. Darüber hinaus werden die Eigenschaften der verschiedenen Datenbanklösungen betrachtet. Dies beinhaltet die konzeptionelle Arbeit bis hin zur prototypischen Anwendung.

1 Einleitung

1.1 Motivation

Laut Amazon ist Cloud Computing die bedarfsgesteuerte Bereitstellung von IT-Ressourcen über das Internet mit Pay-as-you-go-Preisen. Anstatt physische Rechenzentren und Server zu kaufen, zu besitzen und zu warten, können bei Bedarf von einem Cloud-Anbieter wie Amazon Web Services auf Technologiedienste wie Rechenleistung, Speicher und Datenbanken zugegriffen werden. [14] Vier der bekanntesten Marktteilnehmer Amazon, Microsoft, Google und Oracle bieten für Unternehmen alle ihre spezifischen Lösungen in der Cloud an und stellen damit eine Alternative zu Rechenzentren zur Verfügung, die zusätzlich zu der Software auch Betreuung der Hardware benötigen. Die Frage, ob sich diese Services jedoch für jedes Einsatzgebiet eignen, lässt sich nicht pauschal beantworten. Für Unternehmen wie Microsoft und SAP steht fest, dass ein Großteil ihres Umsatzes zukünftig im Cloud-Geschäft liegt [5]. Der Kunde hingegen hat ein großes Interesse daran, Geld zu sparen. Ob nur die Unternehmen davon profitieren oder auch der Kunde, der Kompetenz und Verfügbarkeit in fremde Hände legt, jedoch an Ressourcen wie Zeit und Geld spart, sind diskutabel, hochspannende Themen in der Welt der Informationstechnologie.

1.2 Ziel und Struktur der Arbeit

In dieser Arbeit wird ein Vergleich mehrerer Oracle 18 Datenbanksysteme im Hinblick auf Data Mining in der Hochschullehre erstellt. Dabei werden eine Cloud-Lösung, eine 18XE-Lösung auf einem lokalen Rechner und eine serverseitige Installation im Rechenzentrum der Hochschule gegenübergestellt. Sie lässt sich, zusammen mit der Einleitung, in fünf Kapitel gliedern. Ziel dabei ist es, die für die Hochschule beste Lösung zu finden in den Bereichen Einrichtung, Administration, Kosten und Bedienung.

Zu Beginn werden die Grundlagen thematisiert. Nachdem das aktuelle Kapitel die Motivation für die Arbeit dargelegt hat, werden im Anschluss die Grundlagen erör-

tert, die für diese Arbeit notwendig sind. Es wird ein Blick auf die Bestandteile und Werkzeuge geworfen, in dieser Arbeit zum Einsatz kommen und wie diese funktionieren. Dazu gehören grundlegende Bausteine wie relationalen Datenbanken, Cloud sowie Docker. Das darauffolgende Kapitel beschreibt den praktischen Teil, in dem Konzepte inklusive der praktischen Integration geschildert werden. Kapitel 4 ist das Evaluationskapitel, hier wird der Vergleich der unterschiedlichen Integrationsmöglichkeiten vorgenommen, um in Kapitel 5 ein Fazit zu ziehen und einen Ausblick geben zu können.

1.3 Hinweise zum Betriebssystem

Hinsichtlich der in der Praxis eingesetzten Hardware, wurde für diese Arbeit mit der Containervirtualisierungs-Software Docker gearbeitet. Diese, sowie weitere Tools, wurden auf einem MacBookPro 14,1 mit der Version MacOS 10.15.5 genutzt. Mögliche betriebssystemspezifische Probleme wurden geprüft, sind jedoch nicht gänzlich auszuschließen. Unterschiede aufgrund grafischer Darstellung sind ebenfalls möglich, stellen allerdings keinen Unterschied in der Funktion dar. Darüber hinaus obliegt es dem Benutzer, gegebenenfalls andere Hardware, sowie Software zu nutzen.

2 Grundlagen

Heutzutage, sowie in den vergangenen Jahren, sehen sich unter anderem Unternehmen mit den Problemen der Datenhaltung konfrontiert. Die Datenmengen wachsen auf eine enorme Menge an, ohne dass der Mehrertrag an Daten ausgenutzt werden konnte. In einer Zusammenarbeit mit der Hochschule für Angewandte Wissenschaften Hamburg wird der Einsatz einer Oracle-Datenbankumgebung realisiert, um die Forschung ungeachtet der steigenden Datenflut weiterhin leistungsfähig durchführen zu können. Für die Hochschule ist von Bedeutung, dass ein Lizenzmodell vorliegt, welches die kostenfreie Nutzung für Studenten ermöglicht, und sich zusätzlich auf die Anforderungen des Marktes sauber abbilden lässt.

Dabei ist eines der verfolgten Ziele, Daten zu entnehmen, diese bzgl. Muster zu untersuchen, Merkmale zu finden die in Abhängigkeiten stehen und daraus Vorhersagen zu treffen. Dieses Vorgehen ist unter dem Begriff Data-Mining bekannt und wird ebenfalls in dieser Arbeit behandelt. Die Einsatzmöglichkeiten sind allerdings weitaus umfassender und können deshalb je nach Bedarf variieren. Das Hauptaugenmerk oder auch das Ziel der Arbeit ist dabei vor allem den Vergleich aufzustellen, welche Versionen sich prinzipiell eignen und welche Alternativen der Markt bietet.

Um ein Projekt in dieser Größenordnung lauffähig umzusetzen, bedarf es für die beteiligten Akteure viel Einarbeitung in unterschiedliche Teilbereiche der Informatik. Eines davon sind Aufgaben der Datenbankadministration, wie der Inbetriebnahme und Konfiguration mehrerer Datenbankinstanzen, sowie das Sicherstellen der Verfügbarkeit von Datenbankfunktionalitäten und der Sicherheit. Eine andere Teilaufgabe stellt die Bereitstellung genannter Datenbank dar.

In der modernen Sprache der IT-Welt spricht man vom sogenannten Devops oder DevOps-Engineers, welches sich aus den Worten „Development“ und „IT Operations“ zusammensetzt. Die Kombination soll einen Prozessverbesserungsansatz in den Bereichen Softwareentwicklung und Systemadministration ermöglichen. Das verfolgte Ziel sei dabei eine schnelle Umsetzung von stabiler, hochwertiger Softwarelösung. Angefangen beim Konzept, bis hin zum Kunden oder Anwender. Eine teamorientierte Lösung, welche in der agilen IT-Welt angekommen ist und auch Teil dieser Arbeit ist. Die Betrachtung des gesamten Feldes macht diese Arbeit zu einem spannenden

Thema. Wo tauchen Grenzen bei der Architektur auf? Sind Unterschiede der Lizenzmodelle marginaler als vorher angenommen? Gibt es überhaupt eine Wahl bei diesem Projekt, oder steht die Lösung bereits fest? Alle diese Fragen inklusive der technischen Kniffe fördern den Horizont in technischer Hinsicht, als auch im problem-lösungsorientierten Ansatz. Das Rad wird in dieser Arbeit nicht neu erfunden, doch werden die Materialien auf den Prüfstand gestellt.

Für eine detaillierte Betrachtung benötigt es, sich zunächst ein Überblick über die Grundlagen zu verschaffen. Diese werden in folgendem Kapitel erörtert.

2.1 Relationale Datenbanken

Eine Datenbank ist eine Teilkomponente eines Datenbanksystems, das zusätzlich ein Datenbank-Managementsystem enthält. Von zentraler Bedeutung sind in diesem Zusammenhang die Datenhaltung und die in der Datenbank gespeicherten Daten sowie die Administration, Verwaltung oder Kontrolle innerhalb des Datenbank-Managementsystems. Unabhängig davon, ob die Datenbank eine objektorientierte oder eine relationale Architektur hat, muss jedes Datenbanksystem den folgenden Prinzipien entsprechen.

1. Dauerhaftigkeit. Persistente Speicherung von Daten über einen beliebig langen Zeitraum
2. Große Datenbestände. Das Datenbank-Management-System muss in der Lage sein, große Datenbestände mit seiner Technologie zu verarbeiten, beispielsweise durch Bufferpools.
3. Mehrbenutzersystem
4. Wiederherstellbarkeit. Einen Fehlerfall durch Zurücksetzen der Transaktionen oder Neustart beheben.
5. Ad-Hoc Abfragemöglichkeiten. Die Notwendigkeit, mithilfe einer Abfragesprache Daten zu manipulieren oder zu verwalten.

Das Modell der relationalen Datenbanken besteht seit 1970 und wird für die elektronische Datenverwaltung verwendet. Relationale Datenbanken sind seit Jahrzehnten ein Hauptbestandteil der Software-Entwicklung[8]. Diese folgen zusätzlich zu den fünf Datenbank-Grundsätzen, weitere Regeln, welche sie als relationale Datenbank kennzeichnet.

1. Tabellen. Daten und deren Beziehung werden tabellarisch dargestellt. In den Spalten werden die Attribute dargestellt, die Datensätze werden als Tupel bezeichnet.
2. Primärschlüssel. Die Merkmale oder Kombinationen müssen ein Tabellen-Tupel eindeutig identifizieren.

3. Mengenorientierte Operatoren. Die Tabellen können unter Anwendung der Mengenoperationen Vereinigung, Durchschnitt, Differenz und kartesisches Produkt kombiniert werden. Die Menge der Merkmale muss übereinstimmend sein und die gleichen Wertebereiche haben.
4. Relationenorientierte Operatoren. Dazu zählen Hilfsmittel, mit deren Hilfe Tabellen von Filteroperatoren modifiziert werden können. Diese Selektion ermöglicht es z.B., Tupel nach bestimmten Eigenschaften auszuwählen. Es gibt außerdem Verbund- und Divisionsoperatoren.
5. Viewkonzept. Erlaubt verschiedene Sichten auf einzelne Tabellen zu legen.

Sind alle der oben genannten Punkte innerhalb der Datenbanksoftware erfüllt, so spricht man von einer relationalen Datenbank. Die Zuordnung der in einem RDBS zu speichernden Daten basiert auf einem Entity Relationship Model (ERM). In diesem Modell werden die Daten mit Hilfe von Tabellen und Relationen modelliert. Die Tabellen müssen einer sogenannten Normalform genügen. Der logische Entwurf erfolgt mit Hilfe einer Daten-Definitionssprache (DDL). Diese ist standardmäßig in SQL integriert. Wenn die Daten persistent in einer Datenbank gespeichert sind, erlauben Abfrage- und Manipulationssprachen verschiedene Operationen, wie das Ausblenden oder Verknüpfen von Tabellen. Bekannte kommerzielle Datenbanksysteme werden von Oracle und Microsoft vertrieben. Diese zeichnen sich vor allem durch ihre Optimierungs- und Erweiterungsmöglichkeiten, aber auch durch ihren hohen Preis aus. Als Alternative steht die Open-Source-Datenbank zur Verfügung. Als die wohl bekannteste gilt MySQL.

2.1.1 In-Memory

Bei IN-Memory-Datenbanken werden die Informationen nicht auf der Festplatte, sondern im Arbeitsspeicher gespeichert. Das Lesen der Daten war in der Vergangenheit der Engpass bei der Berechnung von Datenbankabfragen, da das Auslesen von einer Festplatte sehr langsam ist, die Daten aber sehr schnell in der CPU verarbeitet werden konnten. Dieses Problem wird durch das Verschieben der Daten in den Hauptspeicher nicht vollständig beseitigt, aber es kann stark eingeschränkt werden, da das Lesen aus dem Hauptspeicher etwa 10.000 Mal schneller ist als das Lesen von der Festplatte[13]. In IN-Memory-Datenbanken werden die Daten nicht zusammengefasst, die ursprünglichen "Transaktionsdaten" werden gesammelt und dem Benutzer

in Direktabfragen zur Verfügung gestellt. Darüber hinaus ist es wesentlich einfacher, die unstrukturierten Daten in die Queries zu integrieren, da es nicht notwendig ist, die Daten in vordefinierten Datenformaten zu speichern. IN-Memory-Datenbanken minimieren daher nicht nur die Zugriffszeiten, sondern auch die Datenbewegungen und vereinfachen die Nutzung unstrukturierter Daten.

2.1.2 Zeilenorientierte Datenbanken

In zeilenorientierten Datenbanken sind die Zeilen Tupel für Tupel gespeichert. Dies bietet Vorteile für das Einfügen einer Zeile, da das komplette Tupel auf einmal geschrieben werden kann. Darüber hinaus sind Zugriffe auf zeilenorientierte Datenbanken schneller, wenn mehrere Spalten einer Zeile gelesen werden. Operationen, die eine Spalte von mehreren Sätzen lesen, sind jedoch deutlich langsamer als bei spaltenorientierten Datenbanken.

2.1.3 Autonome Datenbanken

Eine autonome Datenbank ist eine Cloud-Datenbank, die maschinelles Lernen nutzt, um menschliche Arbeit zu reduzieren. Die Datenbank ist vollständig vorkonfiguriert mit einer Mehrzahl von verfügbaren Arbeitslasttypen. Die Autonome Datenbank übernimmt die Erstellung der Datenbank sowie die folgenden Wartungsaufgaben:

1. Sichern der Datenbank
2. Patchen der Datenbank
3. Aktualisierung der Datenbank
4. Tuning der Datenbank

Genutzt wird diese Art der Bezeichnung insbesondere bei Oracle, welche die Art des Betriebs insbesondere fördert. Und im Laufe dieser Arbeit noch von Bedeutung ist.

2.2 Cloud

Zur Definition des Cloud Computing wird häufig die Definition der amerikanischen Standardisierungsbehörde NIST (National Institute of Standards and Technology) herangezogen. Darin wird Cloud Computing als Zugriff auf gemeinsam genutzte Computing-Ressourcen wie Rechenleistung, Anwendungen, Server und viele andere beschrieben. Dabei ist wenig Organisierungsaufwand oder Interaktion mit dem Serviceprovider notwendig[18].

1. *On-demand self-service*. Die Zuteilung der benötigten Ressourcen, wie Rechenleistung oder Speicherplatz, erfolgt automatisch und ohne Zutun eines Benutzers.
2. *Broad network access*. Die Dienste sind über das Netzwerk zugänglich, sowie plattform- und kundenunabhängig. So kann der Zugriff mit mobilen Endgeräten, oder mit einem konventionellen Computer ermöglicht werden.
3. *Resource pooling*. Die gesamten Ressourcen des Dienstleisters dienen den Erfordernissen einer großen Anzahl von Kunden. Physikalische und virtuelle Ressourcen werden dem Kunden dynamisch und standortunabhängig bereitgestellt.
4. *Rapid elasticity*. Die dem Benutzer zur Verfügung gestellte Bandbreite ist abhängig von seinem Bedarf. Es erscheint dem Benutzer also, dass seine Geschwindigkeit uneingeschränkt ist.
5. *Measured service*. Die Ressourcenverwendung wird kundenspezifisch überwacht und gegebenenfalls optimiert, oft unter Anwendung von Preismodellen wie Pay-per-use, was eine exakte Aufgliederung der Nutzung voraussetzt.

Cloud-Computing-Angebote sind in verschiedenen Kategorien aufgeteilt: Infrastructure as a Service (IaaS), Platform as a Service (PaaS) und Software as a Service (SaaS)[17].

In den nun folgenden Unterpunkten werden die einzelnen Servicemodelle beschrieben.

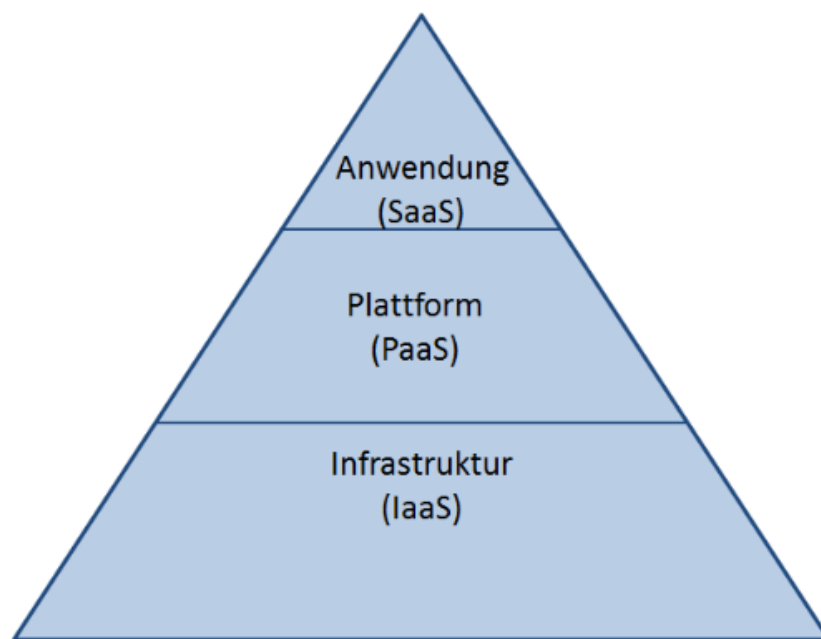


Abbildung 2.1: Architektur der Servicemodelle

2.2.1 Infrastructure as a Service

IaaS sind Basisdienstleistungen, die eine virtuelle Infrastruktur, wie Datenspeicherung oder Rechnerleistung zur Verfügung stellen. Anstatt kostspielige Hardware zu kaufen, können einfach die Serverkapazität gemietet werden, die für die anstehenden Aufgaben benötigt wird. Der Vorteil ist, dass dadurch Ressourcen eingespart werden, da weniger Kapazität angemietet und bei sinkender Nachfrage kurzfristig genutzt werden kann. Obwohl sich die Server unzugänglich in den Rechenzentren der Anbieter befinden, obliegt die Verwaltung und Konfiguration den Benutzern selbst.

2.2.2 Platform as a Service

Für die Softwareentwicklung in der Cloud können die Kunden vordefinierte Plattformen anmieten. Dies wird häufig für die Entwicklung von Apps genutzt. Bei PaaS übernimmt der Anbieter den Betrieb und die Verwaltung der zugrunde liegenden Infrastruktur, sodass sich die Mitarbeiter in einer optimierten Entwicklungsumgebung voll auf die Programmierung konzentrieren können.

2.2.3 Software as a Service

SaaS ist das, was die meisten Menschen mit der Cloud in Verbindung bringen. SaaS umfasst cloudbasierte Anwendungssoftware, auf die der Endbenutzer über das Internet von einer Vielzahl von Endgeräten aus zugreifen kann. Die Online-Dienste werden gewissermaßen gemietet: Der Anbieter kümmert sich um die gesamte Infrastruktur und Software-Wartung. Man muss also nicht über technisches Verständnis verfügen, um die Vorteile von SaaS-Cloud-Angeboten nutzen zu können. Alles ist bereits vorkonfiguriert.

Ein bekanntes Beispiel für Unternehmens-SaaS sind Google Apps und Microsoft Office 365, die in erster Linie die Zusammenarbeit und Produktivität von Teams fördern. Für Privatanwender sind Cloud-Storage-Anbieter wie Dropbox, Google Drive und OneDrive besonders relevant.

2.2.4 Datenbanken in der Cloud

Auch für moderne Datenbank-Technologien kann Cloud Computing viele Vorteile bieten. Durch die dynamische Ressourcenzuweisung können Kosten eingespart werden, da speziell Datenbanksysteme zwar eine hohe Spitzenauslastung haben, im Normalbetrieb aber tendenziell weniger Ressourcen beanspruchen. Normale Serversysteme verbrauchen weiterhin Energie und verursachen somit Kosten. Bei hoher Auslastung kann die Rechenleistung schnell angepasst werden, sodass es zu keinen Verzögerungen bei Abfragen kommt. Darüber hinaus können die Lizenzkosten minimiert werden, da die Anwendungen zentral in der Cloud laufen und nicht mehr auf mehreren Clients installiert werden müssen.

2.3 Docker

2.3.1 Virtualisierung

Unter der Virtualisierung versteht man im Wesentlichen die Zusammenführung mehrerer Hardwareserver auf ein gemeinsames Hardwaresystem. Die bisher einzelnen Hardwareserver werden mit ihren eigenen Diensten auf ein Host-System gebracht und arbeiten dort in einer geschlossenen Umgebung so, als befänden sie sich in der ursprünglichen Hardware-Umgebung. Diese nutzen nicht mehr ihre eigenen Hardware-Festplatten oder Hardware-Netzwerkkarten, sondern teilen sie mit anderen Servern. Das spart nicht nur Stromkosten, sondern erhöht die Effizienz. Die Hosts können voll ausgelastet werden, was insgesamt weniger Hardware erfordert. Server können auch leichter kleinere Aufgaben ausführen und würden keinen eigenen physischen Raum benötigen, da die Ressourcen vom Host-Betriebssystem gekapert werden. Die virtuellen Maschinen erhalten eigene partitionierte Bereiche auf der Festplatte, reservieren RAM und teilen sich CPUs. Alles wird vom Hypervisor, dem Verwalter der Ressourcenverteilung, kontrolliert.

2.3.2 Images

Im Falle von Docker gibt es eine weitere Abstraktion von den ursprünglichen Hardware-Servern. Diese sind in einzelne Teile des Systems unterteilt und stehen in Docker als Images oder Layer zur Verfügung.

Ein Ubuntu-Kernel als Beispiel wird zur untersten Schicht eines Images. Oberhalb dieser Layer wird eine weitere Ebene platziert, in der sich bestimmte Dateien zur Ausführung einer Anwendung befinden. Alles zusammen ergibt ein Image. Die letzte Schicht kann beispielsweise eine Java-Applikation sein, die auf diese Dateien zugreift und durch Ubuntu zum Ausführen gebracht wird. Alles in allem entsteht so ein Image, das nun auf verschiedenen Systemen frei lauffähig ist. Die einzige Voraussetzung ist ein Docker-Host, der die Images und Ebenen verwaltet. Wenn es nun Layer in einem Container gibt, die mehrere Anwendungen verwenden, wie z.B. eine Ubuntu 15.04, können andere Images aus einem weiteren Container denselben Layer verwenden, ohne neue Ressourcen auf der Festplatte bereitstellen zu müssen. Insgesamt gibt es zwei Hauptkomponenten eines Containers. Die Image-Ebene und die Container-Ebene. Die Image-Ebenen sind schreibgeschützt. Wenn Dateien von

der Containerschicht aus beschrieben werden, werden sie auf die Containerschicht kopiert. Die Image-Ebene bleibt unangetastet, wie zum Beispiel das Ubuntu-Image. Die Containerschicht hingegen enthält den einzigen Bereich, in dem die Daten des Containers selbst noch geschrieben werden können. Diese Daten werden vollständig gelöscht, wenn ein Container verworfen wird, es sei denn, sie werden in einem gesonderten Volume gespeichert.

2.3.3 Containers

Der Container selbst ist an sich nur ein Prozess, der ausgeführt wird und auf der Grundlage des Images bestimmte Aufgaben erfüllt. Der Container speichert seine Daten in einer Containerschicht. Wenn der Container gelöscht wird, werden auch die Daten innerhalb der Containerschicht gelöscht. Container arbeiten in der Regel eigenständig und speichern die Daten in sich selbst. Mehrere Container können auf die gleichen Images zugreifen, da sie schreibgeschützt sind. Die Containerdaten werden sozusagen Schicht für Schicht durchforstet. Wenn ein Container auf eine Datei in einer tieferen, nicht beschreibbaren Schicht des Bildes schreiben muss, kopiert der Container diese Datei in die oberste Schicht. Infolgedessen sieht der Container beim erneuten Zugriff auf die tieferliegende Datei nicht die darunter liegende, da die Datei in der obersten Ebene die erste in der Richtung von oben nach unten ist.

2.3.4 Volumes

Volumes sind die optimale Möglichkeit, Daten eines Containers persistent zu speichern. Sie erlauben auch den Datenaustausch zwischen verschiedenen Containern. Wenn ein Container angelegt und gestartet wird, werden drei Parameter angegeben, die seine Funktionsweise erklären. Der erste ist der lokale Dateipfad, also der Ordner, der in dem Container integriert ist. Der zweite Parameter gibt den Speicherort im Container an, an dem der betreffende Ordner integriert wird. Der dritte Parameter gibt an, ob ein Volume nur lesend oder auch schreibend sein kann. Im Container ist der eingebaute Ordner als ein lokaler Ordner auf dem Dateisystem zu verstehen. Wenn man in Richtung der Ebene schaut, würde sich in der obersten Ebene ein Volume befinden. Jedes Volume überschreibt also alle anderen Daten in einem Container. Zusätzlich können bei der Verwendung der Volumes verschiedene Treiber verwendet werden. Die Voreinstellung ist "lokal", was voraussetzt, dass sich der angegebene

Ordner lokal auf dem Host-Dateisystem befindet. Welcher Treiber verwendet werden soll, wird beim Erstellen eines Containers explizit angegeben.

2.3.5 Networks

Es gibt verschiedene Netzwerktreiber in Docker, sowie im Bereich der Volumes. Diese unterscheiden sich in ihrer Funktionalität und Zugänglichkeit außerhalb des Docker-Containers. "BridgeNetzwerke werden hauptsächlich für die Kommunikation innerhalb eines Docker-Hosts verwendet. Sie können Ihre eigenen Bridge-Netzwerke erstellen, z.B. ein sogenanntes "Web-Server-Netzwerk,, das Web-Server und Datenbanken enthalten kann. Da sie sich im selben Bridge-Netzwerk befinden, können sich die Container bereits gegenseitig sehen. "Host,,-Netzwerke teilen sich die gleiche IP-Adresse mit dem Docker-Host. Wenn der Container beispielsweise einen Dienst auf Port 80 anbietet, ist der Container automatisch über die IP-Adresse des Docker-Hosts auf diesem Port erreichbar.

Vereinfacht ausgedrückt, werden "Overlay,,-Netzwerke für die Kommunikation zwischen mehreren Docker-Hosts verwendet, z.B. wenn sie in einem Docker-Schwarm verbunden sind. Wenn die Container innerhalb der Hosts mit einem Container des anderen Hosts kommunizieren müssen, wird der OverlayNetzwerktreiber zu diesem Zweck verwendet.

2.3.6 Technologien

Ein bedeutender Teil von Docker sind die Linux-Namespaces. Sie werden verwendet, um Teilprozesse eines Prozesses zu erstellen, die voneinander isoliert sind und sich gegenseitig nicht sehen können. Es gibt privilegierte Prozesse, wie den Docker-Daemon, der alle anderen Unterprozesse verwalten und erstellen kann. Neben den Prozess-IDs gibt es Namespaces für einige Bestandteile des Docker Systems, wie für Netzwerk-Interfaces.

Libnetwork ist eine Bibliothek, die die Namespaces des Netzwerks nutzt und erweitert. Mit Libnetwork werden die verschiedenen Netzwerktreiber, die für Docker verwendet werden, initialisiert und bereitgestellt. Libnetwork bietet für Docker auch die Einstellungen für die Hostnamen, die IP-Adressen und das Port-Publishing für die Container, sowie der Freischaltung von Ports im Host an. Sozusagen eine Bibliothek, die alle möglichen Netzwerk-Komponenten in einer Bibliothek zur Verfügung stellt.

Linux-CG-Gruppen stellen die Systemressourcen für die Prozesse zur Verfügung. So werden im Docker-Beispiel nur bestimmte CPU-Ressourcen für die Container zur Verfügung gestellt. Die Ressourcen werden unter den Containern aufgeteilt und können leicht verwaltet werden. Dies gilt auch für den Speicher und auch für das Netzwerk. Für die Anwendungen in den CG-Gruppen können Gruppen erstellt werden, die diese Ressourcen dann innerhalb der CG-Gruppe gemeinsam nutzen können. Im Docker weisen die CGroups den Containern Ressourcen zu und schränken die Ressourcen gegebenenfalls ein.

Docker verwendet verschiedene Dateisysteme, die bereits vorhanden sind, und durch Erweiterungen hinzugefügt werden können. Es gibt bereits AUFS, BTRFS, VFS und Devicemapper. Im Prinzip wird im Namensraum jedes Containers ein separates Dateisystem erstellt und der Root-Pfad des Prozesses gespeichert. Dadurch wird verhindert, dass ein Container auf die Dateien auf dem Hostsystem zugreift.

RunC ist eine Kombination aus UnionFS, CGroups und Namespaces. Diese werden verwendet, um ein einheitliches Containerformat zu erzeugen, dass es erlaubt, die Docker-Container vordefiniert und einheitlich zu erstellen.

ContainerD ist die erweiterte Bibliothek von RunC. Hier wird versucht, so wenige Komponenten wie möglich zu integrieren. So werden am Ende nur jene Komponenten und Bibliotheken verwendet, die wirklich benötigt werden.

2.4 Data-Mining

Data Mining ist eine Technik, die bisher unbekannte Beziehungen in Daten aufdeckt[15]. Das World Wide Web enthält eine rapide wachsende Anzahl von Seiten und damit verbundenen Daten und Informationen. Die Menge an verfügbaren Daten verdoppelt sich in immer kürzeren Abständen. Der weltweit tätige US-amerikanischer Einzelhandelskonzern WalMart führt eine Warenkorb-Analyse durch, die 20 Millionen Transaktionen pro Tag einer Datenbank mit einer Kapazität von 24 Terabyte erfordert. Für die Auswertung solcher enormen Datenmengen reicht das simple Durchschauen nicht mehr aus[3].

Durch leistungsfähige Computer sind wir nun nicht nur in der Lage, diese Daten zu speichern, sondern wir können diese Datenmengen auch analysieren und somit interessante Informationen generieren. Das Kaufverhalten von Kunden kann so analysiert werden, um Abhängigkeiten zwischen Produkten zu finden. Wer Windeln kauft, nimmt häufig auch Bier mit[3]. Dieses und anderes Wissen kann zu einen entscheidenden Wettbewerbsvorteil führen.

Die wichtigsten Eigenschaften von Data Mining sind:

1. Automatische Entdeckung von Mustern
2. Vorhersage wahrscheinlicher Ergebnisse
3. Erstellung von verfolgbaren Informationen
4. Fokus auf große Datensätze und Datenbanken

Viele Formen des Data Mining sind prädiktiv. Beispielsweise kann ein Modell das Einkommen auf der Grundlage von Bildung und anderen demografischen Faktoren vorhersagen. Vorhersagen haben eine zugehörige Wahrscheinlichkeit (Wie wahrscheinlich ist es, dass diese Vorhersage zutrifft?). Vorhersagewahrscheinlichkeiten werden auch als Konfidenz bezeichnet (Wie zuversichtlich kann ich sein, dass diese Vorhersage zutrifft?)

Wichtig ist es zu betonen, dass durch Data Mining entdeckte prädiktive Beziehungen keine kausalen Beziehungen sind. Diese Informationen können verwendet werden, um eine Marketingstrategie zu entwickeln. Außerdem entdeckt Data Mining ohne eine Anleitung nicht automatisch Informationen. Die Muster sind sehr unterschiedlich, je nachdem, wie das Problem formuliert wird.

Die folgende Abbildung veranschaulicht die Phasen und den iterativen Charakter eines Data-Mining-Projekts. Der Prozessablauf zeigt, dass ein Data-Mining-Projekt nicht aufhört, wenn eine bestimmte Lösung bereitgestellt wird. Die Ergebnisse des Data Mining lösen Fragen aus, die wiederum zur Entwicklung weiterer Modelle verwendet werden können.

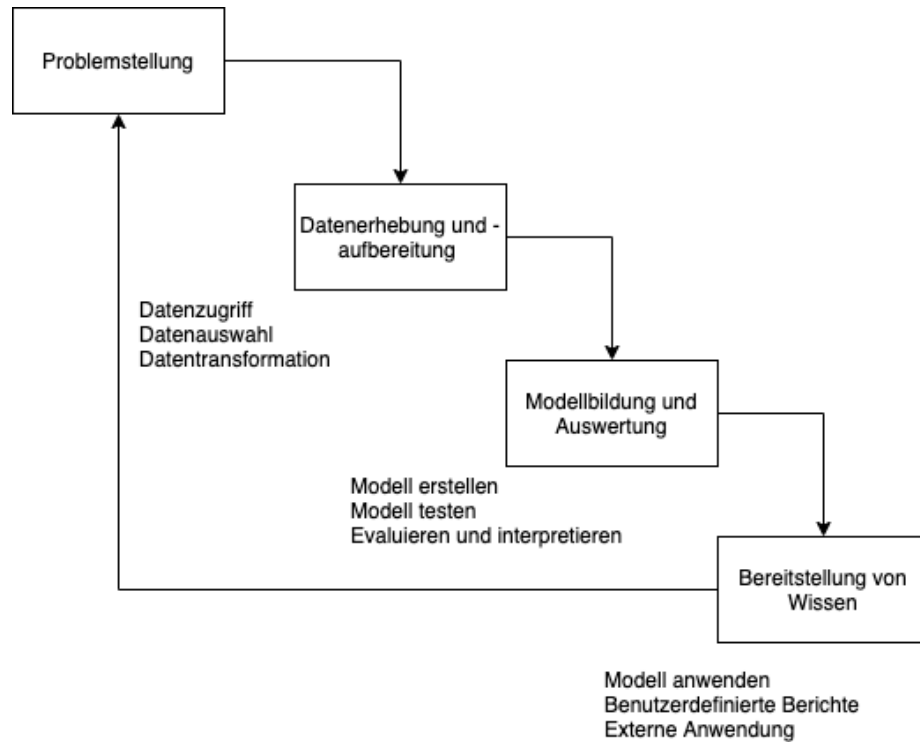


Abbildung 2.2: Der Data-Mining-Prozess

In dieser ersten Phase liegt der Schwerpunkt auf dem Verständnis der Projektziele und -anforderungen. Sobald das Problem aus der Geschäftsperspektive spezifiziert wurde, kann es als Data-Mining-Problem formuliert werden. Eine solche Formulierung könnte beispielsweise lauten "Welche Kunden werden das Produkt am ehesten kaufen?".

Bevor das Modell erstellt werden kann, müssen die Daten zusammengestellt werden, die wahrscheinlich Beziehungen zwischen Kunden enthalten.

Die Phase der Datenerhebung und -aufbereitung umfasst die Sammlung und Untersuchung von Daten. Schaut man sich die Daten genauer an, kann man feststellen, wie das Problem zu lösen ist. Außerdem werden Entscheidungen gefällt, ob einige der Daten entfernt oder zusätzliche hinzugefügt werden müssen. Dies ist auch der Zeit-

punkt, um Datenqualitätsprobleme zu identifizieren und nach Mustern in den Daten zu suchen.

Die Datenvorbereitungsphase umfasst alle Aufgaben, die mit der Erstellung der Tabelle oder des Views zu tun haben, die zum Aufbau des Modells verwendet werden. Zusätzlich können neue berechnete Attribute hinzugefügt werden.

In dieser Phase der Modellbildung und Evaluierung werden verschiedene Modellierungstechniken gewählt und angewendet, außerdem werden die Parameter auf optimale Werte kalibriert. Bei der vorläufigen Modellerstellung ist es oft sinnvoll, mit einem reduzierten Datensatz zu arbeiten, da der endgültige Datensatz Millionen von Zeilen enthalten kann.

Überdies ist es an der Zeit zu bewerten, wie gut das Modell das ursprünglich festgelegte Geschäftsziel erfüllt wird. Wenn das Modell Kunden vorhersagen soll, die wahrscheinlich ein Produkt kaufen werden, unterscheidet es dann ausreichend zwischen den Klassen? Kann das Modell durch Hinzufügen von Textdaten verbessert werden? Sollten transaktionale Daten, wie z.B. Käufe, einbezogen werden? Sollten Kosten im Zusammenhang mit falsch positiven oder falsch negativen Ergebnissen in das Modell einbezogen werden?

In der Bereitstellungs-Phase können aus Daten Erkenntnisse und umsetzbare Informationen abgeleitet werden.

Das Bereitstellen kann Scoring (die Anwendung von Modellen auf neue Daten), die Extraktion von Modelldetails (z. B. die Regeln eines Entscheidungsbaums), oder die Integration von Data-Mining-Modellen in Anwendungen, Data-Warehouse-Infrastrukturen oder Abfrage- und Berichtstools umfassen.

3 Entwicklungszyklus der Bachelor-Thesis

Die vorgestellten Technologien werden nun für die Entwicklungsphase in der Praxis weiter untersucht. Dazu werden unterschiedliche Ausführungen der Oracle Datenbank in Betrieb genommen, außerdem werden, für die in Kapitel 4 folgende Evaluation, Basisfunktionalitäten der Datenbank überprüft. Angefangen bei der Anforderung, welche die Hochschule stellt und der Machbarkeit, bei der geprüft wird, ob ein Projektziel in Einklang mit den wissenschaftlichen Erkenntnissen, rechtlichen oder wirtschaftlichen Rahmenbedingungen des jeweiligen Fachgebiets steht[16]. Dies soll von der Durchführung solcher Projekte abhalten, deren Misserfolg von Anfang abzusehen ist. Anschließend werden die unterschiedlichen Konzeptmodelle in verschiedene Prototypen überführt und dabei der Entwicklungsprozess dokumentiert. Beendet wird die Entwicklung mit einer Testphase, in der die Kriterien gegen den Prototypen getestet werden.

In der nun folgenden Abbildung sind die Schritte festgehalten für einen erfolgreichen Entwicklungszyklus eines Datenbankbetriebs. Aufgrund der Lesbarkeit dieser Arbeit wird auf die genaue Reihenfolge der Grafiken verzichtet.

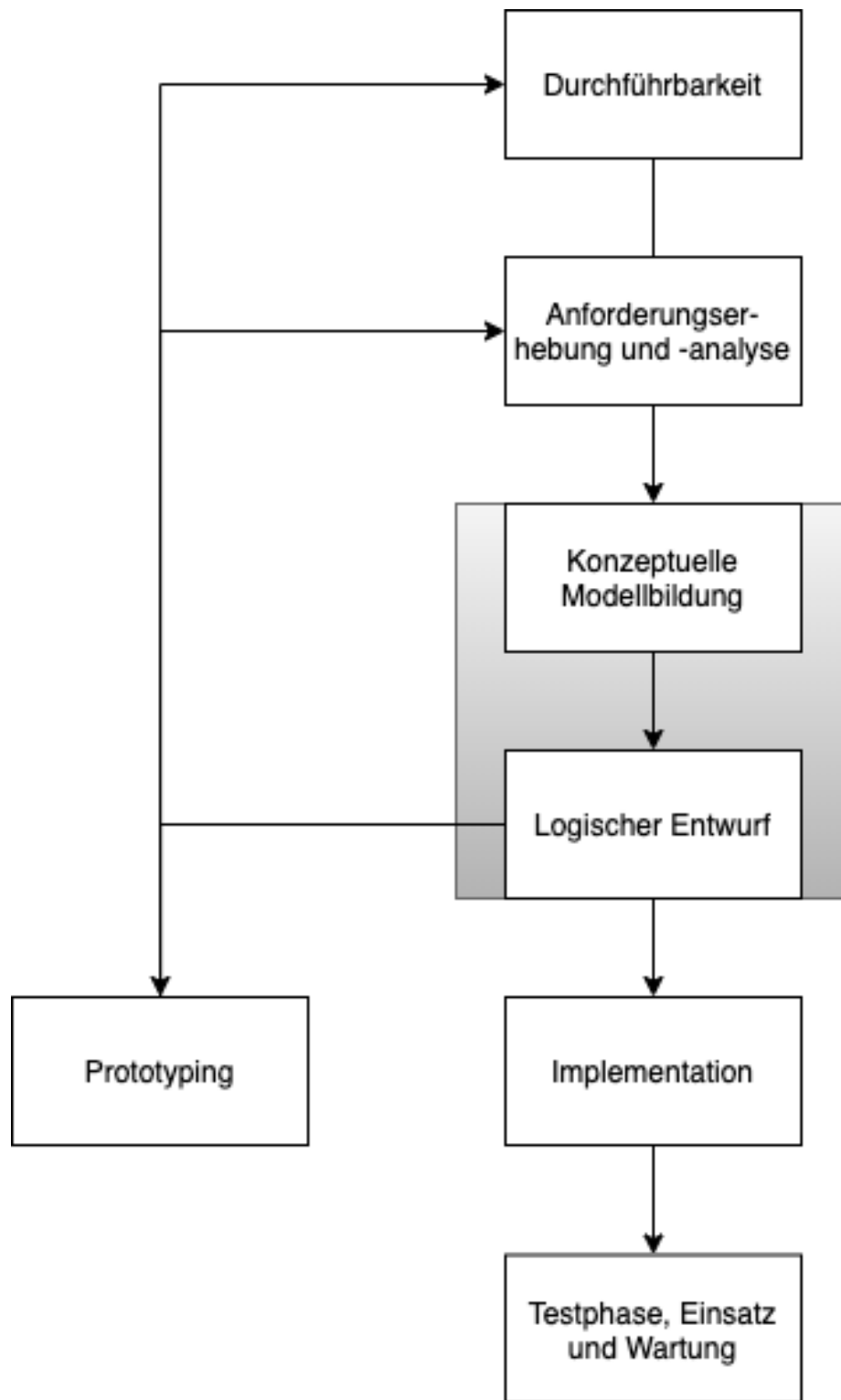


Abbildung 3.1: Entwicklungszyklus eines Datenbankbetriebs

3.1 Anforderung und Durchführbarkeit

Für eine differenziertere Beurteilung sind die Anforderungen in zwei Kategorien aufgeteilt. In dieser Ausarbeitung wird zwischen den kritischen Anforderungen und den weniger kritischen Voraussetzungen unterschieden. Für einen erfolgreichen Projektabschluss ist entscheidend, ob die kritischen Anforderungen umsetzbar sind.

Kritisch

1. Die bisherige Variante des Datenbankverwaltungssystems soll durch eine zeitgemäße Version des DBMS ersetzt werden.
2. Das verwendete DBMS soll kostenfrei erhältlich sein und die Wartung und Pflege soll ebenfalls keine weiteren Kosten erzeugen.
3. Für Lehrzwecke muss die Möglichkeit bestehen, eine Vielzahl von Nutzern zu erstellen, die Schreib- und Leserechte auf das DBMS besitzen.
4. Die gewählte Version soll die Data-Mining verfahren ermöglichen.
5. Der Dienst sollte so ausfallsicher wie möglich sein, oder es sollte problemlos möglich sein, den Dienst neu zu starten.

Weniger kritisch

1. Der administrative Aufwand seitens der Hochschule. Dies umfasst insbesondere die Instandhaltung und Einrichtung in die bestehende Struktur.
2. Die Datenbank muss über ausreichend Speicherkapazität verfügen, um auch eine Vielzahl unterschiedlicher Veranstaltungsdaten in der Datenbank zu speichern.
3. Die Leistung bei der angestrebten Umsetzung darf nicht merklich zu gering ausfallen.
4. Weniger kritisch sind außerdem Sicherheitsmaßnahmen in Bezug auf besonders schützenswerte Daten.

3.2 Konzeption

3.2.1 Oracle-Datenbank

Um DevOps-Anwendern die Installation, Konfiguration und Einrichtung der Systemumgebung zu vereinfachen, werden von Oracle einige sogenannte Einzelinstanzen bereitgestellt, die sich besonders für serverseitige Anwendungen eignen oder lokal auf den einzelnen Clients eingesetzt werden.

Mit Berücksichtigung der kritischen Anforderungen, die zuletzt festgesetzt worden sind, kommt nur eine Version in Frage, die sowohl zeitgerecht, als auch kostenfrei ist: Die Oracle Database 18c (18.4.0) Express Edition (XE), die in der nachstehenden Liste gekennzeichnet ist. [11] Außerdem kann der Liste entnommen werden, welche Versionen noch zur Verfügung stehen, unter der Berücksichtigung, dass entsprechende Version in einem Docker Container lauffähig sind.

Oracle Database 19c (19.3.0) Enterprise Edition and Standard Edition 2

Oracle Database 18c (18.4.0) Express Edition (XE)

Oracle Database 18c (18.3.0) Enterprise Edition and Standard Edition 2

Oracle Database 12c Release 2 (12.2.0.2) Enterprise and Standard Edition 2

Oracle Database 12c Release 1 (12.1.0.2) Enterprise and Standard Edition 2

Oracle Database 11g Release 2 (11.2.0.2) Express Edition (XE)

Ein detaillierter Einblick in den Funktionsumfang der Express Edition, sowie die Vor- und Nachteile, werden in Kapitel 4 vermittelt.

Darüber hinaus betreibt Oracle auch eine Cloud, die ihre Produktpalette zur Verfügung stellt. Versionen unter 18c entfallen, ebenso wie die technischen Voraussetzungen für das Hosten.

3.3 Prototyping

Für die Integration der Oracle Datenbank 18c (18.4.0) Express Edition wird im Rahmen dieser Arbeit die komplette Anwendung in einem Docker-Image abgebildet. Im Hinblick auf die hohe fachliche und technische Komplexität von Docker, die sich für einen eigenständigen Kurs anbietet, wird ein Hilfsmittel eingeführt, das für den Einsatz dieser Anwendung von Vorteil ist.

3.3.1 Docker-Desktop Installation

Unter Windows

Für die Installation unter Windows wird die Datei Docker Desktop Installer.exe benötigt, die auf der Website des Docker-Hubs erhältlich ist. Es ist darauf zu achten, dass die WSL 2-Funktionen während der Installationsroutine aktiviert sind. Bleibt das Walsymbol in der Statusleiste stehen, ist Docker Desktop betriebsbereit und von jedem Terminalfenster aus zugänglich.



Abbildung 3.2: Windows Docker

Unter Mac

Für die Installation unter Mac wird die Datei Docker.dmg benötigt, die auf der Website des Docker-Hubs erhältlich ist. Durch Öffnen des Installationsprogramms Docker.dmg, lässt sich die Installationsdatei in den Ordner Programme verschieben. Das Docker-Menü in der oberen Statusleiste zeigt an, dass Docker Desktop ausgeführt wird und von einem Terminal aus zugänglich ist.



Abbildung 3.3: Windows Docker

Portainer Installieren

Portainer besteht aus zwei Elementen, dem Portainer-Server und dem Portainer-Agenten. Beide Elemente laufen als Lightweight Docker-Container auf einer Docker-Engine, oder innerhalb eines Swarm-Clusters. Aufgrund der Beschaffenheit von Docker gibt es viele mögliche Einsatzszenarios, von denen jedoch im Folgenden die gängigsten im Detail beschrieben werden. Bevor die aufgelisteten Kommandos ausgeführt werden, ist zu beachten, dass es sich hier um die Installation für die Standalone-Version von Portainer handelt.

Unter Linux/Mac verwenden wir folgende Eingaben für die Einrichtung:

```
$ docker volume create portainer_data
$ docker run -d -p 8000:8000 -p 9000:9000 --name=portainer
--restart=always -v /var/run/docker.sock:/var/run/docker.sock
-v portainer_data:/data portainer/portainer
```

Unter Windows verwenden wir folgende Eingaben für die Einrichtung: (Hinweis auf mindestens Windows 1803 oder neuer)

```
$ docker volume create portainer_data
$ docker run -d -p 8000:8000 -p 9000:9000 --name portainer
--restart always -v \\.\pipe\docker_engine:\\.\pipe\docker_engine
-v portainer_data:C:\data portainer/portainer
```

Nach Ausführung der obigen Kommandos werden unter der Adressierung des jeweiligen Client unter Hinzufügen des angehängten Ports 9000 die nachstehenden Darstellungen angezeigt.

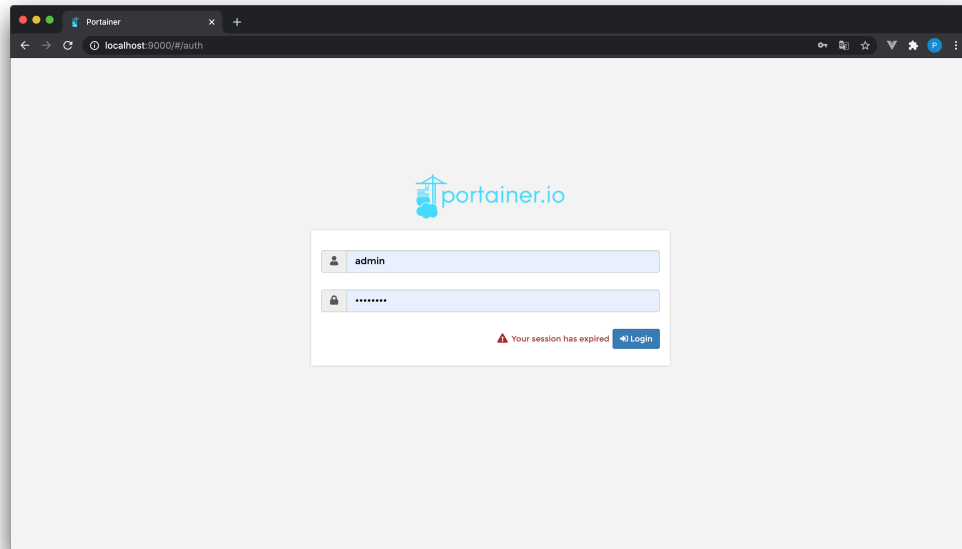


Abbildung 3.4: Login für portainer.io

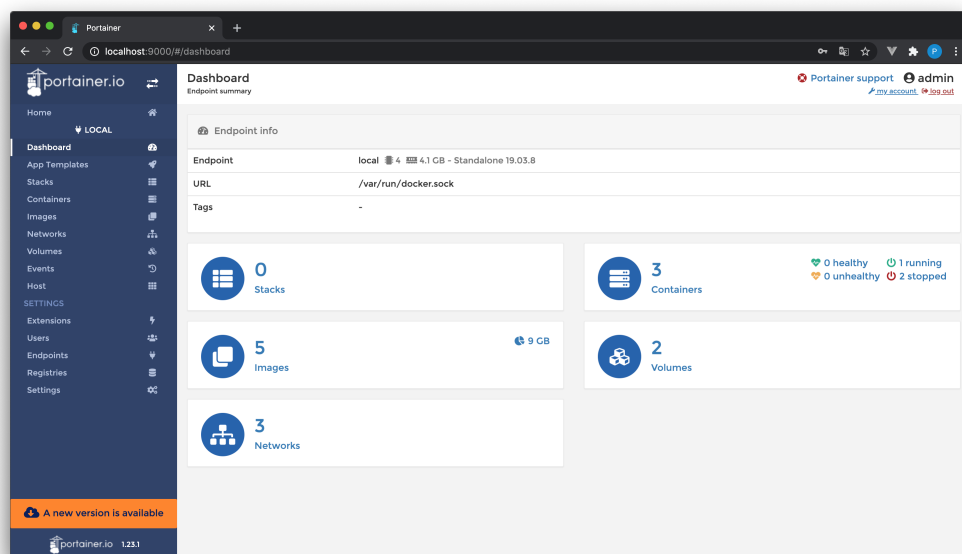


Abbildung 3.5: Dashboard von portainer.io

3.3.2 Client-Server Integration

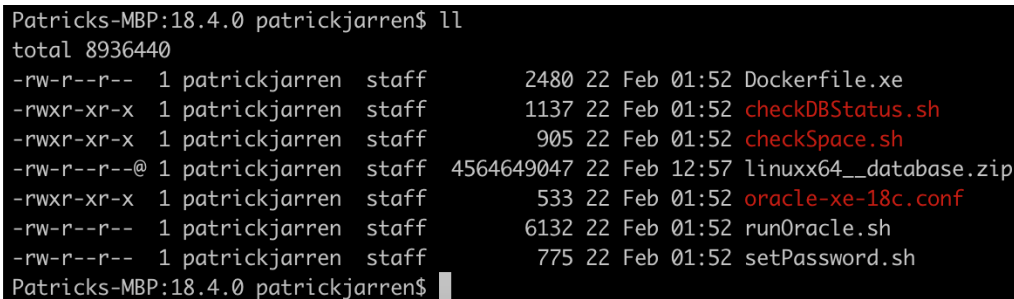
Die Installation erfolgt sowohl für die Server-, als auch für die Clientseite auf die gleiche Weise, wobei ein Docker-Container verwendet wird. Oracle empfiehlt die von GitHub bereitgestellte Einzelinstanz, die aus dem GitHub-Repository abgerufen werden kann[11]. Zunächst klonen wir das Repository, was sowohl über SSH, als auch über HTTPS geschehen kann. In diesem Anwendungsbeispiel wird HTTPS angewandt.

```
$ git clone https://github.com/oracle/docker-images.git
```

Um die Erzeugung der Images zu erleichtern, kann das Skript buildDockerImage.sh genutzt werden, welches sich im Ordner docker-images/OracleDatabase/SingleInstance/ befindet. Das Skript buildDockerImage.sh ist nur ein Shell-Skript, das MD5-Überprüfungen durchführt. Erfahrenen Anwender ist es möglich, ihr docker build direkt mit ihrem bevorzugten Konfiguration aufzurufen. Man muss die Installationsbinaries der Oracle-Datenbank zur Verfügung stellen und sie in den Ordner dockerfiles/<version> legen. Es genügt, die Binärdateien für die Edition, die installiert werden sollen, bereitzustellen. Die Binärdateien können vom Oracle Technology Network heruntergeladen werden.

<https://www.oracle.com/database/technologies/oracle-database-software-downloads.html>
mit der Version 18.4

Es muss unbedingt der Linux-Link verwendet werden: Linux x86-64. Die benötigte Datei sieht wie folgt aus:



```
Patricks-MBP:18.4.0 patrickjarren$ ll
total 8936440
-rw-r--r--  1 patrickjarren  staff      2480 22 Feb 01:52 Dockerfile.xe
-rwxr-xr-x  1 patrickjarren  staff      1137 22 Feb 01:52 checkDBStatus.sh
-rwxr-xr-x  1 patrickjarren  staff        905 22 Feb 01:52 checkSpace.sh
-rw-r--r--@ 1 patrickjarren  staff 4564649047 22 Feb 12:57 linuxx64__database.zip
-rwxr-xr-x  1 patrickjarren  staff        533 22 Feb 01:52 oracle-xe-18c.conf
-rw-r--r--  1 patrickjarren  staff      6132 22 Feb 01:52 runOracle.sh
-rw-r--r--  1 patrickjarren  staff       775 22 Feb 01:52 setPassword.sh
Patricks-MBP:18.4.0 patrickjarren$
```

Abbildung 3.6: Ordner der Installation Version 18.4

Außerdem muss beachtet werden, dass eine Internetverbindung für yum besteht. Die Binärdateien dürfen nicht dekomprimiert werden. Das Skript wird das erledigen und fehlschlagen, wenn es manuell dekomprimiert wird. Wenn alle Dateien am richtigen Ort vorhanden sind, kann das Skript im Verzeichnis dockerfiles ausgeführt werden.

```
$ ./buildDockerImage.sh -h
```

Nach der Ausführung wird folgende Information angezeigt.

```
Usage: buildDockerImage.sh -v [version] [-e | -s | -x] [-i] [-o] [Docker build option]
Builds a Docker Image for Oracle Database.

Parameters:
  -v: version to build
      Choose one of: 11.2.0.2 12.1.0.2 12.2.0.1 18.3.0 18.4.0 19.3.0
  -e: creates image based on 'Enterprise Edition'
  -s: creates image based on 'Standard Edition 2'
  -x: creates image based on 'Express Edition'
  -i: ignores the MD5 checksums
  -o: passes on Docker build option

* select one edition only: -e, -s, or -x

LICENSE UPL 1.0

Copyright (c) 2014-2019 Oracle and/or its affiliates. All rights reserved.
```

Abbildung 3.7: Ausgabe buildDockerImage

Ausgehend von dem Output des Skript sehen die erforderlichen Eingabewerte für die betreffende Version wie folgt aus.

```
$ sh buildDockerImage.sh -v 18.4.0 -x -i
```

Die daraus resultierenden Images werden ein Image mit den installierten Oracle-Binärdateien sein. Beim ersten Start des Containers wird eine neue Datenbank erstellt. Die folgenden Zeilen markieren, wann die Datenbank einsatzbereit ist:

```
#####
DATABASE IS READY TO USE!
#####
```

Um das Oracle Database Docker-Image zu starten, wird nun der docker run Befehl wie folgt verwendet:

```
docker run --name <container name> \  
-p <host port>:1521 -p <host port>:5500 \  
-e ORACLE_PWD=<your database passwords> \  
-e ORACLE_CHARACTERSET=<your character set> \  
-v [<host mount point>:]/opt/oracle/oradata \  
oracle/database:18.4.0 -xe
```

Parameters:

- `--name:` The name of the container (default: auto generated)
- `-p:` The port mapping of the host port to the container port.
Two ports are exposed: 1521 (Oracle Listener), 8080 (APEX)
- `-e ORACLE_PWD:` The Oracle Database SYS, SYSTEM and PDB_ADMIN password (default: auto generated)
- `-e ORACLE_CHARACTERSET:`
The character **set** to use when creating the database (default: AL32UTF8)
- `-v /opt/oracle/oradata`
The data volume to use **for** the database.
Has to be writable by the Unix "oracle" (uid: 54321) user inside the container!
If omitted the database will not be persisted over container recreation.
- `-v /opt/oracle/scripts/startup | /docker-entrypoint-initdb.d/startup`
Optional: A volume with custom scripts to be run after database startup.


```
-v /opt/oracle/scripts/setup | /docker-entrypoint-initdb.d/setup
```

Optional: A volume with custom scripts to be run
after database setup.

In dieser Arbeit wurde der Container wie folgt ausgeführt:

```
$ docker run --name oracle-xe -p 1521:1521 -p 5500:5500 -e  
ORACLE_PWD=ORCLPDB1 oracle/database:18.4.0 -xe
```

Es gibt zwei Ports, die in diesem Image herausgestellt sind:

1521 ist der Port für die Verbindung mit der Oracle-Datenbank.

8080 ist der Port von Oracle Application Express (APEX).

Beim ersten Start des Containers wird ein Zufallspasswort für die Datenbank generiert, wenn es nicht angegeben wird. Die ORACLE_SID für Express Edition ist immer XE und kann nicht geändert werden, daher gibt es keinen ORACLE_SID-Parameter für den XE-Build. Sobald der Container gestartet und die Datenbank erstellt ist, kann man sich wie mit jeder anderen Datenbank verbinden. Die Einrichtung des Beispiels ist nun abgeschlossen. Der Container kann sowohl als Server, als auch direkt auf jedem Rechner in den Computerräumen eingesetzt werden. Es muss lediglich noch die Verbindung zu einer Applikation wie SQLPlus, oder SQLDeveloper eingerichtet werden. Werkzeuge, wie SQLPlus können ebenfalls im Container betrieben werden. Für dieses Szenario besteht jedoch keine dringliche Notwendigkeit.

3.3.3 Cloud Integration

Im Unterschied zu einer Installation der 18xe-Variante als Serverbetrieb ist die Einrichtung einer lauffähigen Instanz der Oracle-Datenbank im Rahmen der Oracle Cloud Infrastructure beinahe mühelos, vorausgesetzt, die erforderlichen Verfahrensschritte sind festgelegt. Oracle Cloud Free Tier ermöglicht es, sich für ein Oracle Cloud-Konto anzumelden, das eine Reihe stets kostenloser Dienste und eine kostenlose Testversion mit 300 US-Dollar kostenlosem Guthaben zur Nutzung aller in Frage kommenden Oracle Cloud-Infrastrukturdienste für bis zu 30 Tage bietet. Die Always Free Services stehen für einen unbegrenzten Zeitraum zur Verfügung. Die Dienste der kostenlosen Testversion können so lange genutzt werden, bis 300 US-Dollar an kostenlosem Guthaben verbraucht sind, oder die 30 Tage abgelaufen sind, je nachdem, was zuerst eintritt.[9]

Zunächst bedarf es eines Oracle Cloud Account, welcher angelegt werden kann, unter der folgenden URL:

<https://www.oracle.com/index.html>

3. Entwicklungszyklus der Bachelor-Thesis

Für das Geschäftsmodell sind viele Informationen zwingend erforderlich. Diese umfassen die nachfolgend aufgeführten Informationen:

1. Account Type. Hier wird unterschieden zwischen der Privatnutzung und der Geschäftsnutzung.
2. Cloud Account Name. Hier sollte ein erkennbarer Name gewählt werden, zur Verwendung in Konto-URLs.
`https://myservices-cloud_accounts_name.console.oraclecloud.com`
3. Mobile Number. Für Rückfragen zum entsprechenden Konto ist es notwendig eine Mobilnummer anzugeben. Außerdem wird ein Bestätigungscode in einer Textnachricht an die Handynummer gesendet.
4. Credit card. Die Zahlungsverarbeitung, die von Oracle zur Verfügung gestellt wird, geschieht mittels einer Kreditkarte.

Nach Abschluss des Registrierungsprozesses ist es erstmals möglich, auf das Dashboard der Cloud-Oberfläche zugreifen zu können.

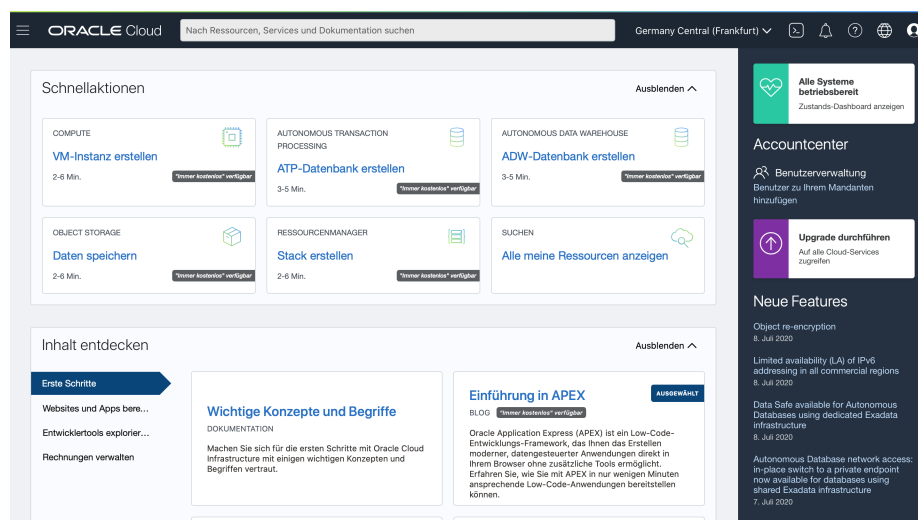


Abbildung 3.8: Dashboard ORACLE Cloud

Damit kann jetzt eine Datenbankumgebung in Form einer ATP-Datenbank eingerichtet werden. Mittels einfachem Klick auf die entsprechende Kachel innerhalb der Schnellzugriffe gelangt man zu der erstellen Maske.

Zusätzlich zu dem Anzeigenamen wird auch der Datenbank ein Name gegeben, welcher für den Aufruf der Datenbank notwendig ist. Die Entscheidung ist außerdem

Autonome Datenbank erstellen

Geben Sie Basisinformationen für die autonome Datenbank an

Compartment
patrickjarren (Root)

Anzeigenname
haw_cloud_1
Ein benutzerfreundlicher Name, anhand dessen Sie die Ressource einfach identifizieren können.

Datenbankname
hawcloud1
Der Name darf nur Buchstaben und Zahlen enthalten und muss mit einem Buchstaben beginnen. Maximal 14 Zeichen.

Workload-Typ auswählen

Data Warehouse
Konfiguriert die Datenbank für eine Workload in einem Decision-Support-System oder in einem Data Warehouse mit Schwerpunkt auf umfangreichen Datenscanvorgängen. ✓

Transaktionsverarbeitung
Konfiguriert die Datenbank für eine Transaktions-Workload mit Schwerpunkt auf hohen Volumen an zufälligen Datenzugriffen.

[Autonome Datenbank erstellen](#) [Abbrechen](#)

Abbildung 3.9: ATP-Datenbank Schritt 1

Immer kostenlos ⓘ
☒ Nur Konfigurationsoptionen vom Typ "Immer kostenlos"

⚠ Ihre autonome Datenbank vom Typ "Immer kostenlos" wird automatisch gestoppt, wenn 7 aufeinander folgende Tage lang keine Aktivität verzeichnet wird. Ihre Daten werden beibehalten, und Sie können die Datenbank neu starten, um sie weiter zu verwenden. Wenn die Datenbank 3 Monate lang nicht gestartet wird, können Sie nicht mehr darauf zugreifen. [Weitere Informationen](#)

Datenbankversion auswählen
19c

OCPU-Anzahl
1
Autonome Datenbanken vom Typ "Immer kostenlos" können bis zu 1 Core nutzen. Die Anzahl der CPU-Cores kann nicht angepasst werden.

Speicher (TB)
0,02
Autonome Datenbanken vom Typ "Immer kostenlos" können bis zu 0,02 TB (20 GB) Speicher nutzen. Die Speichergröße kann nicht angepasst werden.

☐ Autoscaling
Dadurch kann das System die bis zu dreifache Anzahl bereitgestellter Cores verwenden, wenn die Workload zunimmt. [Weitere Informationen](#).

Abbildung 3.10: ATP-Datenbank Schritt 2

in Abhängigkeit vom Anwendungsfall zu fällen, welcher Workload-Typ bereitgestellt wird. Dieses Praxisbeispiel nutzt den Typ eines Data Warehouse, es handelt sich hierbei nicht um eine allgemeine Empfehlung für die Hochschule.

Als nächster Schritt kann der Anwender sich entscheiden, entweder es handelt sich

um eine stets kostenlose Datenbank oder es entstehen eventuelle Kosten. Wird die freie Variante ausgewählt, stehen die Optionen zur Hardware-Skalierung nicht zur Verfügung und es werden Standardwerte festgelegt. Es ist auch zu beachten, dass die Datenbank ohne Datenverlust automatisch gestoppt wird, falls keine Aktivität innerhalb von sieben Tagen verzeichnet wird. Ein Neustart ist jederzeit möglich. Falls verschiedene Datenbankversionen zur Verfügung stehen, kann außerdem innerhalb der Zahlungsbedingungen eine Version ausgewählt werden. Entsprechend der Auswahl, die zum Verfassungszeitpunkt dieses Dokuments zur Verfügung gestanden hat, ist lediglich die 19C-Version abrufbar. Mit entsprechender Begrenzung der Rechnerleistung und des Festplattenspeichers.

Benutzername SCHREIBGESCHÜTZT

ADMIN

Benutzername des Administrators kann nicht bearbeitet werden.

Kennwort

Kennwort bestätigen

Netzwerkzugriff auswählen

Zugriffstyp

Sicheren Zugriff von überall aus zulassen

Sie können den Zugriff auf bestimmte IP-Adressen und VCNs einschränken. ✓

Virtuelles Cloud-Netzwerk

Nur privater Zugriff über ein VCN.

☐ Zugriffskontrollregeln konfigurieren ⓘ

Abbildung 3.11: ATP-Datenbank Schritt 3

Für den Zugriff auf die erstellte Datenbank wird zusätzlich ein Administrator-Passwort zugewiesen. Dieses kann später jederzeit geändert werden, ist jedoch für die erste Inbetriebnahme erforderlich. Zusätzlich kann die gewünschte Zugriffsart festgelegt und der komplette Zugriff auf VCN beschränkt werden.

Lizenztyp auswählen

Bring Your Own License (BYOL)

Oracle Database-Softwarelizenzen meiner Organisation im Datenbankservice verwenden. [Weitere Informationen.](#)

Lizenz inklusive

Neue Oracle Database-Softwarelizenzen und Datenbankservice abonnieren. ✓

[Erweiterte Optionen anzeigen](#)

[Autonome Datenbank erstellen](#) [Abbrechen](#)

Abbildung 3.12: ATP-Datenbank Schritt 4

Danach wird entschieden, ob die Lizenz innerhalb der Instanz eingeschlossen und erstellt wird, oder ob es eigene Lizenzen gibt, die verwendet werden können, was auch unter der Abkürzung BYOL firmiert. Schlussendlich erfolgt die Bestätigung und die Datenbank wird eingerichtet. Eine Liste der erzeugten Datenbankinstanzen wird im folgenden Bereich aufgeführt. Mit einem Klick auf den Datenbanknamen wird eine Weiterleitung zum Dashboard der Datenbank vorgenommen.

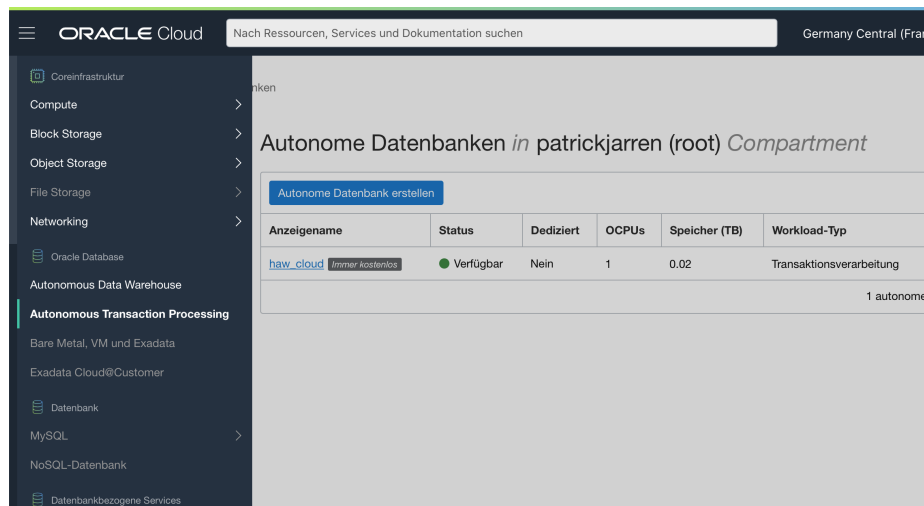


Abbildung 3.13: ATP-Datenbank Schritt 5

3.4 Testszenarien

Nun, da die Datenbanken betriebsbereit vorliegen, besteht die Möglichkeit, einige Testszenarios zu durchlaufen. Dabei wird die Anbindung an eine Entwicklungsumgebung getestet, es werden Datenbankbenutzer angelegt und mögliche Data-Mining-Anbindungen aufgeführt.

3.4.1 Verbindung zum SQL Developer

Oracle SQL Developer ist die kostenlose, integrierte Entwicklungsumgebung, die die Entwicklung und Verwaltung der Oracle-Datenbank, sowohl in traditionellen, als auch in Cloud-Bereitstellungen, vereinfacht. SQL Developer bietet eine vollständige End-to-End-Entwicklung von PL/SQL-Anwendungen, ein Arbeitsblatt zum Ausführen von Abfragen und Skripten, eine DBA-Konsole zum Verwalten der Datenbank, eine Berichtsschnittstelle, eine vollständige Datenmodellierungslösung und eine Migrationsplattform zum Verschieben von Datenbanken von Drittanbietern nach Oracle[10]. Die Installation ist auf jedem Betriebssystem identisch. Auf der Produktseite kann eine Installationsdatei bezogen werden. Da der SQL Developer auf Java basiert, muss die Java-Laufzeitumgebung auf dem System installiert sein.

Fehlerbehebung MacOS

Bei der Installation von SQL Developer auf Mac können sich Fehler aufgrund einer fehlerhaften Version der Java-Laufzeitumgebung einschleichen. Sollte dies auftreten, kann man sich wie folgt helfen.

1. Löschen der folgenden Java-Dateien und Ordner auf dem Mac.

```
/Library/Internet Plug-Ins/JavaAppletPlugin.plugin  
/Library/Java/JavaVirtualMachines/
```

2. Herunterladen und installieren von Java 8.

<https://www.oracle.com/java/technologies/javase-jre8-downloads.html>

Anbindung Docker

Für den exemplarischen Beispielfall mit der lokalen Docker-Installation kann sich wie im Bild nachstehend angegeben angemeldet werden. Für den exemplarischen Anwendungsfall mit einer lokalen Docker-Installation erfolgt die Anmeldung wie in der folgenden Abbildung dargestellt. Das Passwort entspricht dem von Benutzer im vorherigen Schritt gewählten Kennwort, um den Container zu starten.

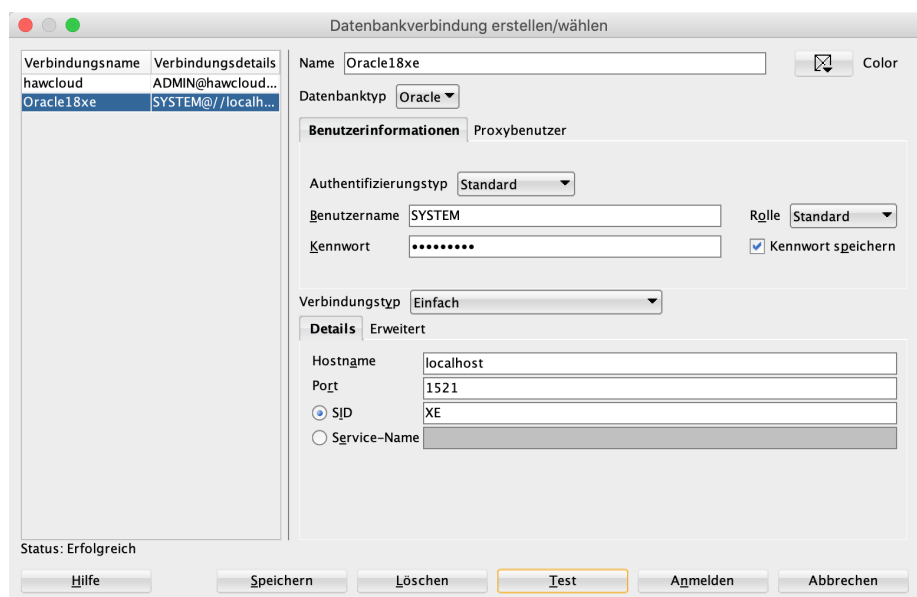


Abbildung 3.14: Docker-Anmeldung bei SQL Developer

Anbindung Cloud

Im Gegensatz zu der eher als gängige Verbindungsvariante bekannte Methode, die bisher verwendet wurde, wird für die Anbindung an die Cloud ein sogenanntes Wallet verlangt, das alle Client-Zugangsdaten enthält. Dies kann vom Dashboard der Datenbankinstanz in der Cloud heruntergeladen und später in SQL Developer nach Anpassung der Verbindungsart hinzugefügt werden.

Datenbankverbindung

HilfeSchließen

Sie benötigen die Clientzugangsdaten und die Verbindungsinformationen für die Verbindung mit der Datenbank. Die Clientzugangsdaten umfassen das Wallet, das für alle Verbindungstypen erforderlich ist.

Clientzugangsdaten (Wallet) herunterladen

Wählen Sie zum Herunterladen Ihrer Clientzugangsdaten den Wallet-Typ aus, und klicken Sie auf **Wallet herunterladen**. Daraufhin werden Sie zum Erstellen eines Kennworts für das Wallet aufgefordert.

Wallet-Typ ⓘ

Instanz-Wallet

Wallet herunterladenWallet rotieren

Letzte Rotation des Wallets: -

Schließen

Abbildung 3.15: Datenbankverbindung mit Instanz-Wallet

Die anschließende Verbindungsherstellung im SQL Developer gestaltet sich wie folgt:

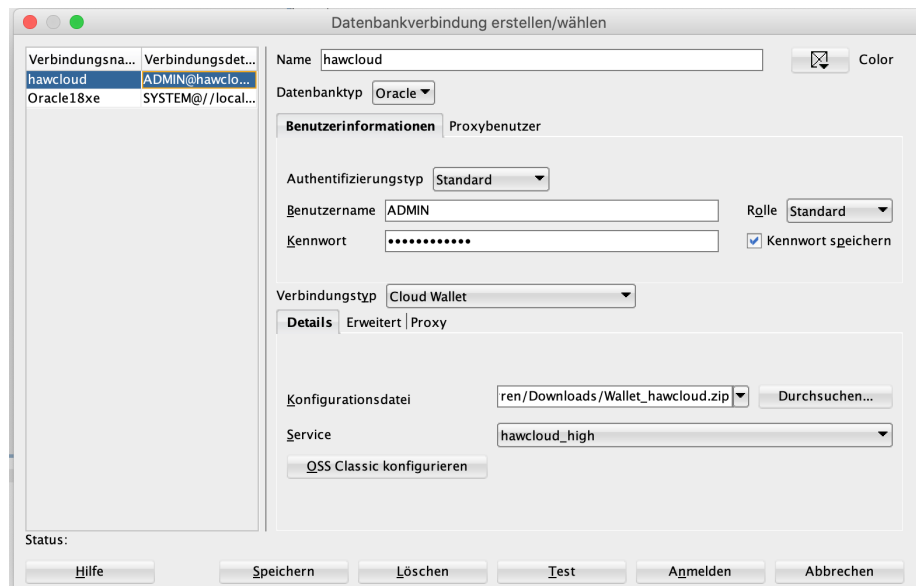


Abbildung 3.16: Datenbankverbindung mit Instanz-Wallet 2

Das hinterlegte Wallet darf dabei nicht endkomprimiert werden, sondern muss im gleichen Zustand wie heruntergeladen auch hinzugefügt werden. Das Passwort entspricht den bei der Erstellung der Instanz hinzugefügten Admin-Benutzern.

Alternativ zu dieser Option bietet Oracle im Dashboard der Cloud den SQL Developer Web. Dieser lässt sich von überall direkt starten, außerdem können SQL-Anweisungen und Skripte in einem Arbeitsblatt ausgeführt, Daten exportiert, Überwachungsschnittstellen bereitgestellt und Data Modeler-Diagrammen entworfen werden. Der ADMIN-Benutzer kann anderen Datenbankbenutzern den Zugriff auf SQL Developer Web ermöglichen. Datenbankbenutzer, die keine Service-Administratoren sind, haben keinen Zugriff auf die Servicekonsole Autonomous Database. Der ADMIN-Benutzer ermöglicht den Zugriff auf SQL Developer Web, indem er den Zugriff für einen Benutzer aktiviert und eine URL für den Zugriff auf SQL Developer Web bereitstellt. Die Anweisung, den Schemazugriff auf SQL Developer Web zu aktivieren:

BEGIN

```
ORDS_ADMIN.ENABLE_SCHEMA(  
    p_enabled => TRUE,  
    p_schema => 'schema-name',  
    p_url_mapping_type => 'BASE_PATH',  
    p_url_mapping_pattern => 'schema-alias',  
    p_auto_rest_auth => TRUE  
);  
COMMIT;
```

END;

/

1. *schema-name*. Ist der Name des Datenbankschemas in Großbuchstaben.
2. *schema-alias*. Ist ein Alias für den Schemanamen, der in der URL für den Zugriff auf SQL Developer Web verwendet werden soll.
3. *p_auto_rest_auth*. Spezifiziert den REST /metadata-catalog/ Endpunkt, für den eine Autorisierung erforderlich ist. REST verwendet den Metadaten-Katalog, um eine Liste der im Schema veröffentlichten Dienste zu erhalten. Dieser Parameter ist auf TRUE zu setzen.

Nachdem der Benutzerzugriff für das angegebene Schema aktiviert wurde, erstellt der Admin einem Benutzer die URL für den Zugriff auf SQL Developer Web.

3.4.2 Datenbankbenutzer anlegen

Nicht nur bei Oracle besteht die Anforderung, Datenbankbenutzer einzurichten, beispielsweise für maschinelles Lernen oder auch einfach für Studierende im Rahmen einer Praxisvorlesung, die sich mit den Anforderungen des Dozenten befassen.

Anbindung Docker

Nach der erfolgreichen Verbindung innerhalb der Entwicklungsumgebung zu der Datenbank gibt es unter Berücksichtigung folgenden Queries die Möglichkeit, Benutzer anhand eigener Regeln anzulegen.

```
alter session set "_ORACLE_SCRIPT"=true;
```

[2]

Der Benutzer kann anschließend über die visuelle Schnittstelle, oder mit Hilfe von Abfragen hinzugefügt werden, was im Folgenden beispielhaft dargestellt wird.

— USER SQL

```
CREATE USER "patrick" IDENTIFIED BY "patrick1"  
DEFAULT TABLESPACE "USERS"  
TEMPORARY TABLESPACE "TEMP";
```

— QUOTAS

```
ALTER USER "patrick" QUOTA UNLIMITED ON "USERS";
```

— ROLES

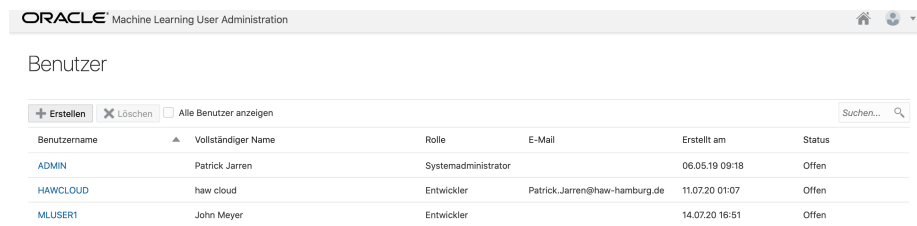
```
GRANT "CONNECT" TO "patrick" ;  
GRANT "RESOURCE" TO "patrick" ;  
ALTER USER "patrick" DEFAULT ROLE "CONNECT", "RESOURCE";
```

— SYSTEM PRIVILEGES

```
GRANT CREATE VIEW TO "patrick" ;  
GRANT CREATE TABLE TO "patrick" ;  
GRANT CREATE SYNONYM TO "patrick" ;
```

Anbindung Cloud

Neben der konventionellen Einrichtung der Datenbanknutzer bietet die Cloud die Möglichkeit, dies in einer Webanwendung zu tun, die im Dashboard der Datenbank zu finden ist. Unter der Oracle ML-Benutzerverwaltung werden die Benutzer für das maschinelle Lernen definiert, sie sind jedoch ebenfalls Datenbankbenutzer. Es obliegt dem Datenbankadministrator, wie er die Nutzer erstellen will und kann.



The screenshot shows the Oracle Machine Learning User Administration web interface. At the top, there's a header with the Oracle logo and the title 'Machine Learning User Administration'. Below the header, the word 'Benutzer' (Users) is displayed. There are buttons for '+ Erstellen' (Create), 'X Löschen' (Delete), and a checkbox 'Alle Benutzer anzeigen' (Show all users). A search bar with the placeholder 'Suchen...' is also present. The main content is a table with the following columns: 'Benutzername' (Username), 'Vollständiger Name' (Full Name), 'Rolle' (Role), 'E-Mail', 'Erstellt am' (Created at), and 'Status'. The table contains three rows of user data.

Benutzername	Vollständiger Name	Rolle	E-Mail	Erstellt am	Status
ADMIN	Patrick Jaren	Systemadministrator		06.05.19 09:18	Offen
HAWCLOUD	haw cloud	Entwickler	Patrick.Jaren@haw-hamburg.de	11.07.20 01:07	Offen
MLUSER1	John Meyer	Entwickler		14.07.20 16:51	Offen

Abbildung 3.17: Benutzerverwaltung in der Cloud

3.5 Data-Mining-Lösung in der Praxis

Drei Methoden zur Advanced Analytics in Datenbanken, die für diese Arbeit ausgewählt wurden, werden im Folgenden dargestellt, sowie deren mögliche Integration.

3.5.1 SQL Developer

Die Erste ist der SQL Developer. Dieser wird denjenigen vertraut sein, die mit der Oracle-Datenbank bereits entwickelt haben. Es ist im Grunde ein lokales Tool, mit dem mit der Datenbank gearbeitet werden kann. Und es gibt ein Plug-In, mit dem maschinenlernende Abhängigkeiten entwickelt werden können. Zum Beispiel, um Modelle zu erstellen und Bewertungen und Vorhersagen zu machen, und dergleichen mehr.

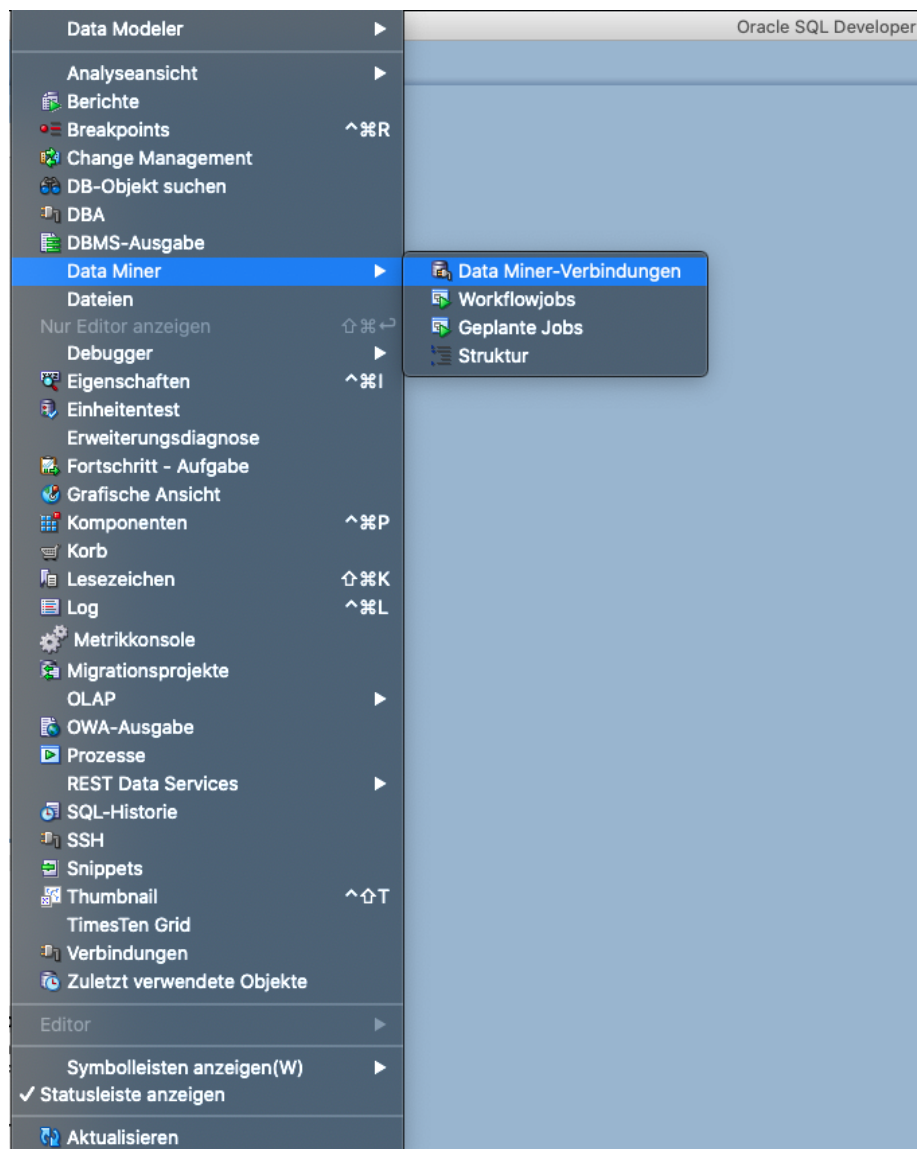


Abbildung 3.18: Data Miner-Verbindung im SQL Developer

Die Voraussetzungen für die Nutzung des Data-Mining Werkzeug lassen sich wie folgt zusammenfassen. Eine Verbindung zu einem Konto in einer Oracle Datenbank 11g Release 2 (11.2.0.4) und höher. Das Konto, mit dem eine Verbindung hergestellt wird, muss über alle Berechtigungen verfügen. Definiert sein muss dazu mindestens eine Verbindung, um das Konto anzugeben, in dem Workflow-Knoten vorhanden sind. Die Schaffung der notwendigen Voraussetzungen wurde zuletzt in den Abschnitten beschrieben und erfordert lediglich Genehmigungen entsprechend den Personalerfordernissen.

3.5.2 R zu Oracle

Für eine detaillierte Datenanalyse unter Verwendung einer Programmiersprache für statistische Berechnungen und Grafiken gibt es die Möglichkeit des ROracle-Pakets, das hier kurz vorgestellt wird. Inwieweit der Mehrwert mit der Programmiersprache R und nicht mit Python oder vergleichbaren Mitteln erreicht werden soll, wird an dieser Stelle nicht untersucht. Jedoch fördert Oracle selbst den Einsatz dieser Sprache. SQL und R sind zwei der leistungsstärksten Tools für die Datenanalyse. ROracle ist ein Open Source R-Paket, das bei CRAN verfügbar ist und einem Netzwerk von FPT- und Webservern, die aktuelle Versionen von Code und Dokumentation für R beinhalten. Das ROracle-Paket ist eine Implementierung des R DBI-Paketes, das Oracle Open Catalog Interface für hohe Leistung und Skalierbarkeit verwendet. Es erfordert die Installation von Oracle Instant Client oder Oracle Database Client auf dem Client-Rechner.

```
library(ROracle)

drv <- dbDriver("Oracle")
con <- dbConnect(drv, "rquser", "rqpasswd")
dbWriteTable(con, "TEST_TABLE", test_table)
dbGetQuery(con, "select count(*) from TEST_TABLE")
d <- dbReadTable(con, "TEST_TABLE") dbDisconnect(con)
```

[7]

3.5.3 Oracle Machine Learning

Hinter der Servicekonsole innerhalb des Datenbank-Dashboards in der Oracle Cloud, unterhalb des Entwicklungsbereichs, befindet sich der Bereich Oracle Machine Learning Notebooks. Oracle Machine Learning wird von einem Administrator auf System- und Anwendungsebene verwaltet. Der Administrator erstellt, bearbeitet und löscht Oracle Machine Learning-Benutzerkonten. Der Administrator ist nicht berechtigt, Notebooks auszuführen. Der Administrator kann Notizbücher nur lesen[12].

Mit einem nunmehr eingerichteten Benutzer kann ein Notebook für das Training und die Vorhersage unserer Modelle angelegt werden.

3. Entwicklungszyklus der Bachelor-Thesis

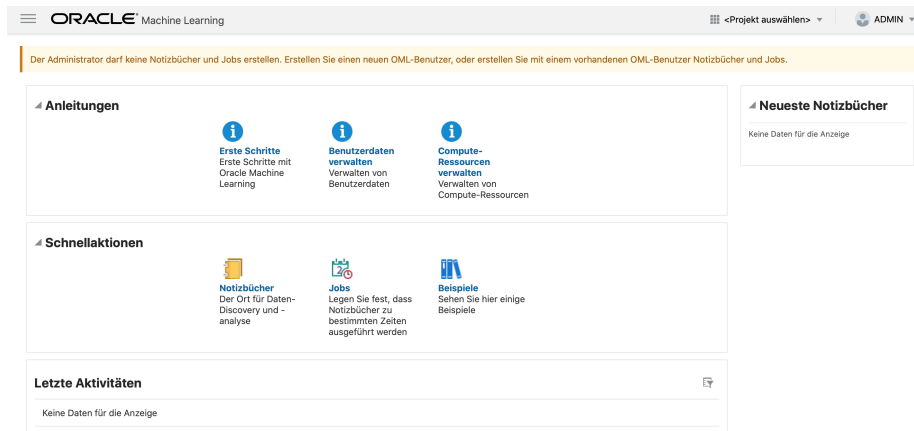


Abbildung 3.19: Admin Homepage

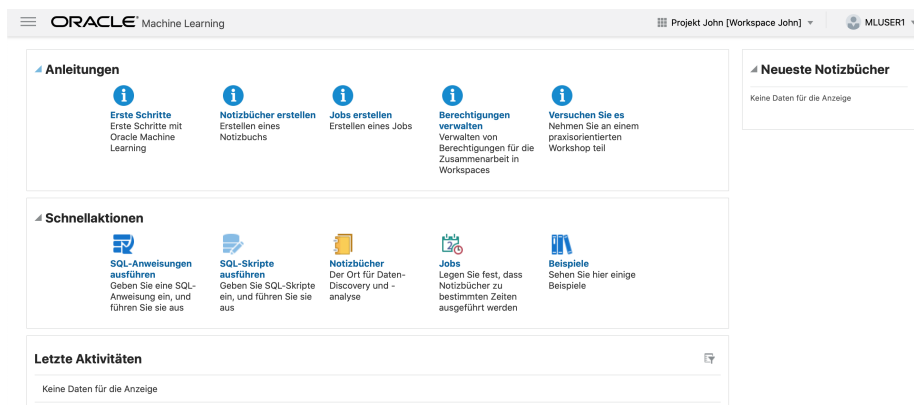


Abbildung 3.20: Entwickler Homepage

4 Evaluation

In diesen Kapiteln sind alle Informationen für die Schlussfolgerung aufbereitet, die in Kapitel 5 anschließt. Zunächst ist es erforderlich, die Besonderheiten der beiden eingesetzten Produktvarianten zu berücksichtigen. Auf Grund der Kostenfreiheit beider Produkte sind auch einige Einschränkungen zu beachten. Sodann werden die beiden Varianten auf Basis der kritischen und weniger kritischen Anforderungen gegenübergestellt. Eine Kosten-/Nutzen-Analyse ist nicht notwendig, aufgrund der Monetarisierung beider Modelle.

4.1 Express Edition(XE) 18.4.0

Der Verzicht auf den vierteljährlichen Patch-Zyklus und den Kundendienst ist ein umständliches Unterfangen, da es von Oracle keinerlei Unterstützung für mögliche Sicherheitsschwachstellen gibt. Fragen zu 18c XE müssen eigenständig gelöst werden, oder das von Oracle geschaffene Diskussionsforum muss in Anspruch genommen werden. Dementsprechend sollte die Express Edition eine Datenbankalternative für unkritische Anwendungsfälle sein. Ferner verfügt diese Version nicht über zahllose Ressourcen. Ein Blick auf die bereits belegten Ressourcen ist daher ausdrücklich zu empfehlen. Aufgrund der nicht existenten Lizenzgebühren ist der Betrieb auf ansonsten lizenz-kritischen virtuellen Infrastrukturen empfehlenswert.

Einschränkungen

- Maximal 12 GB Nutzdaten, unabhängig vom Komprimierungsgrad
- Maximal 2 GB SGA+PGA
- Maximal 2 CPUs für Vordergrundprozesse
- Maximal 1 XE Datenbankinstanz pro Server
- Maximal 3 Pluggable Databases

[6]

In vorangegangenen Versionen gab es eine Beschränkung von nur 1 XE pro Computer (nicht pro VM). Im Lizenzleitfaden wird diese Beschränkung nicht mehr erwähnt. Dies bedeutet schlussfolgernd, dass mehrere XEs auf derselben Maschine haben können, aber in verschiedenen VMs sein müssen. Der andere einschränkende Faktor ist, dass das Orakel SID auf XE fixiert ist: Dies könnte für Upgrade-Szenarien problematisch sein. Es können nicht zwei aktive Datenbanken - einen alten 18XE und den neuen - in derselben Umgebung sein.

4.1.1 Vorteile der benannten Lösung

Private Verwendung wird erleichtert. Neben einer Installation wie in dieser Arbeit, kann auch die Veröffentlichung eines Images innerhalb einer Docker Registry vorgenommen werden. Jeder Nutzer kann somit mit genau den selben Datenstand experimentieren und weiterarbeiten. Die Durchführung der persönlichen Projekte ist einfacher, ohne Zugriff auf die Cloud zu benötigen.

4.1.2 Nachteile der benannten Lösung

Skalierung. Der Anwender ist verantwortlich für den Umfang des beanspruchten Festplattenspeicherplatzes, sowie für die zur Berechnung der Anfragen erforderliche Hardware. Die 12 GB Benutzerdaten bedeuten, dass nicht viele Dokumentdateien in der Datenbank gespeichert werden können. Für die meisten gut strukturierten Daten sollte dies ausreichend sein, solange keine Video-, Audio- oder sehr große Fotosammlungen verwendet werden. Mit Hyperthreading könnte eine CPU problemlos bis zu 4 oder sogar 8 Threads unterstützen. Dies ist mittlerweile auf zwei Threads beschränkt. Oracle versucht damit sicherzustellen, dass die Einschränkungen nicht durch den Einsatz fortschrittlicher Hardware umgangen werden. Automatische Skalierung ist nicht erforderlich, wenn der Server oder das Endgerät über genügend Ressourcen verfügt. Darüber hinaus werden auch die Servicekosten für eine automatische Skalierung eliminiert, und können mit den vorhandenen Ressourcen budgetiert werden.

Docker als Fehlerquelle. Docker hat seinen Platz in der Informationstechnologie bereits gefunden. Die bloße Anwendung durch Starten eines Containers ist für die

meisten Anwender grundsätzlich kein Problem. Docker ist jedoch ebenfalls eine mögliche Störungsquelle. Werden mehrere Container verwendet, können sich die Ports überlappen oder der Container muss bei Störungen neu gestartet werden. Auch wenn diese Art von Fehlern zunächst unbedeutend erscheint, so erschweren sie die Bedienung zusätzlich. Docker wird lediglich als Hilfsmittel für die Realisierung verwendet. Obwohl auch eine Realisierung ohne Docker denkbar wäre, wird aufgrund der Docker-eigenen Architektur empfohlen, diese zu nutzen.

Erhöhter Verwaltungsaufwand. Der Verwaltungsaufwand sollte theoretisch nicht höher ausfallen, als in jeder Cloud Lösung. Dennoch ist dies der Fall, da die Entwicklung für die administrative Pflege innerhalb der Cloud Web Oberfläche ein vielfaches einfacher handhabbar ist, als in der klassischen Lösung. Ein gutes Beispiel hierfür ist die Pflege der Benutzerkonten für das Machine Learning.

4.2 Oracle Cloud Infrastructure

Auf der Oracle OpenWorld 2019 hat Oracle die “Cloud Free Tier” mit “Always Free”-Diensten eingeführt, so dass jeder die erste autonome Datenbank der Welt und die Oracle Cloud-Infrastruktur für unbegrenzte Zeit ausprobieren kann. Es bietet Kunden die Möglichkeit, die volle Funktionalität der autonomen Oracle-Datenbank und der Oracle-Cloud-Infrastruktur, einschließlich Compute VMs, Block- und Objektspeicherung und Load Balancer, aufzubauen, zu erlernen und zu erforschen. Anstatt wie bei der XE Version deren Einschränkung zu benennen, ist es von Vorteil, die gebotenen Inhalte aufzuführen. Zu den Diensten die immer kostenlos sind gehören:

- Die Wahl zwischen autonomer Transaktionsverarbeitung oder autonomem Data Warehouse. Insgesamt 2 Datenbanken, jede mit 1 OCPU und 20 GB Speicherplatz.
- 2 virtuelle Maschinen mit je 1/8 OCPU und 1 GB Speicher.
- 2 Block-Volumes, insgesamt 100 GB. 10 GB Objektspeicher. 10 GB Archivspeicher.
- Load Balancer:

- 1 Instanz
- 10 Mbps Bandbreite
- 1 Milliarde Abruf-Datenpunkte
- Benachrichtigungen: 1 Million über https versendete Daten pro Monat
- 1.000 über E-Mail versendete Daten pro Monat
- Ausgehender Datentransfer: 10 TB pro Monat

Wie in der Express Edition Version, steht der Community-Support über die Foren zur Verfügung. Kunden, die ausschließlich Always Free Ressourcen verwenden, haben keinen Anspruch auf Oracle Support. Anwendungen, die mit Always Free Lösungen betrieben werden, sollten ebenfalls für unkritische Anwendungsfälle eingesetzt werden.

4.2.1 Vorteile der benannten Lösung

Intuitive Bedienung der Oberfläche. Die moderne Webschnittstelle ist sehr intuitiv und erleichtert es erfahrenen Benutzern, wie auch Einsteigern, die Tätigkeit auszuführen. Weiterhin ist jederzeit eine Übersicht über das Ressourcenmanagement gewährleistet. Darüber hinaus kann eine neue Instanz einer Datenbank oder einer virtuellen Maschine in unmittelbarer gebootet werden.

Hochverfügbarkeit. Innerhalb einer geographisch eingeschränkten Region sind ein oder mehrere Datenzentren isoliert voneinander und garantieren durch eine Redundanz rund um die Uhr eine hohe Verfügbarkeit. Realistischerweise kann davon ausgegangen werden, dass kein selbst betriebenes Rechenzentrum eine höhere Verfügbarkeit garantieren kann.

Unentgeltlichkeit/Pay-per-use. Ist man nicht zahlungswillig, erhält man in der Regel auch keine Dienstleistung. Eine vollkommen kostenfreie Ausführung der nahezu nichts fehlt, ist eine zeitgemäße, gute Serviceleistung. Dies beinhaltet die Möglichkeit, die eigenen Ressourcen nach Verbrauch ohne hohen Neukostenaufwand zu dimensionieren.

Umfangreiche Dokumentationen. Die Dokumentation ist Bestandteil der Arbeitsumwelt des Entwicklers wie der Computer selbst. Obwohl Drittquellen selten sind, können die meisten dieser Informationen gut strukturiert in der Wissensdatenbank bei Oracle abgerufen werden. Sämtliche der angebotenen Produkte werden auch dokumentiert.

4.2.2 Nachteile der benannten Lösung

Angebot jederzeit variabel. Die angebotenen Produkte können abhängig vom Zeitraum, in dem sie eingesetzt werden sollen, unterschiedlich sein. Dies stellt in der Regel nicht unbedingt einen Nachteil dar, da die neuen Versionen unmittelbar in der Cloud verfügbar sind. Allerdings kann dies dazu beitragen, dass bisherige Funktionalitäten in zukünftigen Versionen möglicherweise nicht mehr unterstützt werden.

Gespeicherte Benutzerdaten. Für die Erstellung von Benutzerkonten sind Informationen erforderlich, die den Vornamen, Nachnamen und die E-Mail-Adresse voraussetzen. Diese Angaben werden abgespeichert, damit der Anwender identifizierbar ist, der für etwaige Veränderungen verantwortlich war. Dies könnte zu Datenschutzproblemen führen.

4.3 Gegenüberstellung

Im nachfolgenden Vergleich werden sowohl die positiv- als auch die Negativfaktoren beider Lösungsvarianten gegenübergestellt. Die Auswahl einzelner Punkte erfolgt auf Basis der im vorhergehenden Kapitel dargelegten Kriterien. Dabei werden zunächst alle Vorteile aufgezeigt.

Vorteile	
Express Edition(XE) 18.4.0	Oracle Cloud Infrastructure
Private Verwendung wird erleichtert	Intuitive Bedienung der Oberfläche Hochverfügbarkeit Pay-per-use/Unentgeltlichkeit Umfangreiche Dokumentationen

Tabelle 4.1: Vorteile der Alternativen

Als Vorteil der Express Edition gegenüber der Cloud Variante findet sich der Punkt der privaten Verwendung wieder. Im Gegensatz dazu hat die Cloud-Variante jedoch vier Vorteile. Was die Vorteile als auch die folgenden Nachteile betrifft, so ist es dem Leser freigestellt, den Grad der Beurteilung festzulegen, da die genannten Punkt stets abhängig vom vorliegenden Szenario sind. Bei einer empirischen Bewertung jedoch, geht der Punkt an die Cloud-Variante, da diese deutlich mehr Vorteile bietet. Wie bei den Nachteilen kann der sachverständige Leser anfänglich negieren, dass diese Vorzüge nur in einer der Varianten vorkommen. Jedoch sollte hinzugefügt werden, dass sich der Umfang oder die Qualität zu sehr unterscheiden, um den Punkt nicht nur einer der Varianten zuzuordnen.

Nachteile	
Express Edition(XE) 18.4.0	Oracle Cloud Infrastructure
Skalierung	Angebot jederzeit variabel
Docker als Fehlerquelle	Gespeicherte Benutzerdaten
Erhöhter Verwaltungsaufwand	

Tabelle 4.2: Nachteile der Alternativen

Unter Berücksichtigung der Nachteile zieht die erste Variante ebenfalls den Kürzeren, bedingt durch das Skalierbarkeitsproblem, Docker als mögliche Fehlerquelle und dem erhöhten Verwaltungsaufwand. Auf der anderen Seite verursacht die Variante zwei Nachteile weniger.

4.4 Bewertung auf der Grundlage der festgelegten Anforderungen

Zunächst wird ein Blick auf die in Kapitel 3.1 anfänglich aufgelisteten kritischen Anforderungen geworfen. Entsprechend der Sequenz, die dort ebenfalls angewendet wurde.

Kritisch

	18c Express Edition	Cloud Infrastructure
Zeitgemäße Variante	✓	✓
kostenfrei	✓	✓
Benutzerverwaltung	✓	✓
Data-Mining-Verfahren	✓	✓
Ausfallsicherheit		✓

Tabelle 4.3: Der Vergleich von kritischen Anforderungen

In den Belangen Aktualität, Kostenfreiheit, Benutzerverwaltung sowie Data-Mining-Verfahren sind beide Produkte relativ gleichgestellt. Die Cloud macht die Arbeit leichter, doch die Funktionalität wird von beiden zur Verfügung gestellt. Weit problematischer ist der Aspekt der Zuverlässigkeit. Eine garantierte Ausfallsicherheit stellt keine der genannten Lösungen zur Verfügung, sodass Wahrscheinlichkeiten in Betracht gezogen werden müssen. Dieser Punkt geht eindeutig an Oracle Infrastructure. Die Wahrscheinlichkeit eines Ausfalles ist sehr sehr gering einzuschätzen demnach ist nur die Verbindung zum Rechenzentrum Frankfurt ein möglicher Flaschenhals.

Weniger kritisch

	18c Express Edition	Cloud Infrastructure
Administrative Aufwand	✓	✓
Ausreichend Speicherkapazität		
Performance		
Sicherheitsmaßnahmen		✓

Tabelle 4.4: Der Vergleich von weniger kritischen Anforderungen

Bei der Untersuchung der eher weniger kritischen Aspekte fällt auf, dass beide Optionen mangelnden Service bieten, was vermutlich aus Kostengründen der Fall ist. Der Aufwand für die Verwaltung ist bei keiner der Lösungen wesentlich schlechter. Die Leistungsfähigkeit und der Speicherplatz, den beide Lösungen bieten, ist ebenfalls zu bemängeln. Dies liegt insbesondere an Oracle, die mit diesen Angeboten Kunden dazu verleiten will, auf ihre kostenpflichtigen Varianten umzusteigen. Für die Sicherheit ist der Nutzer bei der XE-Variante selbst mitverantwortlich, während die Cloud-Variante Zugriff auf ein eigenes Sicherheitsprogramm hat.

4.4.1 Offene Fragen

Wie viele Verbindungen kann Oracle Express Edition (XE) verarbeiten?

Oracle selbst liefert keine konkrete Antwort im Rahmen ihrer Unterlagen. Standardmäßig hat Oracle XE den Parameter PROCESSES=300. Dieser Parameter gibt die maximale Anzahl von OS-Benutzerprozessen an, die sich mit der Oracle-Instanz verbinden.

```
SQL> show parameter processes
```

NAME	TYPE	VALUE
aq_tm_processes	integer	1
db_writer_processes	integer	1
gcs_server_processes	integer	0
global_txn_processes	integer	1
job_queue_processes	integer	4000
log_archive_max_processes	integer	4
processes	integer	300

```
SQL> select count(*) from v$process;
```

COUNT(*)
53

Das bedeutet, dass nur noch 247 Prozesse übrig bleiben, die Oracle akzeptieren wird. In diesem Fall können z.B. mit SQL*PLUS nur noch 247 Sessions geöffnet werden.

5 Fazit und Ausblick

5.1 Zusammenfassung

Die Ausrichtung auf die Cloud-Technologie erweist sich nicht nur für ORACLE als der richtige Schachzug, auch die SAP offeriert ihre HANA-Technologie in der Cloud, um damit ihren Kunden alle Vorteile einer Cloud zu bieten. Besonders interessant für Unternehmen ist das Pay-per-use-Prinzip, was bedeutet, dass der Kunde lediglich bezahlt was er verbraucht. Statt relativ kostenintensive Lizenzgebühren inklusive der Infrastruktur zu bezahlen, werden nur genutzte Dienste berechnet. Ebenfalls in dieser Arbeit konnte herausgearbeitet werden, dass auch die herkömmlichen Lösungen voll funktionstüchtig und dank modernster Containertechnik universell und überall gleichermaßen lauffähig sind. Bei einem Vergleich zwischen beiden Alternativen sollte die Entscheidung auf objektiver Basis bereits getroffen werden können. Warum viel Aufwand in den Datenbankaufbau investieren, wenn die All-Inclusive-Datenbank nach 5 Minuten Aufbau kostenfrei zur Verfügung steht. Doch möglicherweise gibt es auch den Puristen, der sich zurecht Sorgen um seine privaten Daten macht und nicht zu Unrecht argumentiert, wenn er feststellt, dass die Wirtschaft viel weniger von der Cloud geprägt ist, als Oracle es gerne hätte.

5.2 Aussichten

Oracle Database 20c ist als vorläufige Version verfügbar. Die wohl größte Neuerung ist die Multitenant-Architektur. Wie unter Experten vermutet [4], wird Oracle die NON-CDB-Architektur ab 20c nicht mehr unterstützen und nur noch auf die CDB-Architektur setzen. Eine mandantenfähige Container-Datenbank kann viele steckbare Datenbanken enthalten. Eine bestehende Datenbank kann leicht übernommen werden, ohne dass Anwendungsänderungen erforderlich sind. Das Gerücht über eine Express Edition der 20c-Variante ist ebenfalls eine begrüßenswerte Nachricht, wurde aber noch nicht bestätigt [1]. In Anbetracht der ausgearbeiteten Ergebnisse stellt sich die Frage, ob ein kostenloses Produkt, wie die angebotene Express Edition, für Oracle zukünftig noch von Nutzen sein wird. Als Konkurrenzlösung zu den bekannten, freien Datenbanken wie MySQL, das inzwischen selbst zur Produktpalette gehört, führte Oracle die Express Edition ein. Doch bei der Sorglosigkeit, mit der dieses Produkt behandelt wird, liegt die Vermutung nahe, dass es auch in Zukunft ausschließlich bei den kostenpflichtigen Varianten bleiben wird. Die Foren werden kaum verwendet und Oracle als Produkt ist zu mächtig, um für Hobbyprojekte ohne monetären Absichten eingesetzt zu werden. Die Alternativen mit PostgreSQL und MariaDB sind möglicherweise attraktiver und können dank Drittbibliotheken oder Anbietern wie knime.com für Data Mining genutzt werden.

5.3 Schlussfolgerung

Eine Entscheidungsfindung zwischen den beiden Varianten ist anspruchsvoll. Aufgrund der Notwendigkeit, Oracle einzusetzen, statt Alternativen wie PostgreSQL zu verwenden, die keine implementierte Technologie für maschinelles Lernen aufweisen, ist es notwendig, einen geeigneten Ansatz zu finden, um die erforderliche Funktionalität bereitzustellen, ohne dass viele der Funktionen der kostenpflichtigen Produkte verloren gehen. Der Vergleich hat gezeigt, dass die Wahl, falls sie getroffen werden sollte, bei der Hochschule auf die Cloud-Lösung fällt. Der Umfang der Möglichkeiten ist breit, und Oracle stellt jederzeit die neuesten Versionen zur Verfügung. Beispiele von erfahrenen Oracle Entwicklern zeigen wie viel mächtiger Oracle nun im Bereich der Business Intelligence geworden ist. Unternehmen wie Panasonic oder der Krisen-Sieger 2020 “Zoom” haben sicherlich einen Teil ihres Erfolgs dieser hocheffektiven Lösung zu verdanken. Der Zugriff ist, wenn konfiguriert, jederzeit und überall mög-

lich. Und der Administrator kann ohne grösseren Aufwand wechseln und verwalten. Dennoch kann eine Lösung innerhalb eines Containers in Betracht gezogen werden. Für Projekte abseits von Lehrveranstaltungen und Praktika, würde sie sich eignen, selbständig und unkompliziert Instanzen zu starten.

Dies ist gleichzeitig die abschließende Bewertung. Der Favorit ist die Cloud-Variante, wenn diese sorgfältig administriert wird. Die letzte Alternative sollte sein, die Installation eigenständig auf jedem Client-Rechner durchzuführen. Die Empfehlung des Autors ist es, ein Docker-Image zu erstellen, das von jedem Studierenden voll funktionsfähig für Lehrzwecke genutzt werden kann, einschließlich einer IDE innerhalb des Containers.

Literaturverzeichnis

- [1] BRUIJN, Niels de: *XE 20c will come, but XE 19c will be skipped.* – URL <https://twitter.com/nielsdb/status/1197156813839437825>. – Zugriffsdatum: 2020-07-08
- [2] BURLESON, Donald: *ORA-65096: create 12c user without c prefix.* – URL http://www.dba-oracle.com/t_ora_65096_create_user_12c_without_c_prefix.htm. – Zugriffsdatum: 2020-07-15
- [3] CLEVE, Jürgen ; LÄMMEL, Uwe: *Data Mining.* De Gruyter Studium, 2016. – ISBN 9783110456752
- [4] DIETRICH, Mike: *3 PDBs included with Oracle 19c and desupport of non-CDBs with 20c.* – URL <https://mikedietrichde.com/2019/09/17/3-pdb-included-with-oracle-19c-and-desupport-of-non-cdb-with-20c/>. – Zugriffsdatum: 2020-07-26
- [5] DUSOLD, Julia: *SAP wird profitabler und will 2020 kräftig wachsen.* – URL <https://www.produktion.de/wirtschaft/sap-wird-profitabler-und-will-2020-kraeftig-wachsen-383.html>. – Zugriffsdatum: 2020-06-15
- [6] GERASCH, Frank: *Oracle 18c XE verfügbar.* – URL <https://frankgerasch.de/2018/10/oracle-18c-xe-verfuegbar/>. – Zugriffsdatum: 2020-07-09
- [7] HORNICK, Mark: *R to Oracle Database Connectivity: Use ROracle for both Performance and Scalability (2018).* – URL <https://blogs.oracle.com/r/r-to-oracle-database-connectivity%3a-use-roracle-for-both-performance-and-scalability-2018>. – Zugriffsdatum: 2020-07-15
- [8] KÖSTERKE, Jessy: *Was sind eigentlich relationale Datenbanken?.* – URL <https://t3n.de/news/eigentlich-relationale-datenbanken-683688/2/>. – Zugriffsdatum: 2020-06-22

- [9] ORACLE: *Frequently Asked Questions*. – URL <https://www.oracle.com/cloud/free/?source=:ow:o:h:po:0HPPanel1nav0625&intcmp=:ow:o:h:po:0HPPanel1nav0625>. – Zugriffsdatum: 2020-07-14
- [10] ORACLE: *Oracle SQL Developer*. – URL <https://www.oracle.com/database/technologies/appdev/sqldeveloper-landing.html>. – Zugriffsdatum: 2020-07-15
- [11] ORACLE: *SingleInstance*. – URL <https://github.com/oracle/docker-images/tree/master/OracleDatabase/SingleInstance>. – Zugriffsdatum: 2020-07-08
- [12] ORACLE: *Using Oracle Machine Learning*. – URL <https://docs.oracle.com/en/cloud/paas/autonomous-data-warehouse-cloud/omlug/administer-oracle-machine-learning.html#GUID-E74F0E2E-EEE5-4421-A0BB-96A58811C04A>. – Zugriffsdatum: 2020-07-15
- [13] PLATTNE, Prof. H.: *In-memory data management. On-line-kurs auf openhpi.com*. – URL <https://openhpi.de/courses/e84ffbdd-4312-4efb-8590-2c10aaf03a49..> – Zugriffsdatum: 2020-06-26
- [14] SINGH, Arun K.: *AWS vs AZURE vs GCP vs Oracle*. – URL <https://medium.com/@arunksingh16/aws-vs-azure-vs-gcp-vs-oracle-1ba04484e6c4>. – Zugriffsdatum: 2020-06-15
- [15] SURAMPUDI, Sarika: *Data Mining Concepts*. – URL <https://docs.oracle.com/database/121/DMCON/GUID-8232ABAD-E6B9-4C70-B227-E00738040932.htm#DMCON111>. – Zugriffsdatum: 2020-06-29
- [16] WIKIPEDIA: *Machbarkeit* — *Wikipedia - Die freie Enzyklopädie*. – URL <https://de.wikipedia.org/w/index.php?title=Machbarkeit&oldid=194840326>. – Zugriffsdatum: 2020-07-04
- [17] WWW.BOXCRYPTOR.COM: *Cloud Definition – Was ist eine Cloud?*. – URL <https://www.boxcryptor.com/de/blog/post/what-is-the-cloud-a-beginner-s-guide/>. – Zugriffsdatum: 2020-06-16

- [18] WWW.CSRC.NIST.GOV: *The NIST Definition of Cloud Computing*. – URL
<https://www.csrc.nist.gov/publications/detail/sp/800-145/final>. –
Zugriffsdatum: 2020-06-16

A Abbildungsverzeichnis

2.1	Architektur der Servicemodelle	12
2.2	Der Data-Mining-Prozess	20
3.1	Entwicklungszyklus eines Datenbankbetriebs	23
3.2	Windows Docker	26
3.3	Windows Docker	26
3.4	Login für portainer.io	28
3.5	Dashboard von portainer.io	28
3.6	Ordner der Installation Version 18.4	29
3.7	Ausgabe buildDockerImage	30
3.8	Dashboard ORACLE Cloud	34
3.9	ATP-Datenbank Schritt 1	35
3.10	ATP-Datenbank Schritt 2	35
3.11	ATP-Datenbank Schritt 3	37
3.12	ATP-Datenbank Schritt 4	37
3.13	ATP-Datenbank Schritt 5	38
3.14	Docker-Anmeldung bei SQL Developer	39
3.15	Datenbankverbindung mit Instanz-Wallet	40
3.16	Datenbankverbindung mit Instanz-Wallet 2	41
3.17	Benutzerverwaltung in der Cloud	44
3.18	Data Miner-Verbindung im SQL Developer	45
3.19	Admin Homepage	47
3.20	Entwickler Homepage	47

B Tabellenverzeichnis

4.1	Vorteile der Alternativen	53
4.2	Nachteile der Alternativen	54
4.3	Der Vergleich von kritischen Anforderungen	55
4.4	Der Vergleich von weniger kritischen Anforderungen	56

Ich versichere, die vorliegende Arbeit selbstständig ohne fremde Hilfe verfasst und keine anderen Quellen und Hilfsmittel als die angegebenen benutzt zu haben. Die aus anderen Werken wörtlich entnommenen Stellen oder dem Sinn nach entlehnten Passagen sind durch Quellenangaben eindeutig kenntlich gemacht.

Hamburg, 03.08.2020

Patrick Jarren